







**Note**

Before using this information and the product it supports, read the information in "Notices" on page 167.

**December 2008**

This edition applies to version 6, release 2, modification 0 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 International Business Machines Corporation 2006, 2008. All rights reserved. US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© **Copyright International Business Machines Corporation 2006, 2008.**

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 of the Adapter for Siebel Business Applications . . . . .	3
Outbound processing . . . . .	3
Inbound processing . . . . .	8
Business objects . . . . .	12
External service wizard . . . . .	13

## Chapter 2. Planning for adapter implementation . . . . . 15

Before you begin . . . . .	15
Security . . . . .	15
Support for protecting sensitive user data in log and trace files . . . . .	15
User authentication. . . . .	16
Deployment options . . . . .	18
WebSphere Adapters in clustered environments . . . . .	20
Migrating to version 6.2 . . . . .	21
Migration considerations . . . . .	22
Performing the migration. . . . .	23
Updating but not migrating a version 6.0.2.x or version 6.1.x project . . . . .	24

## Chapter 3. Samples and tutorials . . . . . 27

## Chapter 4. Configuring the module for deployment. . . . . 29

Road map for configuring the module . . . . .	29
Configuring the Siebel application to work with the adapter. . . . .	31
Creating an authentication alias . . . . .	33
Creating the project. . . . .	35
Adding external software dependencies . . . . .	38
Configuring the module for outbound processing . . . . .	39
Setting connection properties for the external service wizard . . . . .	40
Selecting business objects and services . . . . .	43
Configuring the selected objects . . . . .	49
Setting deployment properties and generating the service . . . . .	51
Configuring the module for inbound processing . . . . .	55
Setting connection properties for the external service wizard . . . . .	55
Selecting business objects and services . . . . .	58
Configuring the selected objects . . . . .	64
Setting deployment properties and generating the service . . . . .	66

## Chapter 5. Changing interaction specification properties using the assembly editor . . . . . 73

## Chapter 6. Deploying the module . . . . . 75

Deployment environments . . . . .	75
Deploying the module for testing . . . . .	75
Adding external dependencies . . . . .	75
Generating and wiring a target component for testing inbound processing . . . . .	77
Adding the module to the server . . . . .	78
Testing the module for outbound processing using the test client. . . . .	79
Deploying the module for production . . . . .	80
Adding external software dependencies on the server . . . . .	80
Installing the RAR file (for modules using stand-alone adapters only) . . . . .	80
Exporting the module as an EAR file. . . . .	82
Installing the EAR file. . . . .	82

## Chapter 7. Administering the adapter module . . . . . 85

Changing configuration properties for embedded adapters . . . . .	85
Setting resource adapter properties for embedded adapters . . . . .	85
Setting managed (J2C) connection factory properties for embedded adapters. . . . .	87
Setting activation specification properties for embedded adapters. . . . .	89
Changing configuration properties for stand-alone adapters . . . . .	91
Setting resource adapter properties for stand-alone adapters . . . . .	91
Setting managed (J2C) connection factory properties for stand-alone adapters . . . . .	92
Setting activation specification properties for stand-alone adapters . . . . .	94
Starting the application that uses the adapter . . . . .	95
Stopping the application that uses the adapter. . . . .	96
Monitoring performance using Performance Monitoring Infrastructure. . . . .	96
Configuring Performance Monitoring Infrastructure. . . . .	97
Viewing performance statistics . . . . .	99
Enabling tracing with the Common Event Infrastructure (CEI) . . . . .	100

## Chapter 8. Troubleshooting and support . . . . . 103

Support for the Log and Trace Analyzer . . . . .	103
Configuring logging and tracing . . . . .	104
Configuring logging properties . . . . .	104

Changing the log and trace file names . . . . .	105
First-failure data capture (FFDC) support . . . . .	106
Business faults . . . . .	107
Solutions to some common problems . . . . .	110
XAResourceNotAvailableException . . . . .	113
Self-help resources. . . . .	113

**Chapter 9. Reference information . . . . . 115**

Business object information. . . . .	115
Application-specific information . . . . .	115
Supported operations. . . . .	117
Naming conventions for business objects representing Siebel business services, integration components, business objects, and business components . . . . .	117
Outbound configuration properties . . . . .	121
Connection properties for the wizard . . . . .	123
Resource adapter properties . . . . .	129
Managed connection factory properties. . . . .	134

Interaction specification properties . . . . .	139
Inbound configuration properties. . . . .	140
Connection properties for the wizard . . . . .	142
Resource adapter properties . . . . .	148
Activation specification properties . . . . .	153
Globalization . . . . .	161
Globalization and bidirectional data transformation . . . . .	161
Properties enabled for bidirectional data transformation . . . . .	163
Adapter messages. . . . .	164
Related information . . . . .	164

**Notices . . . . . 167**

Programming interface information . . . . .	169
Trademarks and service marks . . . . .	169

**Index . . . . . 171**

---

## 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 application, without special coding.

WebSphere Adapter for Siebel Business Applications provides a way to create integrated processes that exchange information with a Siebel application. With the adapter, an application can send requests to the Siebel Business Applications server or receive notifications of changes from the server.

The adapter creates a standard interface to the applications and data on the Siebel Business Applications server, so that the application does not have to understand the lower-level details (the implementation of the application or the data structures) on the Siebel Business Applications server. An application, for example, can send a request to the Siebel Business Applications 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, to be notified that a customer record has been updated. This provides you with improved business workflow and processes to help manage your customer relations.

WebSphere Adapter for Siebel Business Applications 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. WebSphere Adapter for Siebel Business Applications makes it possible for JCA-compliant application servers to connect to and interact with the Siebel server in a standard way by using business objects or JavaBeans™.

The following example assumes you are setting up an adapter using WebSphere Integration Developer and deploying the application that includes the adapter to WebSphere Process Server.

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, needs to first obtain a list of customers.

The Siebel Business Applications function does not have a Web service interface, however, so the application component used by the promotions department needs to understand the low-level API and data structures of the Siebel Business Applications function in order to make the call to the function. Information technology resources are needed to create the links between the component and the Siebel Business Applications function.

With WebSphere Adapter for Siebel Business Applications, 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 either 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 Process 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 WebSphere Integration Developer, 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 does not have to deal directly with the Siebel Business Applications function; it is the Siebel Business Applications 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 Adapter for Siebel Business Applications. In return, the application component receives the results (the list of customers) in the form of a standard business object. The application component does not need to 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 notifies 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 Web site.

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 and click **Compatibility Matrix** beneath the **Related** heading in the **Additional support links** section: <http://www.ibm.com/software/integration/wbiadapters/support/>.
- Technotes for WebSphere Adapters provide workarounds and additional information that are not included in the product documentation. To view the technotes for your adapter, go to the following Web page, select the name of your adapter from the **Product category** list, and click the search icon: <http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>.



---

## Technical overview of the Adapter for Siebel Business Applications

WebSphere Adapter for Siebel Business Applications supports the exchange of information between your existing 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 application), and inbound processing (event notification from a Siebel application server to an application).

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

Adapter for Siebel Business Applications is metadata driven, which means that after you import the adapter into WebSphere Integration Developer, you can continue to add new business objects to the existing module. The adapter can process new business objects without additional coding or customization.

### Outbound processing

WebSphere Adapter for Siebel Business Applications supports synchronous outbound processing. This means that when 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 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 perform 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, Delete), the adapter populates that Siebel object (business or integration component hierarchy) with data from the hierarchy of WebSphere business objects.

### Supported outbound operations

WebSphere Adapter for Siebel Business Applications supports the following outbound operations: Apply Changes, 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.
Delete	Deletes the business component and its children.
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 and populates it as the container business graph.
Update	Updates the Siebel application with the incoming business object.

## Outbound operations

Application components need to invoke operations such as data retrieval from a database. An operation is the action that an adapter can perform on the database during outbound processing. The name of the operation indicates the type of action that the adapter takes.

The 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 is one that contains the complete state of the business object after all desired changes have been made to it. A delta business object is one that contains only key values and the data to be changed. Delta business objects are used only in operations that update 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 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. ChangeSummary 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 the Siebel application.
- 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 the Siebel application but is not updated. The retrieval is done so that any further updates needed to the business object's children can be processed at a later time.

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

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

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

- With a Create operation, ChangeSummary 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 as well as any old values (existing values from the Siebel application). Because it is a Create operation, and the values are new, they 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 for the Delete operation, no new values are created.

### **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 children is created first and then the parent is created. After all of the children are created, the parent is saved. By saving the parent after creating all the children, you can roll back all of the children 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. Siebel 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 new child business object. Instead, set the value on the parent. For example, if you pick an existing `Business Address` instance under `Account`, set the `PrimaryAddressId` instead of creating a `Business Address` instance. However, to create a new child instance, you must populate the non-key attributes in the child business component so that the adapter will try 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 `Picklist`'s list of values, the adapter tries to populate a value. If the list is unbounded and `Insert` is enabled, the operation will be successful. If the adapter is unable to populate the value by itself, the operation will fail.
- With multi value links, the child record is created. If the child `Create` operation fails, the child is rolled back. If the parent `Create` operation fails, all of the children and the parent are rolled back.
- The adapter attempts to select an existing child business component instance (for example, `Association`) based on the value of all of 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 is rolled back. If the parent `Create` operation fails, only the parent is rolled back. Because association components can also be individual components, as well as children of other components, a child or parent cannot be rolled back once an associated child has been created.

#### **Delete operation:**

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

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 will fail.

#### **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 children 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. Before the adapter retrieves the child, the `From` attribute value from the parent must be set against the `To` attribute in the child. If the `From` attribute is not set, the adapter retrieves the child without setting any values on the `To` attribute. For example, only `executeQuery` is called.
- With multi-value link and association, if the child record is found, the attribute value is set on the parent.

#### **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 have key and non-key attributes set. All of 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, it will be a 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 will be applied to the operation if it has already been set. The business object that is returned will be a container business object of the business graph.

The steps involved in processing the RetrieveAll operation is similar to the Retrieve operation, except the adapter does not check to make sure that all of the keys have been 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 no records were 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 children will be processed and updated. If the child Create operation fails because the record already 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. This means that the adapter polls the Siebel Business 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 for the Siebel event business component or integration component 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. To keep track of inbound events as they make their way through the system, the adapter uses an event store.

The creation, update, or deletion of an event record in the Siebel business application 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 continually updated by the adapter until the events are delivered to a configured export in WebSphere Process Server or WebSphere Enterprise Service Bus.

For example, if you have a customer component and a new customer has just been added to it, this signals an update. If the adapter is configured to receive the events about the new update, there will be 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 was 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. Do this in the ActivationSpec 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 of 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
Object key	A unique identifier that provides the key name and value of the actual component instance that was updated (for example: An Account component instance was updated and it triggers to insert an event into the event component). The event field will hold the key name and value of the Account component instance as a name-value pair for the adapter to do the retrieval later on when polling.	Id=1-20RT
Object name	The name of the business object, corresponding to the Siebel business component for which the event was detected.	B0AccountBCAccount
Event type	The type of event such as Create, Update, or Delete.	Create
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"> <li>• 0: Identifies a new event.</li> <li>• 1: Identifies an event that has been delivered to an export.</li> <li>• -1: An error occurred while processing the event.</li> </ul> This column cannot be null.	0
Description	Any comment associated with the event.	Account create Event
Event ID	The key field ID of the event component. The value will be auto-generated by Siebel.	
Event timestamp	The time stamp for the event. The format is MM/dd/yyyy HH:mm:ss	02/24/2007 11:37:56



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

Field	Description	Example
XID	The transaction ID. This is to ensure 'assured once delivery'. This will remove the dependency on the EDT. For more information on XID, check the AFC FS.	
CONNECTOR ID	The ID set for the Siebel adapter	Siebel001

## Event store structure for Siebel business services

The event is retrieved from the 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 Siebel business services

Field	Description	Example
Object key	A unique identifier that provides the key name and value of the actual component instance that was updated (for example: An Account component instance was updated and it triggers to insert an event into the event component). The event field will hold the key name and value of the Account component instance as a name-value pair for the adapter to do the retrieval later on when polling.	Name=TestName;Location=BGM, where 'Name' and 'Location' are the keys in the integration component 'Account' under Integration object 'Acocunt (PRM ANI) and 'TestName' and 'BGM' are the values specified, and ; is the event key delimiter.
Object name	The name of the business object, corresponding to the Siebel business component for which the event was detected.	IOAccountPRMANIICAccount
Event type	The type of event such as Create, Update, or Delete.	Create
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"> <li>• 0: Identifies a new event.</li> <li>• 1: Identifies an event that has been delivered to an export.</li> <li>• -1: An error occurred while processing the event.</li> </ul> This column cannot be null.	0
Description	Any comment associated with the event.	Account PRM ANI Event
Event ID	The key field ID of the event component. The value will be auto-generated by Siebel.	

Table 3. Event store structure for Siebel business services (continued)

Field	Description	Example
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
XID	The transaction ID. This is to ensure 'assured once delivery'. This will remove the dependency on the EDT. For more information on XID, check the AFC FS.	
CONNECTOR ID	The ID set for the Siebel adapter	Siebel001

#### Related tasks

“Configuring the Siebel application to work with the adapter” on page 31  
 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 or an employee record, or unstructured text.

### How business objects are created

You create business objects by using the external service wizard, launched from WebSphere Integration Developer. 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 children 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 in version 6.0.2 to specify additional information about the operation to be performed. In version 6.2, 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 a version of WebSphere Integration Developer earlier than version 6.2

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

#### Related reference

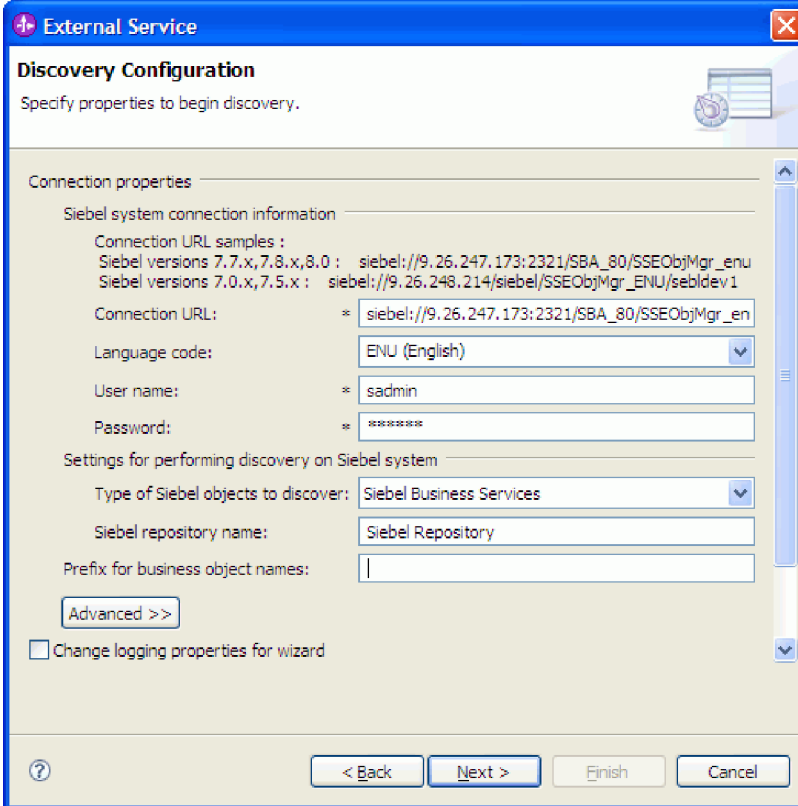
“Business object information” on page 115

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

## External service wizard

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

Using WebSphere Integration Developer, you establish a connection to the Siebel server to browse the metadata repository on the Siebel server. You also specify connection information, such as the Connection URL, user name, and password needed to access the server as shown in the following figure.



The screenshot shows the 'External Service' window with the 'Discovery Configuration' tab selected. The window title is 'External Service' and the subtitle is 'Discovery Configuration'. Below the subtitle, it says 'Specify properties to begin discovery.' The main area is divided into several sections:

- Connection properties**:
  - Siebel system connection information**:
    - Connection URL samples :
      - Siebel versions 7.7.x, 7.8.x, 8.0 : siebel://9.26.247.173:2321/SBA\_80/SSEObjMgr\_enu
      - Siebel versions 7.0.x, 7.5.x : siebel://9.26.248.214/siebel/SSEObjMgr\_ENU/sebldev1
    - Connection URL: \* siebel://9.26.247.173:2321/SBA\_80/SSEObjMgr\_en
    - Language code: ENU (English) (dropdown menu)
    - User name: \* sadmin
    - Password: \* (masked)
  - Settings for performing discovery on Siebel system**:
    - Type of Siebel objects to discover: Siebel Business Services (dropdown menu)
    - Siebel repository name: Siebel Repository
    - Prefix for business object names: (empty text field)
- Advanced >>** (button)
- Change logging properties for wizard

At the bottom of the window, there are four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'. A help icon (?) is located in the bottom left corner.

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 WebSphere Process Server 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.

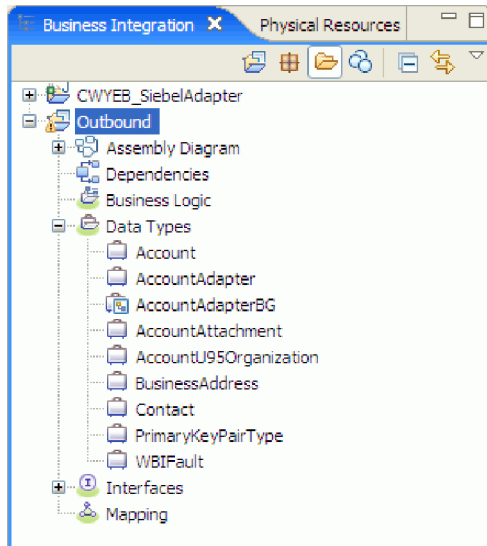


Figure 2. An example of the Outbound module generated by the external service wizard

---

## Chapter 2. Planning for adapter implementation

Before you configure WebSphere Adapter for Siebel Business Applications Software, consider whether you will 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 should possess a thorough understanding of business integration concepts, the capabilities and requirements of the integration development tools and runtime environment you will use, and the Siebel server environment where you will build and use the solution.

To configure and use WebSphere Adapter for Siebel Business Applications, you should 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 will use to build the solution. You should 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 will use for the integration solution. You should 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 should 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 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 WebSphere Process Server or WebSphere Enterprise Service Bus.

---

### Support for protecting sensitive user data in log and trace files

The adapter provides the ability to prevent sensitive or confidential data in log and trace files from being seen by those 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

must 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 situations like this, the adapter lets you specify whether or not 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 134 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 129

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

“Managed connection factory properties” on page 134

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 the Siebel application to extract, or *discover*, information about the objects and services that you can access with the adapter.
- At run time on WebSphere Process Server or WebSphere Enterprise Service Bus, when the adapter connects to the Siebel application to process outbound requests and inbound events.

## Authentication in the wizard

The external service wizard asks for 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 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 completes.

## 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. The external service wizard lets you configure the adapter to get the user information by using 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 the 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 accesses that Siebel application, including the adapters embedded in application EAR files as well as adapters that are separately installed on the server.

By using a data source, you can 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 can 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 33

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.

---

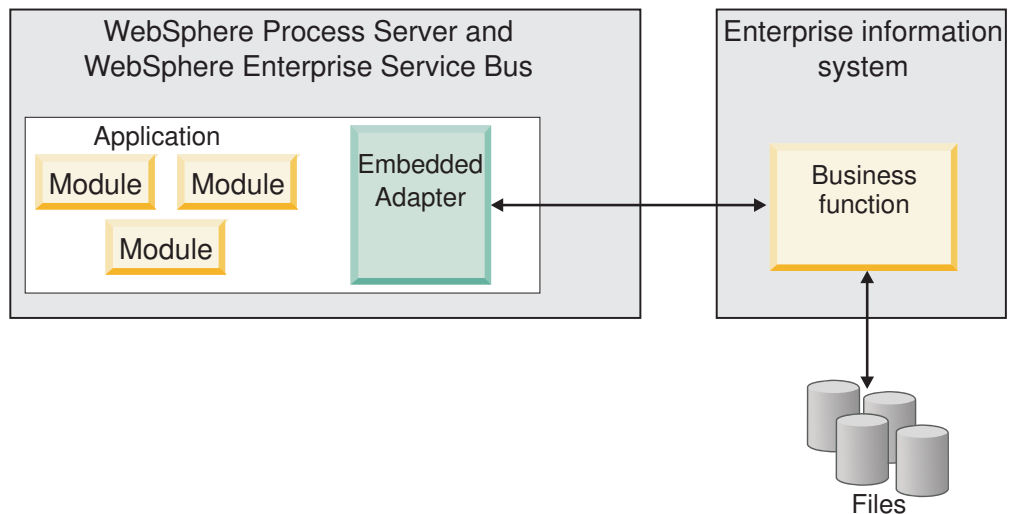
## Deployment options

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 will affect the type of deployment option you choose.

The deployment options are described below:

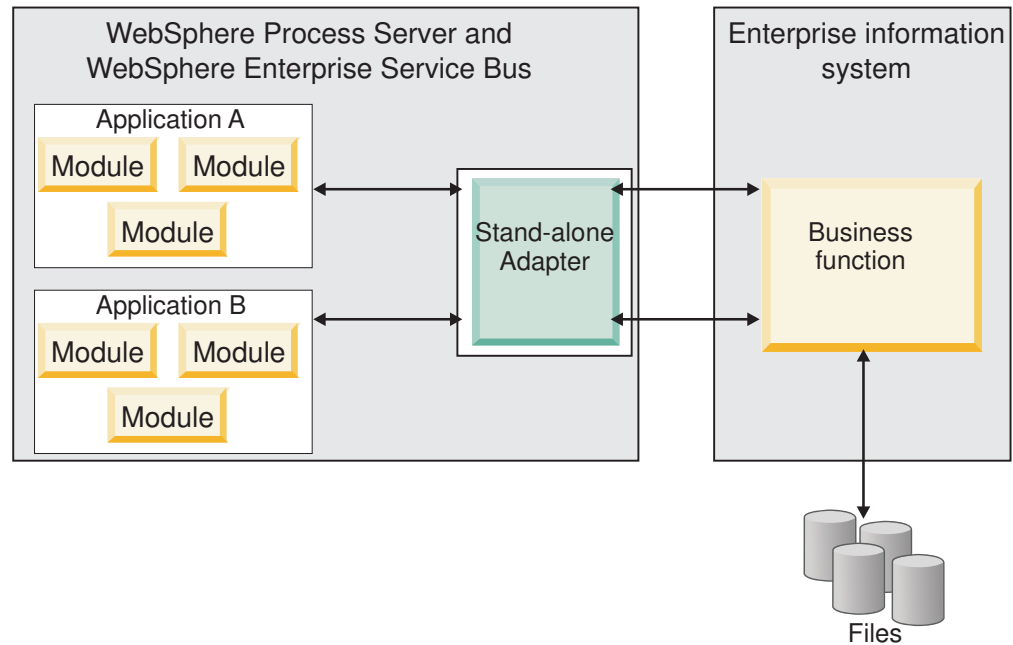
- **With module for use by single application.** With the adapter files embedded in the module, you can deploy the module to any application server. Use an embedded adapter when you have a single module using the adapter or if multiple modules need to run different versions of the adapter. Using an embedded 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 WebSphere Integration Developer, you can choose how to package the adapter [either bundled with the (EAR) file or as a stand-alone (RAR) file]. Your choice will affect how the adapter is used in the runtime 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 on running 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

Take into consideration the following items if you plan on embedding 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 runtime environments. *Class loader isolation* means 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

Take into consideration the following items if you plan on using 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.

---

## WebSphere Adapters in clustered environments

You can improve adapter performance and availability by deploying the module to a clustered server environment. The module is replicated across all servers in a cluster, regardless of whether you deploy the module using a stand-alone or embedded adapter.

WebSphere Process Server or WebSphere Enterprise Service Bus, WebSphere Application Server Network Deployment, and WebSphere Extended Deployment support clustered environments. Clusters are groups of servers that are managed together to balance workloads and to provide high availability and scalability. When you set up a server cluster, you create a Deployment Manager profile. The HAManager, a subcomponent of the Deployment Manager, notifies the Java 2 Platform, Enterprise Edition (J2EE) Connector Architecture (JCA) container to activate the adapter instance. The JCA container provides a runtime environment for adapter instances. For information about creating clustered environments, see the following link: [http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun\\_wlm\\_cluster\\_v61.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/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 instead of a static workload manager, which is used by WebSphere Application Server Network Deployment. The dynamic workload manager 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 machines with different capacities

and configurations to evenly handle load variations. For information on the benefits of WebSphere Extended Deployment, see the following link: <http://publib.boulder.ibm.com/infocenter/wxdinfo/v6r1/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 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.

**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 on workload management, see the following link: [http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun\\_wlm.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm.html).

---

## **Migrating to version 6.2**

By migrating to version 6.2 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 utilize features and capabilities present in version 6.2.

## Migration considerations

WebSphere Adapter for Siebel Business Applications version 6.2 includes updates that might affect your existing adapter applications. Before migrating applications that will 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 6.2 is fully compatible with version 6.0.2.x and version 6.1.x of the adapter and can work with custom business objects (XSD files) and data bindings.

Because version 6.2 of WebSphere Adapter for Siebel Business Applications is fully compatible with version 6.0.2.x and version 6.1.x, any of your applications that used version 6.0.2.x or version 6.1.0.x of WebSphere Adapter for Siebel Business Applications will run unchanged when you upgrade to version 6.2. However, if you want your applications to use features and functionality present in version 6.2 of the adapter, run the migration wizard.

The migration wizard replaces (upgrades) version 6.0.2.x or version 6.1.x of the adapter with version 6.2 and enables version 6.2 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 6.2 of the adapters. If you are upgrading to version 6.2 and any of your applications embed an adapter that is version 6.1.x or earlier, and you want your applications to take advantage of the features and functions in version 6.2, you might need to make changes to those applications.

If artifacts are inconsistent with regard to versioning within a single module, the entire module is marked and will not be selectable for migration. Version inconsistencies are recorded in the workspace log, because they might indicate corruptions to the project.

If the WebSphere Integration Developer version does not match that of WebSphere Adapter for Siebel Business Applications, the products cannot be run together.

The external service wizard cannot be run when the following product combinations are used:

*Table 4. Product combinations not supported*

WebSphere Integration Developer	WebSphere Adapter for Siebel Business Applications
version 6.2	version 6.1.x
version 6.1.x	version 6.0.2.x
version 6.0.2.x	version 6.2
version 6.0.2.x	version 6.1.x

### Deciding whether to upgrade or to upgrade and migrate

By default, the migration wizard upgrades the adapter and migrates the application artifacts so that the applications can use features and functions in

version 6.2 of the adapter. When you choose to upgrade the adapter by selecting a project, the wizard automatically selects the associated artifacts for migration.

If you decide that you want to upgrade the adapter from version 6.0.2.x or version 6.1.x to version 6.2, 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 any adapter artifacts selected will install and upgrade your adapter, but your artifacts will not be migrated and your applications will not be able to take advantage of the features and capabilities that exist in version 6.2 of the adapter.

## Running the migration wizard in a test environment first

Because adapter migration might require you to make changes to those applications that will use version 6.2 of WebSphere Adapter for Siebel Business Applications, you should 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 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 6.2, by 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 WebSphere Integration Developer, complete the following steps.

**Note:** After migration is complete, the module will no longer be compatible with previous versions of WebSphere Process Server or WebSphere Enterprise Service Bus, or WebSphere Integration Developer.

**Note:** The following steps describe how to run the adapter migration wizard from the connector project context menu while in the Java EE perspective in WebSphere Integration Developer.

### Procedure

1. Import the PI (project interchange) file for an existing project or the EAR (enterprise archive) file for a deployed application into the workspace.

2. Change to the Java EE perspective.
3. Right-click the module and select **Update** → **Update Connector Project**.

**Note:** Additionally, you can launch the adapter migration wizard in the following ways:

- Right-click the project in the Java EE perspective and select **Update** → **Migrate Adapter Artifacts**.
- From the Problems view, right-click a migration-specific message and select **Quick Fix** to correct the problem.

When you launch the migration wizard from the connector project context menu while in the Java EE perspective, by default all of the dependent artifact projects are selected. If you deselect a dependent artifact project, that project is not migrated. You will be allowed to migrate any projects that you deselect 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.

4. Review the tasks and warnings presented on the welcome page, and click **Next**.
5. A warning window opens with the message, “The properties that are not supported in the version of the target adapter will be removed during the migration.” Click **OK** to continue to the next step.
6. On the Review changes window you can review the migration changes that will occur in each of the artifacts that you are migrating by clicking the + sign.
7. Click **Finish** to perform the migration.

Prior to running the migration process, the wizard performs a backup up of all projects affected by the migration. The projects are backed up to a temporary folder within the workspace. Should the migration fail for any reason, or should 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. Check the Problems view for messages from the migration wizard, which start with the string CWPAD.
9. If you are migrating an EAR file, optionally create a new EAR file with the migrated adapter and artifacts, and deploy it to WebSphere Process Server or WebSphere Enterprise Service Bus. For more information about exporting and deploying an EAR file, see the topics devoted to it in this documentation.

## Results

The project or EAR file is migrated to version 6.2. You do not need to run the external service wizard after exiting the adapter migration wizard.

## Updating but not migrating a version 6.0.2.x or version 6.1.x project

You can upgrade the adapter from version 6.0.2.x or version 6.1.x to version 6.2, while choosing not to migrate the adapter project artifacts.

### About this task

Because the internal name of the adapter changed in version 6.2, artifacts in a version 6.0.2.x or version 6.1.x project must be updated to use the new name before you can use the adapter wizard in WebSphere Integration Developer, version 6.2. Use the migration wizard to update a version 6.0.2.x or version 6.1.x

project. Then use the Quick Fix feature of WebSphere Integration Developer to change the adapter name in project artifacts.

### Procedure

1. Import the project interchange (PI) file into the workspace.
2. In the Java EE perspective, right-click the project name and click **Update** → **Update Connector Project**. The adapter migration wizard opens.
3. On the welcome page, click **Next**.
4. On the Select Projects window, deselect the dependent artifact projects, and then click **Finish**.
5. A warning window opens with the message, "The properties that are not supported in the version of the target adapter will be removed during the migration." Click **OK** to continue to the next step.
6. In the Quick Fix window, make sure the fix **Rename the referenced adapter** is selected, and then click **OK**.
7. If the error remains visible, click **Project** → **Clean**, select the project you just updated, and then click **OK**.

### Results

The project can now be used with WebSphere Adapter for Siebel Business Applications, version 6.2.





---

## Chapter 3. Samples and tutorials

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

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

- From the welcome page that opens when you start WebSphere Integration Developer. To see samples and tutorials for WebSphere Adapter for Siebel Business Applications, click **Retrieve**. Then browse the displayed categories to make your selections.
- At this location on the Web: Business Process Management Samples and Tutorials <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 WebSphere Process Server or WebSphere Enterprise Service Bus, use WebSphere Integration Developer 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 WebSphere Integration Developer. 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.

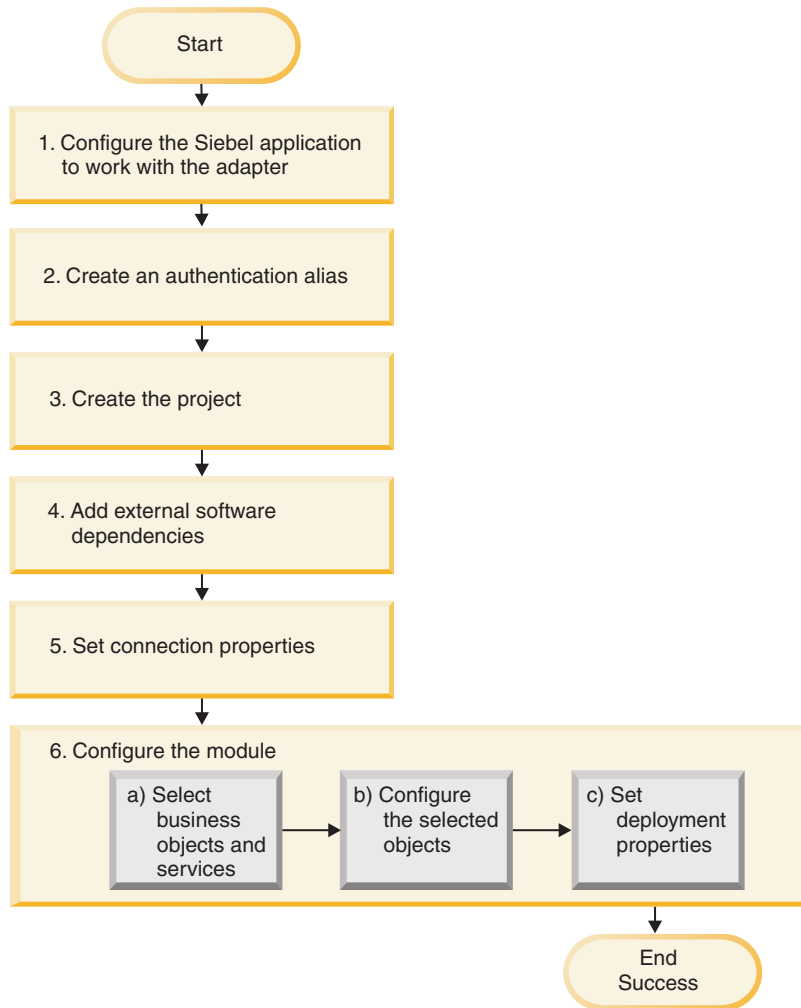


Figure 3. Road map for configuring the module

### Configuring the module for deployment

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 WebSphere Integration Developer 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 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 of 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 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 to retrieve Siebel business objects and services.

Also, because you will perform this task on the Siebel 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 will need to open Siebel Sales Enterprise (or the Siebel application you are using) on your local database, and to do this, you must have administrative privileges.

### About this task

To configure the Siebel application, you will create an event table and 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 in which to store the events.
  - a. In the event table, create the columns shown in the following table.

*Table 5. 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 5. 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 new business component called IBM Event.
  - c. Create a new 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 new 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 new 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\_Q' 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 new 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 new 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 new 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 on 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 server, and replace the original repository file with the newly created one.
  14. Restart the Siebel 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. To keep track of inbound events as they make their way through the system, the adapter uses an event store.

---

## **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 WebSphere Process Server or WebSphere Enterprise Service Bus. You must also know the user name and password to use to connect to the Siebel application.

The following procedure shows you how to gain access to the administrative console through WebSphere Integration Developer. If you are using the administrative console directly (without going through WebSphere Integration Developer, log in to the administrative console and skip to step 2 on page 34.

### **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 WebSphere Integration Developer, perform the following steps:

  - a. In the Business Integration perspective of WebSphere Integration Developer, click the **Servers** tab.
  - b. If the server does not show a status of **Started**, right-click the name of the server (for example, **WebSphere Process Server** 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 will use when you configure the adapter properties.

### Related concepts

“User authentication” on page 16

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, start the external service wizard in WebSphere Integration Developer. The wizard creates a project that is used to organize the files associated with the module.

### Before you begin

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

### About this task

Start the external service wizard to create a project for the adapter in WebSphere Integration Developer. If you have an existing project, you can select it instead of having the wizard create one.

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

### Procedure

1. To start the external service wizard, go to the Business Integration perspective of WebSphere Integration Developer, and then click **File** → **New** → **External Service**.
2. In the External Service window, make sure Siebel from the **Available Types** of adapters is selected, and click **Next**.

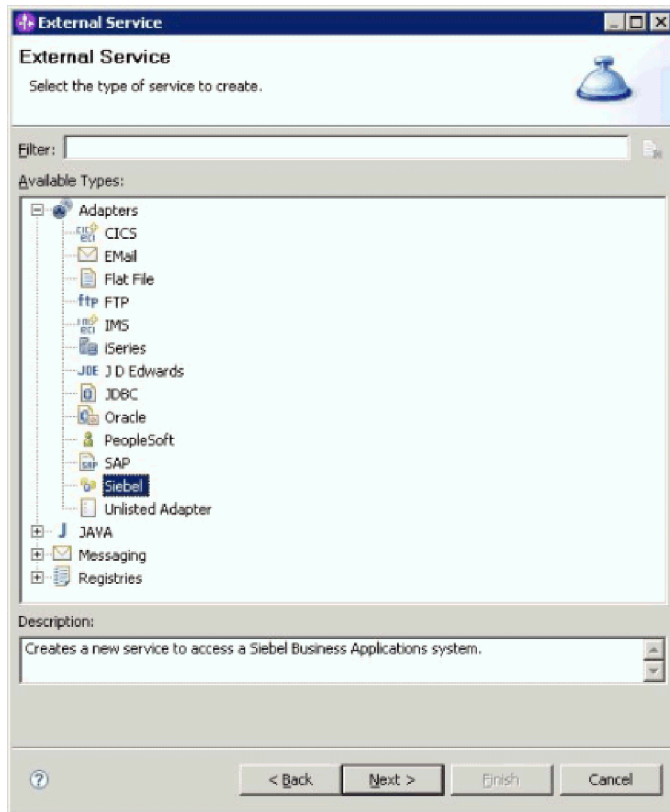


Figure 4. The External Service window

3. From the Select an Adapter window, you can either create a project or select an existing project.
  - To create a project, perform the following steps:
    - a. Select **IBM WebSphere Adapter for Siebel Business Applications (IBM : 6.2.0.0)**, and click **Next**.
    - b. In the Adapter import window, either accept the default name for the project (**CWYEB\_SiebelAdapter**) or provide another name in the **Connector project** field, select the server in the **Target server** field (for example, **WebSphere Process Server v6.2**), and click **Next**.
  - To select an existing project, perform the following steps:
    - a. Expand **IBM WebSphere Adapter for Siebel Business Applications (IBM : 6.2.0.0)**.
    - b. Select a project. For example, if you have an existing project named **CWYEB\_SiebelAdapter**, you can expand **IBM WebSphere Adapter for Siebel Business Applications (IBM : 6.2.0.0)**, and select **CWYEB\_SiebelAdapter**, as shown in the following figure.

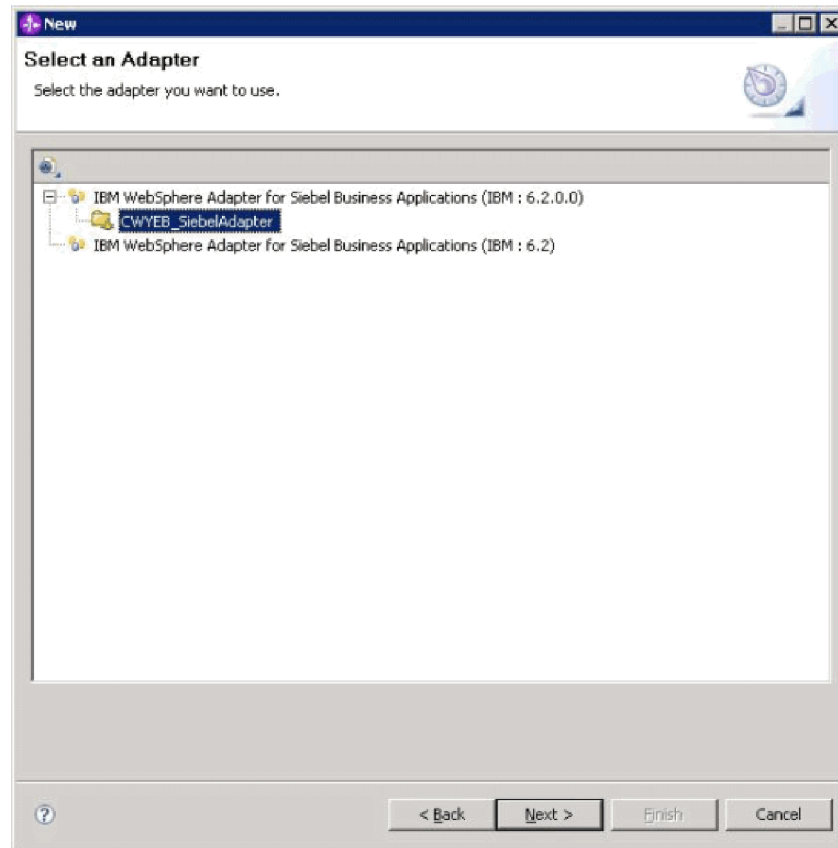


Figure 5. The Select an Adapter window

4. Click Next.

## Results

If you created a new project or selected an existing project, it is listed in the Business Integration window.

- If you created a new project, you will be prompted by the external service wizard to provide the location of the external software dependency files required to connect to the Siebel Business Applications server.
- If you selected an existing project, and that project does not already have the required external software dependency files associated with it, you will be prompted by the external service wizard to provide the location of the external software dependency files required to connect to the Siebel Business Applications server.
- If you selected an existing project, and that project already has the required external software dependency files associated with it, you will be prompted by the external service wizard to specify the information required to establish a connection between the external service wizard and the Siebel Business Applications server.

---

## Adding external software dependencies

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

### Before you begin

You should be running the external service wizard in WebSphere Integration Developer 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 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 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, and 8.0)
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 a local folder on the workstation where you run the external service wizard.
3. From 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.

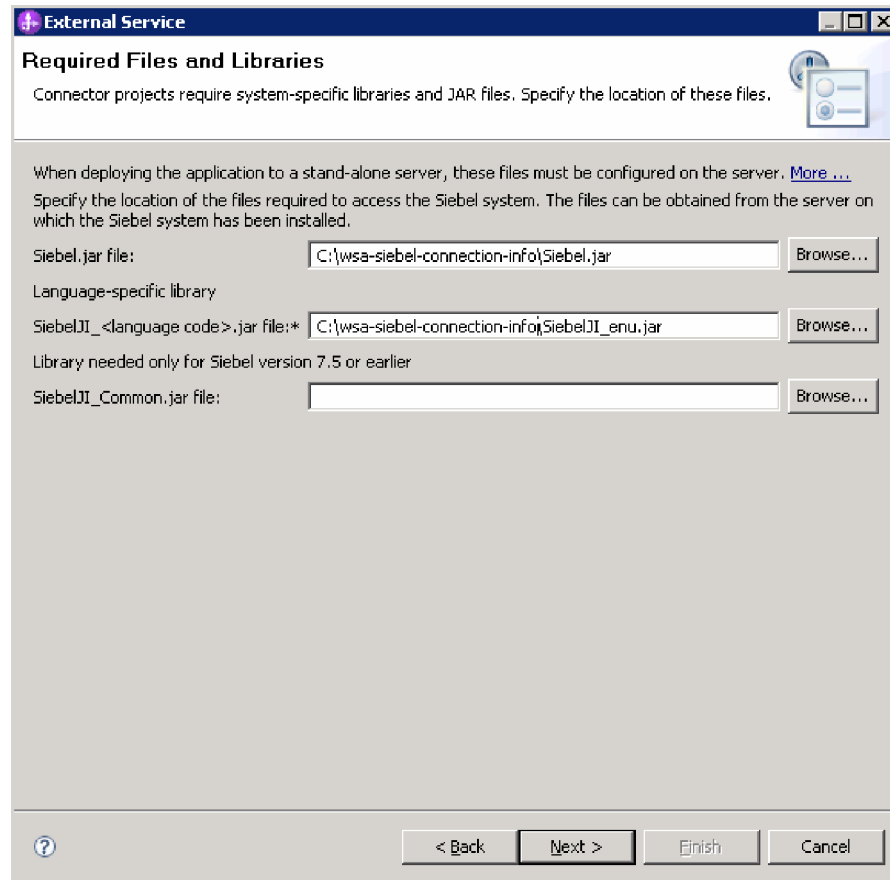


Figure 6. The Required Files and Libraries window, showing the required files selected for Siebel Business Applications, versions 7.7x, 7.8x, and 8.0

4. Click Next.

### Results

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

### What to do next

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

---

## Configuring the module for outbound processing

To configure a module to use the adapter for outbound processing, use the external service wizard in WebSphere Integration Developer to find and select business objects and services from the Siebel application, and to generate business object definitions and related artifacts.

### Related concepts

“Outbound processing” on page 3

WebSphere Adapter for Siebel Business Applications supports synchronous outbound processing. This means that when 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.

## Setting connection properties for the external service wizard

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

### Before you begin

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

### About this task

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

To specify the connection properties, use the following procedure.

### Procedure

1. On the Adapter Style window, select **Outbound** (to send data to the Siebel server from the adapter), and click **Next**.
2. In the Discovery Agent Configuration window, specify the configuration properties:
  - a. In the **Connection URL** field, type the URL needed to connect to the Siebel server.
  - b. If necessary, change the default setting (ENU - English) for **Language code** by selecting another language from the drop down list.
  - c. Type the username and password you use to access the Siebel server, as shown in the following figure.  
The password is case-sensitive.

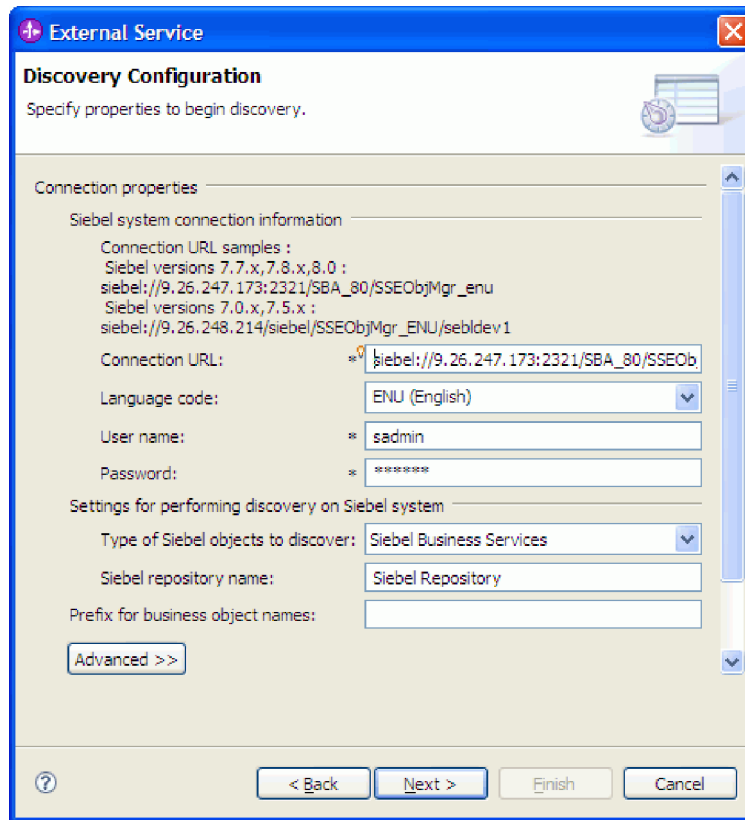


Figure 7. The Discovery Configuration window

- d. If necessary, change the default setting for **Type of Siebel objects to discover** (Siebel Business Services) by selecting Siebel Business Objects from the drop down list.
  - e. If necessary, change the default setting for **Siebel repository name** by typing in the name of the repository from which the objects are to be discovered.
  - f. If necessary, type a prefix in the **Prefix for business object names** field to have it attached to the front of the business object name.
3. To set additional advanced properties, click **Advanced**. When you select **Advanced**, the related properties are displayed.

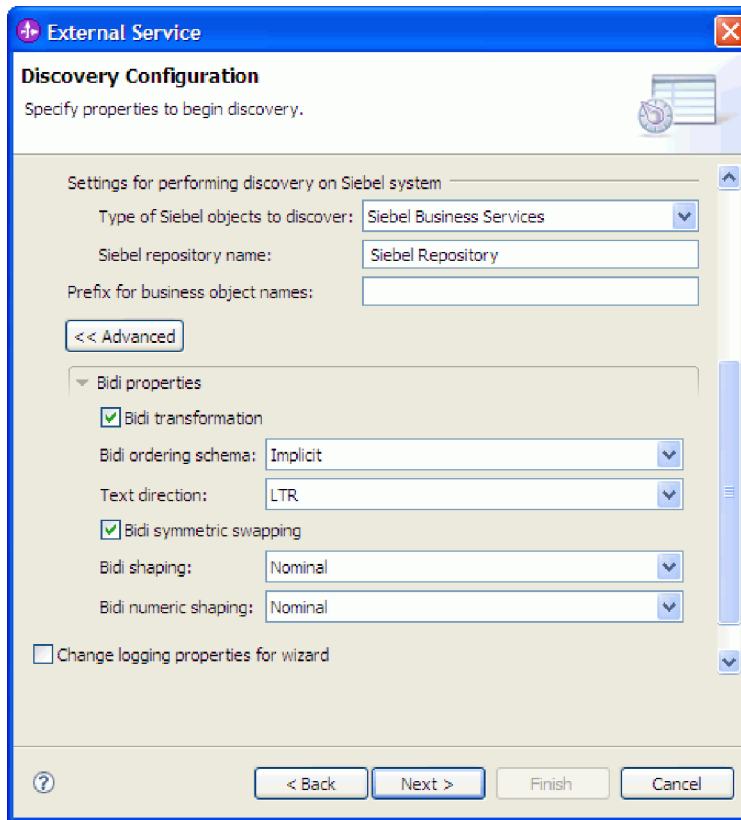


Figure 8. The Discovery Configuration window, showing the additional advanced property sections after clicking the **Advanced** button

4. If you need to set bidirectional properties, perform the following steps:
  - a. In the Bidi Properties section, select **Bidi transformation**.
  - b. Set properties for your environment. See “Connection properties for the wizard” on page 123 for more information about these properties.
5. To set logging properties for the external service wizard, perform the following steps:

- a. Select **Specify the level of the logging desired**.
- b. Change the location of the log file output location by clicking **Browse** and selecting a different location.

- c. Set the **Logging Level**.

In a test environment, select **FINEST**, which provides the highest level of logging. In a production environment, select a level lower than **FINEST** to optimize the logging process.

**Note:** This log pertains to the external service wizard only, not to the operation of the adapter.

6. Click **Next**.

## Results

The external service wizard contacts the Siebel server using the information you provided (such as user name and password) to log in. You see the Object Discovery and Selection window.



## What to do next

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

### Related reference

“Connection properties for the wizard” on page 123

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

## Selecting business objects and services

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

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

### Selecting business objects

1. In the Discovery Configuration window, select **Siebel Business Objects** as the **Type of Siebel Objects to discover**. The check box **Prompt for additional configuration settings when adding a business object** is displayed.
2. Select the check box **Prompt for additional configuration settings when adding a business object**. You need to select this if you want to select the attributes of the business objects.
3. Click **Next**.
4. In the Object Discovery and Selection window,
  - a. Click **Edit Query**.
  - b. In 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 Object Discovery and Selection window.

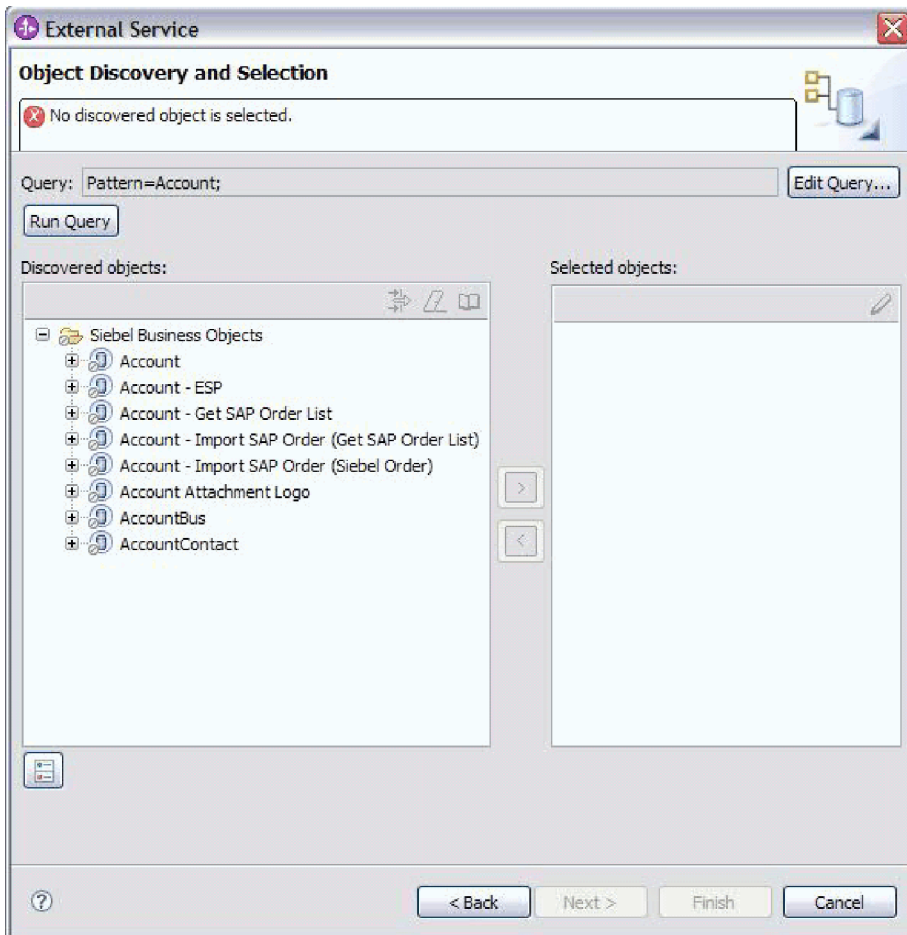


Figure 9. Object Discovery and Selection window, showing the Siebel Business Objects folder expanded with the list of discovered objects

- e. Expand the required business objects (for example, Account), to see the business components that you want to work with.

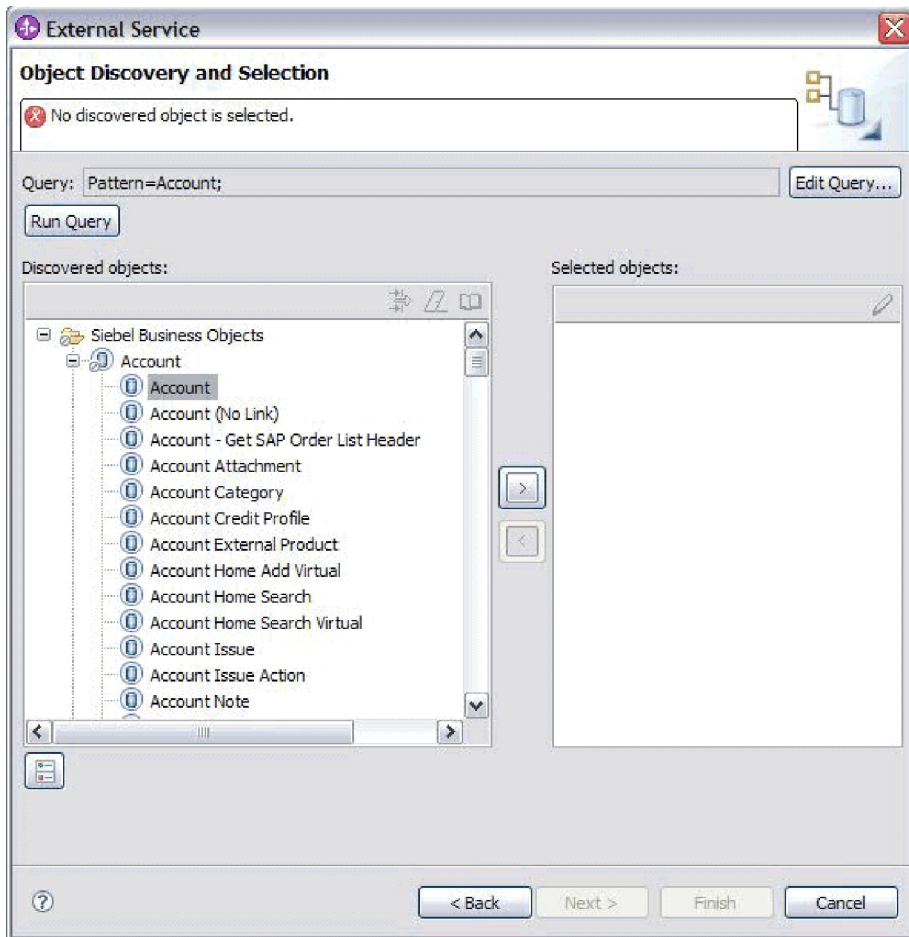


Figure 10. Object Discovery and Selection window, showing the Siebel Business Objects folder expanded with the list of discovered objects

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

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

- g. Select the attributes you want to generate from the Configuration Properties window.

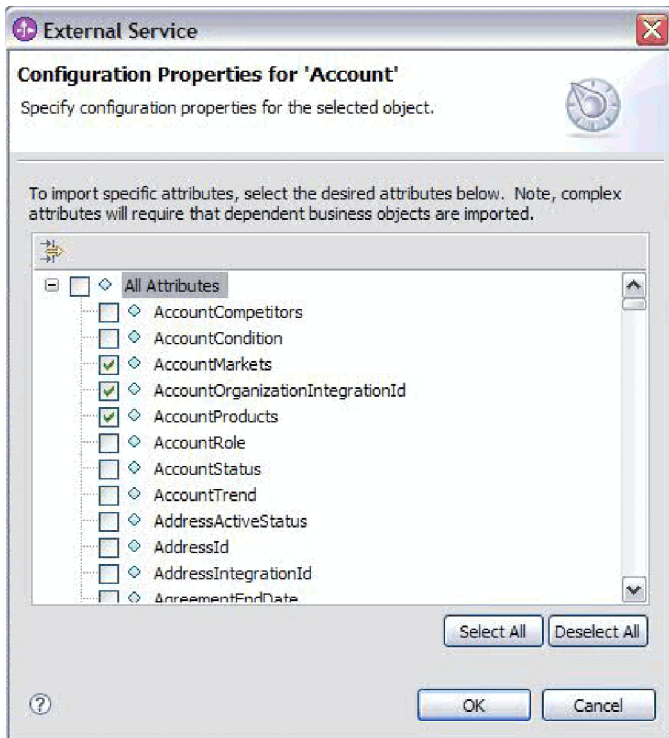


Figure 11. Configuration Properties window for selecting and importing the attributes to the Selected Objects pane

- h. If you do not select an attribute, an empty business object is generated. Also, if you add a primary business component, it is displayed in the **Selected objects** pane. If you add a dependent business component the primary business component is also added but it is not displayed in the **Selected objects** pane. You can select more than one object at a time.
- i. Click **Next**.

#### Selecting business services

1. In the Discovery Configuration window, select **Siebel Business Services** from the **Type of Siebel Objects to discover** list. Click **Next**.
2. In the Object Discovery and Selection window,
  - a. Click **Edit Query**.
  - b. In the Query Properties window, type the name of the business service 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 with which you want to work (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 Configuration Properties window is displayed.

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

**Note:**

- 1) When a business service has only 1 complex type Input/Output argument, the Configuration Properties 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: 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 so you do not need to 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 you do not need to select the **Integration Object** and it is not added as a business object.
- 6) Siebel Enterprise Information System (EIS) has restriction that it is not possible to change the existing input and output argument names for any method of a standard business service or create new argument names as they are hard coded internally in the underlying class of the business service along with the logic. So make sure that you are working on a custom Siebel business service if you want to use custom Siebel message.

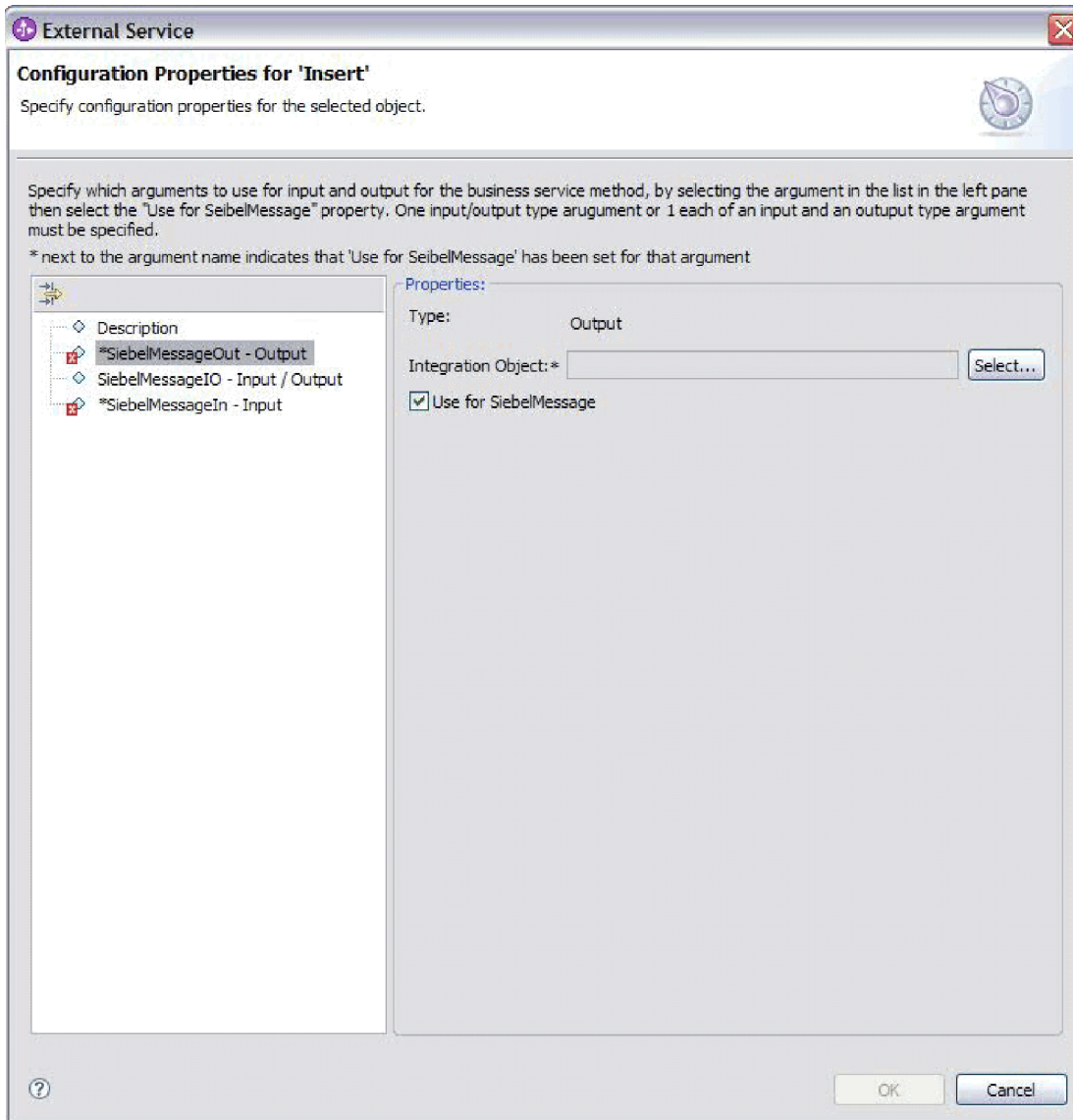


Figure 12. Specifying the integration value for the business methods

- i. Click Next.

### Results

You have selected the Siebel business object or business service you want to work with and named it.

### What to do next

Configure the objects you have selected.

#### Related concepts

“Business objects” on page 12

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 or an employee record, or unstructured text.

### Related reference

“Business object information” on page 115

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

## 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 should be stored.

### Before you begin

Make sure you have selected and imported the business function.

### About this task

To configure the business object, use the following procedure.

### Procedure

1. In the Configure Composite Properties window of the external service wizard, fill out the **Business object namespace field**. Use the default namespace (<http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel>) except in the following circumstance. 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/siebel1>



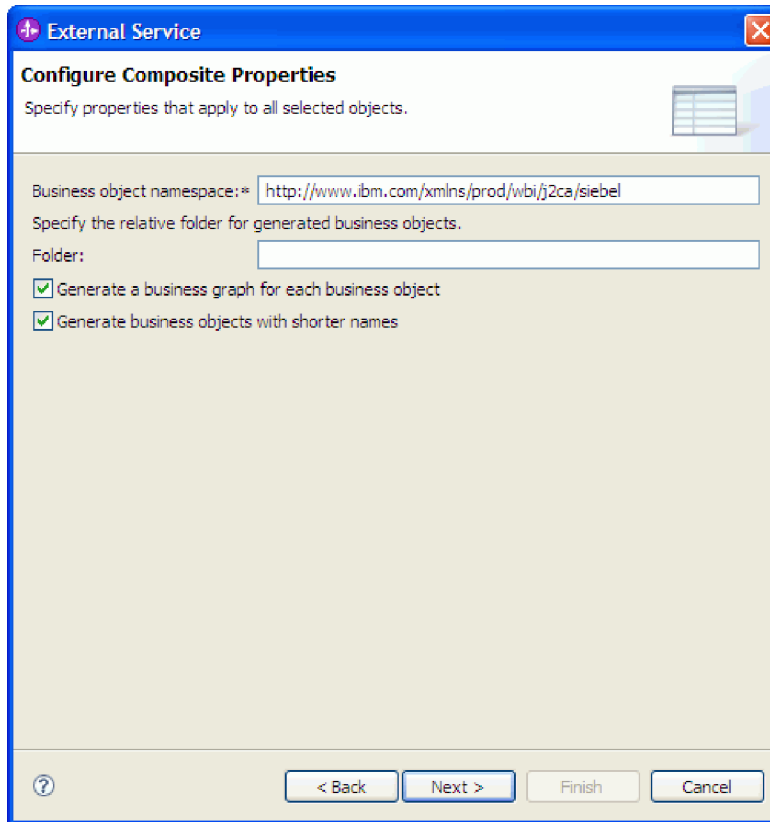


Figure 13. Configure Composite Properties window, showing the default Business object namespace

2. To indicate where the business object information should be stored, type the path to the location in the **Folder** field. This is an optional step.
3. If you want the business object or business objects to be enclosed within a business graph, leave **Generate a business graph for each business object** selected. Otherwise, remove the check.
4. If you want the business object or business objects to be generated with shorter names, select **Generate business objects with shorter names**.

## 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 Service Generation and Deployment Configuration window is displayed.

## What to do next

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

### Related concepts

“Business objects” on page 12

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 or an employee record, or unstructured text.

#### **Related reference**

“Business object information” on page 115

A business object is a structure that contains application-specific information (metadata) about how the adapter should 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**

To generate the module, which is the artifact that is deployed on WebSphere Process Server or WebSphere Enterprise Service Bus, you create the module, include the adapter in the module, and specify an alias used to authenticate the caller to the Siebel Business Applications.

### **Before you begin**

Make sure you have configured the business object. The Service Generation and Deployment Configuration window should be displayed.

### **About this task**

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

To generate the module, use the following procedure.

### **Procedure**

1. Optionally 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**.
2. In **Deploy connector project**, specify whether to include the adapter files in the module. Choose 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.
3. In the **J2C Authentication Data Entry** field, enter the name you specified in the Security section of the administrative console.
4. 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.

5. In the Connection properties section, set or change any connection properties that apply to your configuration. See “Managed connection factory properties” on page 134 for more information about these properties.  
Properties marked with an asterisk (\*) are required.

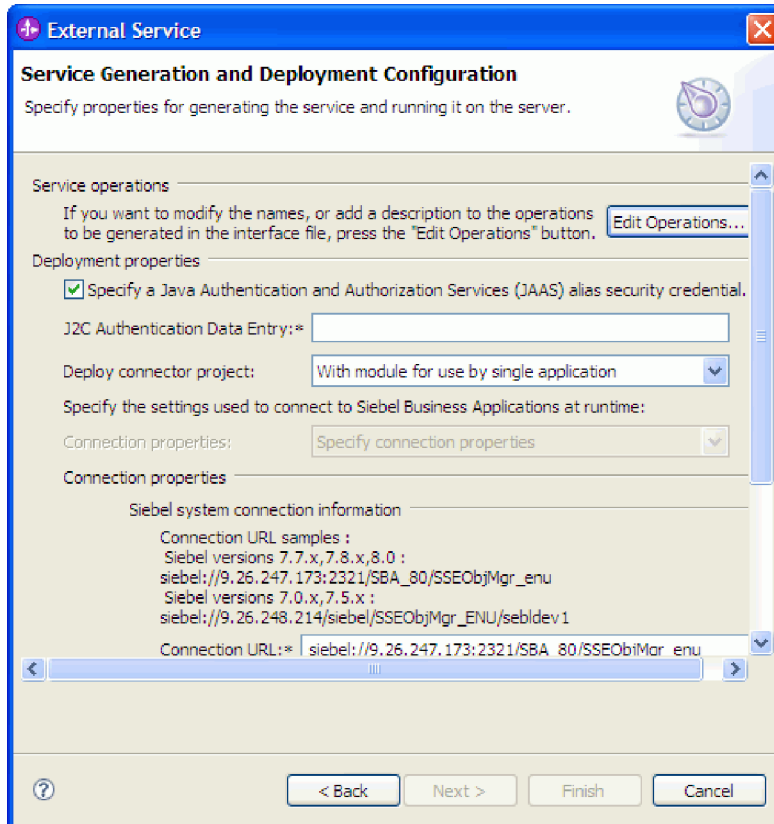


Figure 14. Service Generation and Deployment Configuration window, showing the **Connection properties** section

6. 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 Service Generation and Deployment Configuration window.

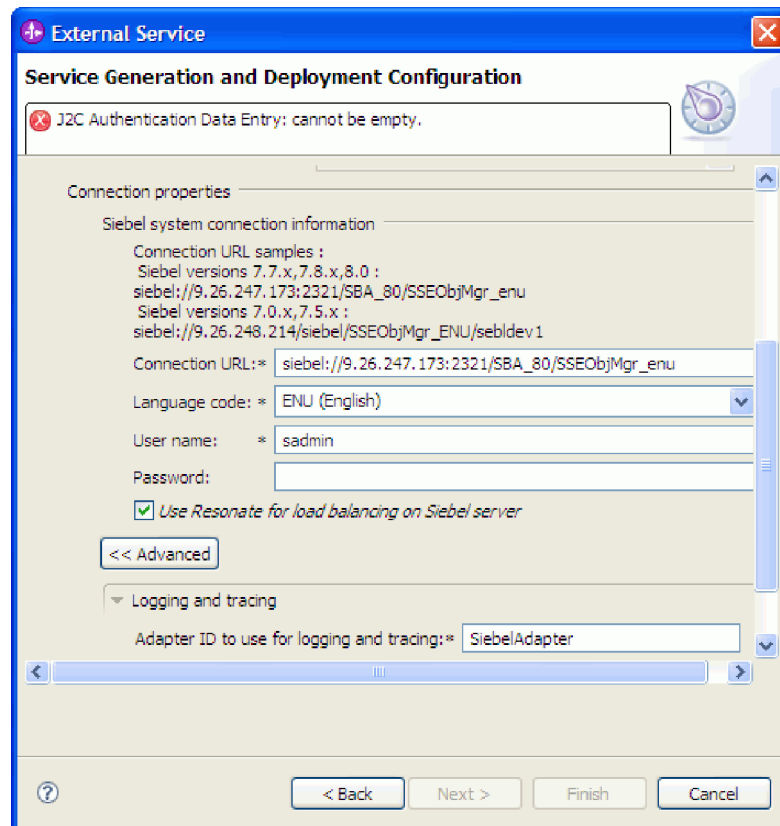


Figure 15. The Service Generation and Deployment Configuration window, showing the advanced property sections after clicking the **Advanced** button

See “Managed connection factory properties” on page 134 for more information about these optional properties.

- **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**.

7. Click **Next**. The Service Location Properties window opens.

8. Create a module.

- In the Service Location Properties window, click **New** in the **Module** field.
- In the Integration Project window, click **Create a module project** or **Create a mediation module project** and click **Next**.

9. In the New Module window, perform the following tasks:

- Type a name for the module.

As you type the name, it is added to the workplace specified in the **Directory** field.

This is the default location. If you want to specify a different location, remove the check from **Use default location** and type a new location or click **Browse** and select the location.

- b. Specify whether you want to open the module in the assembly diagram (for module projects) or whether you want to create a mediation flow component (for mediation module projects). By default, these choices are selected.
  - c. Click **Finish**.
10. In the Service Location Properties window, perform the following tasks:
  - a. If you want to change the default namespace, clear the **Use default namespace** check box and type a new path in the **Namespace** field.
  - b. Specify the folder within the module where the service description should be saved by typing a name in the **Folder** field or browsing for a folder. This is an optional step.

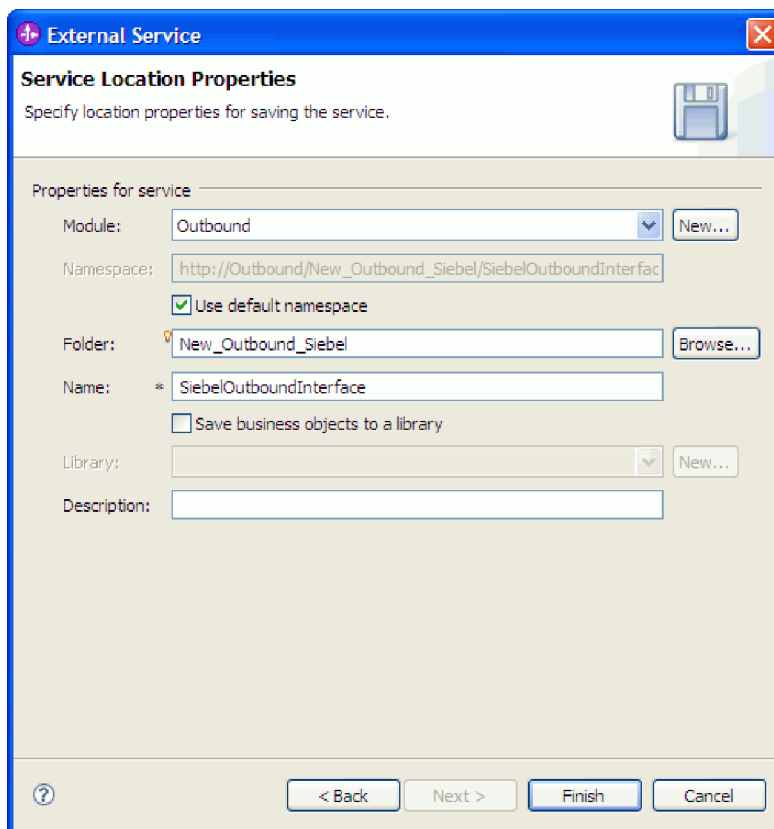


Figure 16. Service Location Properties window, showing Properties for service information

- c. If you want to save the business objects so that they can be used by another application, click **Save business objects to a library** and then select a library from the list or click **New** to create a new library.
  - d. Optionally type a description of the module.
11. 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 140

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 WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

“Outbound configuration properties” on page 121

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 WebSphere Process Server or WebSphere Enterprise Service Bus using WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

“Globalization” on page 161

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.

---

## Configuring the module for inbound processing

To configure a module to use the adapter for inbound processing, use the external service wizard in WebSphere Integration Developer 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. This means that the adapter polls the Siebel Business 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.

## Setting connection properties for the external service wizard

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

### Before you begin

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

### About this task

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

To specify the connection properties, use the following procedure.

### Procedure

1. On the Adapter Style window, select **Inbound** (to send data to the adapter from the Siebel server), and click **Next**.
2. On the Discovery Configuration window, specify the configuration properties:
  - a. In the **Connection URL** field, type the URL needed to connect to the Siebel server.
  - b. If necessary, change the default setting (ENU - English) for **Language code** by selecting another language from the drop down list.
  - c. Type the user name and password you use to access the Siebel server.  
The password is case-sensitive.

The screenshot shows the 'Discovery Configuration' window for an external service. It includes fields for connection URL, language code, user name, password, and settings for discovery on the Siebel system. The 'Type of Siebel objects to discover' is set to 'Siebel Business Services' and the 'Siebel repository name' is 'Siebel Repository'. There is an 'Advanced >>' button and navigation buttons at the bottom.

Figure 17. The Discovery Configuration window

- d. If necessary, change the default setting for **Type of Siebel objects to discover** (Siebel Business Services) by selecting Siebel Business Objects from the drop down list.
  - e. If necessary, change the default setting for **Siebel repository name** by typing in the name of the repository from which the objects are to be discovered.
  - f. If necessary, type a prefix in the **Prefix for business object names** field to have it attached to the front of the business object name.
3. To set additional advanced properties, click **Advanced**. When you select **Advanced**, the related properties are displayed.

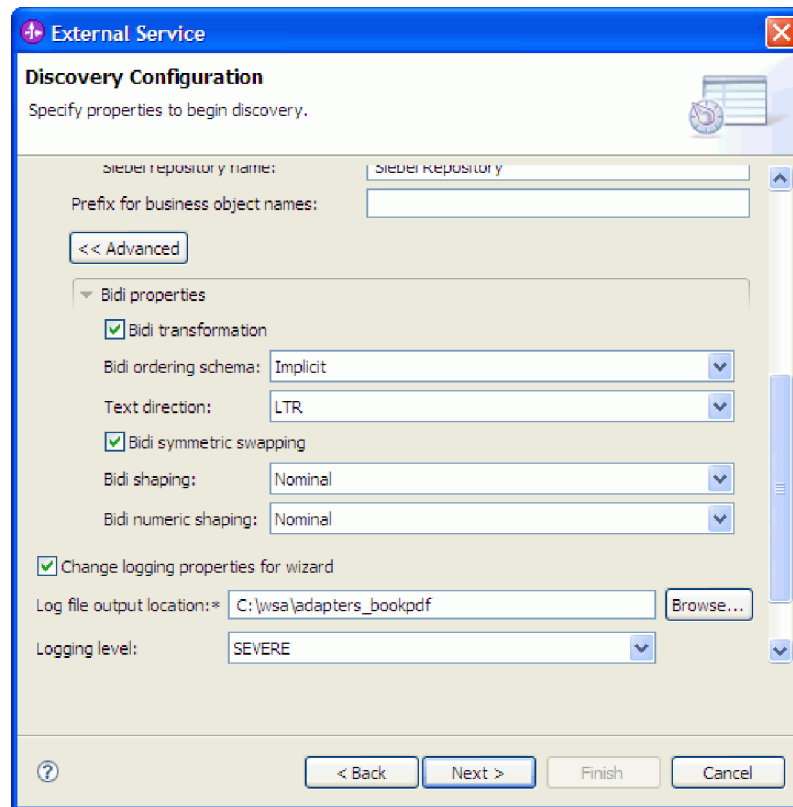


Figure 18. The Discovery Configuration window, showing the additional advanced property sections after clicking the **Advanced** button

4. If you need to set bidirectional properties, perform the following steps:
  - a. In the Bidi Properties section, select **Bidi transformation**.
  - b. Set properties for your environment. See “Connection properties for the wizard” on page 123 for more information about these properties.
5. To set logging properties for the external service wizard, perform the following steps:
  - a. Select **Specify the level of the logging desired**.
  - b. Change the location of the log file output location by clicking **Browse** and selecting a different location.
  - c. Set the **Logging Level**.

In a test environment, select **FINEST**, which provides the highest level of logging. In a production environment, select a level lower than **FINEST** to optimize the logging process.

**Note:** This log pertains to the external service wizard only, not to the operation of the adapter.

6. Click **Next**.

## Results

The external service wizard contacts the Siebel application, using the information you provided (such as Connection URL, user name and password) to log in. You see the Object Discovery and Selection window.

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

#### **Related reference**

“Connection properties for the wizard” on page 123

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

## **Selecting business objects and services**

Select business objects and services through the external service wizard to specify the business functions that you want to call and business objects 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 for and select business objects or business functions, use the following procedure.

#### **Selecting business objects**

1. In the Discovery Configuration window, select **Siebel Business Objects** as the **Type of Siebel Objects to discover**. The check box **Prompt for additional configuration settings when adding a business object** is displayed.
2. Select the check box **Prompt for additional configuration settings when adding a business object**. You need to select this if you want to select the attributes of the business objects.
3. Click **Next**.
4. In the Object Discovery and Selection window,
  - a. Click **Edit Query**.
  - b. In 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 Object Discovery and Selection window.



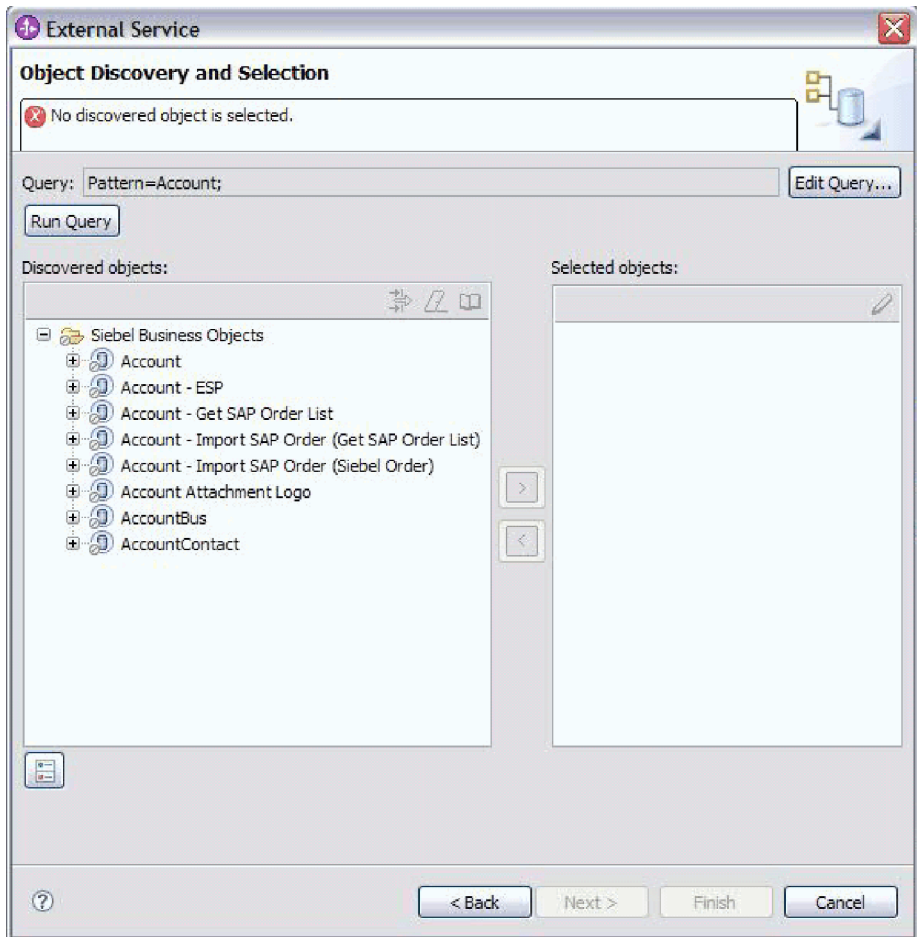


Figure 19. Object Discovery and Selection window, showing the Siebel Business Objects folder expanded with the list of discovered objects

- e. Expand the required business objects (for example, Account), to see the business components that you want to work with.

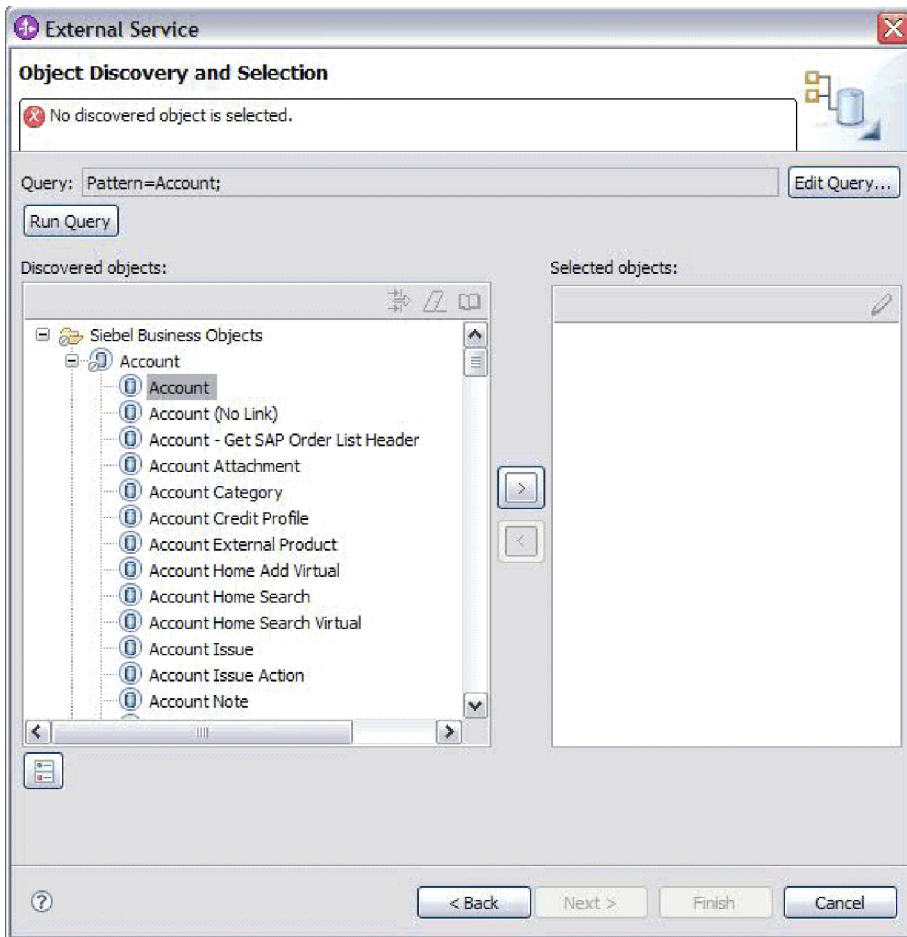


Figure 20. Object Discovery and Selection window, showing the Siebel Business Objects folder expanded with the list of discovered objects.

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

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

- g. Select the attributes you want to generate from the Configuration Properties window.

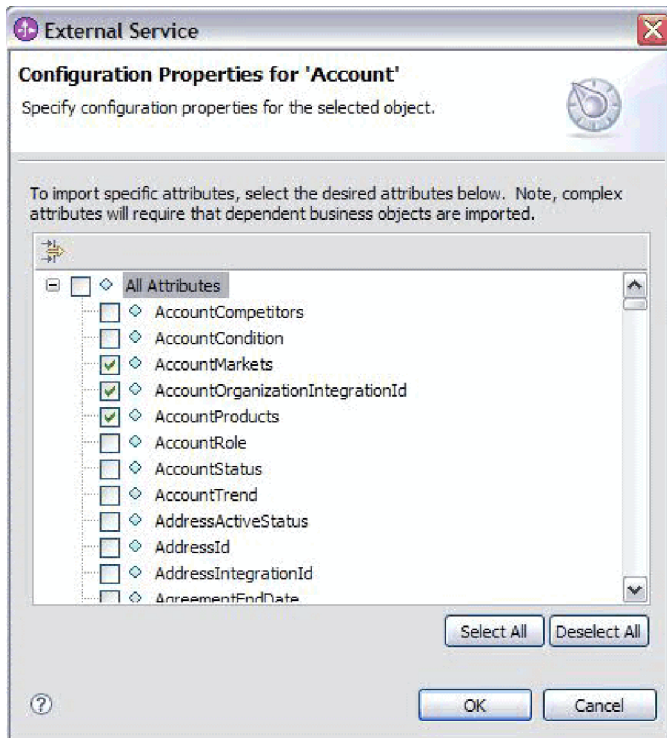


Figure 21. Configuration Properties window for selecting and importing the attributes to the Selected Objects pane

- h. If you do not select an attribute, an empty business object is generated. Also, if you add a primary business component, it is displayed in the **Selected objects** pane. If you add a dependent business component the primary business component is also added but it is not displayed in the **Selected objects** pane. You can select more than one object at a time.

**Note:** If you do not select the required attributes, like primary key of the business object, you will get errors at runtime.

- i. Click **Next**.

### Selecting business services

1. In the Discovery Configuration window, select **Siebel Business Services** from the **Type of Siebel Objects to discover** list. Click **Next**.
2. In the Object Discovery and Selection window,
  - a. Click **Edit Query**.
  - b. In the Query Properties window, type the name of the business service 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 with which you want to work (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 Configuration Properties window is displayed.

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

**Note:**

- 1) When a business service has only 1 complex type Input/Output argument, the Configuration Properties 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: 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 so you do not need to 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 you do not need to select the **Integration Object** and it is not added as a business object.
- 6) Siebel Enterprise Information System (EIS) has restriction that it is not possible to change the existing input and output argument names for any method of a standard business service or create new argument names as they are hard coded internally in the underlying class of the business service along with the logic. So make sure that you are working on a custom Siebel business service if you want to use custom Siebel message.

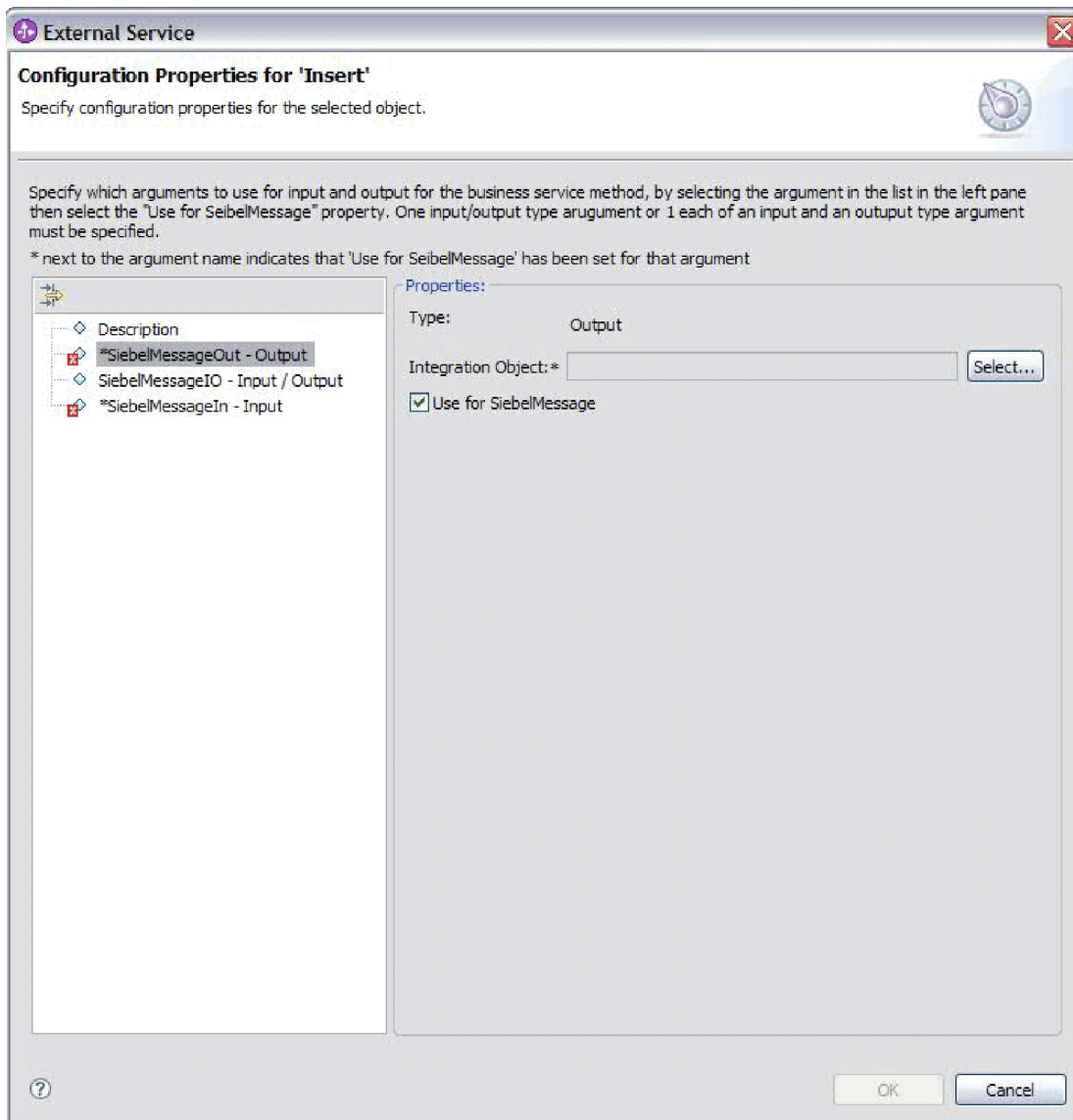


Figure 22. Specifying the integration value for the business methods

- i. Click Next.

### Results

You have selected the Siebel business object or business service you want to work with and selected a name for it.

### What to do next

From the Configure Objects 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 will 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 12

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 or an employee record, or unstructured text.

#### **Related reference**

“Business object information” on page 115

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

## **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 should be stored.

### **Before you begin**

Make sure you have selected and imported the business function.

### **About this task**

To configure the business object, use the following procedure.

### **Procedure**

1. In the Configure Composite Properties window of the external service wizard, fill out the **Business object namespace field**. Use the default namespace (<http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel>) except in the following circumstance. 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/siebel1>

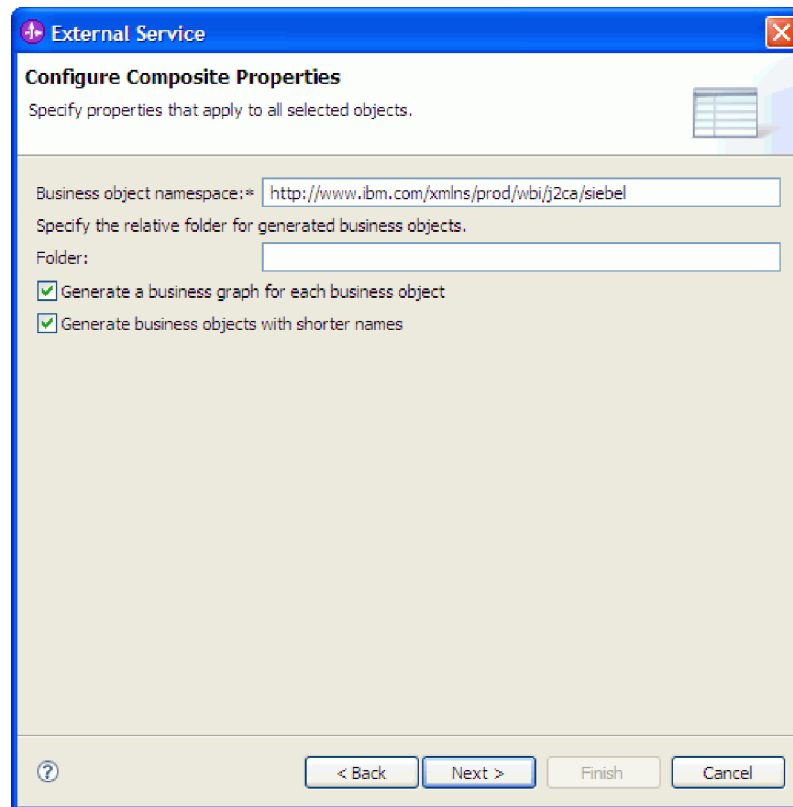


Figure 23. Configure Composite Properties window, showing the default Business object namespace

2. To indicate where the business object information should be stored, type the path to the location in the **Folder** field. This is an optional step.
3. If you want the business object or business objects to be enclosed within a business graph, leave **Generate a business graph for each business object** selected. Otherwise, remove the check.
4. If you want the business object or business objects to be generated with shorter names, select **Generate business objects with shorter names**.

## 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 Service Generation and Deployment Configuration window is displayed.

## What to do next

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

### Related concepts

“Business objects” on page 12

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 or an employee record, or unstructured text.



### Related reference

“Business object information” on page 115

A business object is a structure that contains application-specific information (metadata) about how the adapter should 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

To generate the module, which is the artifact that is deployed on WebSphere Process Server or WebSphere Enterprise Service Bus, you create the module, include the adapter in the module, and specify an alias used to authenticate the caller to the Siebel Business Applications.

### Before you begin

Make sure you have configured the business object. The Service Generation and Deployment Configuration window should be displayed.

### About this task

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

To generate the module, use the following procedure.

### Procedure

1. Optionally 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**.
2. In **Deploy connector project**, specify whether to include the adapter files in the module. Choose 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.
3. In the **J2C Authentication Data Entry** field, enter the name you specified in the Security section of the administrative console.
4. 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.



5. In the Connection properties section, set or change any connection properties that apply to your configuration. See “Activation specification properties” on page 153 and “Connection properties for the wizard” on page 123 for more information about these properties.

Properties marked with an asterisk (\*) are required.

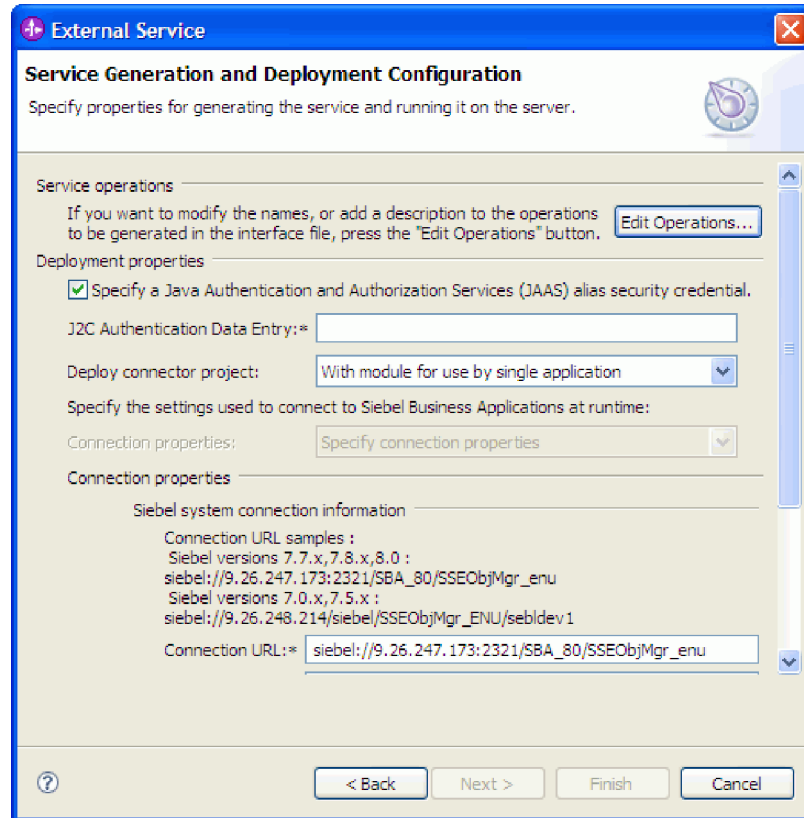


Figure 24. Service Generation and Deployment Configuration window, showing the **Connection properties** section

6. Optionally, 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
  - Logging and tracing properties
  - Other properties

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

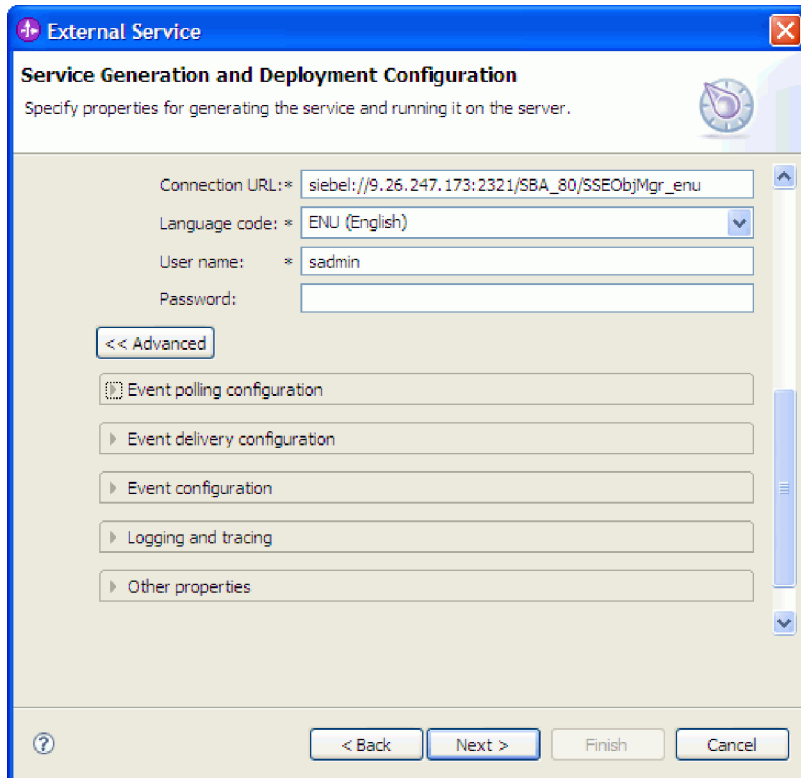


Figure 25. The Service Generation and Deployment Configuration window, showing the advanced property sections after clicking the **Advanced** button

**Note:** See “Activation specification properties” on page 153 and “Connection properties for the wizard” on page 123 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 157.
- 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 158.
- 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 160.
- 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 159.
- 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 161.
- 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 160.

- **Event delivery configuration**

- a. In **Type of delivery**, select the delivery method. The methods are described in “Type of delivery (DeliveryType)” on page 155.
- b. If you want to ensure that events are delivered only once and to only one export, select **Ensure once-only delivery**. This option might reduce performance but does not result in duplicate or missing event delivery. For more information, see “Ensure assured-only event delivery (AssuredOnceDelivery)” on page 156.

- c. 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 156.

- d. 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 159 and “Maximum connections (MaximumConnections)” on page 158.

- **Event configuration**

In **Siebel business object name for event store**, 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 127.

- **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**

- a. 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 125.
- b. 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 129.

7. Click **Next**. The Service Location Properties window opens.

8. Create a module.

- a. In the Service Location Properties window, click **New** in the **Module** field.
- b. In the Integration Project window, click **Create a module project** or **Create a mediation module project** and click **Next**.

9. In the New Module window, perform the following tasks:
  - a. Type a name for the module.  
As you type the name, it is added to the workplace specified in the **Directory** field.  
This is the default location. If you want to specify a different location, remove the check from **Use default location** and type a new location or click **Browse** and select the location.
  - b. Specify whether you want to open the module in the assembly diagram (for module projects) or whether you want to create a mediation flow component (for mediation module projects). By default, these choices are selected.
  - c. Click **Finish**.
10. In the Service Location Properties window, perform the following tasks:
  - a. If you want to change the default namespace, clear the **Use default namespace** check box and type a new path in the **Namespace** field.
  - b. Specify the folder within the module where the service description should be saved by typing a name in the **Folder** field or browsing for a folder.  
This is an optional step.

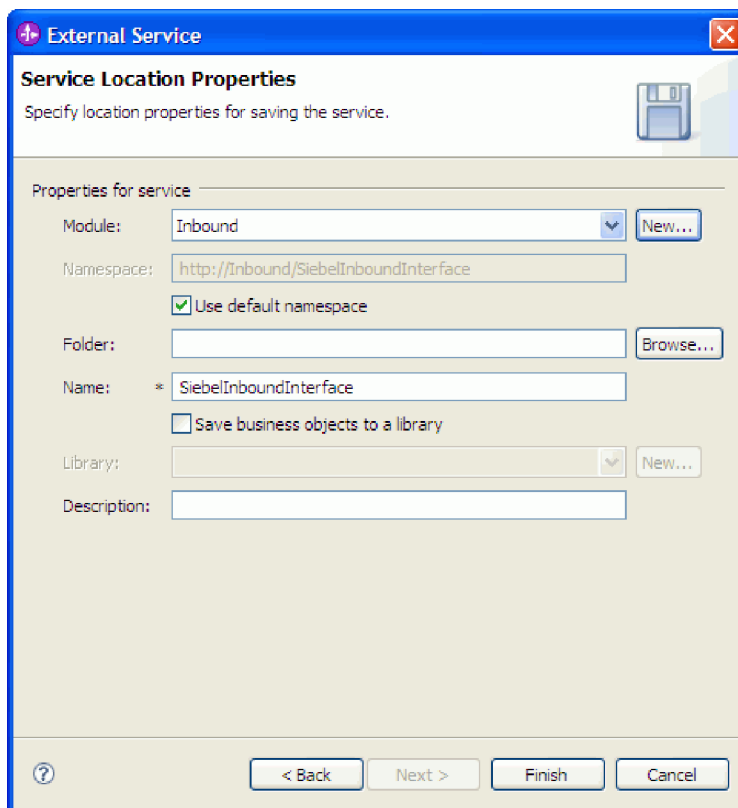


Figure 26. Service Location Properties window, showing Properties for service information

- c. If you want to save the business objects so that they can be used by another application, click **Save business objects to a library** and then select a library from the list or click **New** to create a new library.
  - d. Optionally type a description of the module.
11. 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 140

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 WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

“Outbound configuration properties” on page 121

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 WebSphere Process Server or WebSphere Enterprise Service Bus using WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

“Globalization” on page 161

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. Changing interaction specification properties using the assembly editor

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

### 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 will 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 WebSphere Integration Developer, expand the module name.
2. Expand **Assembly Diagram** and double-click the interface.
3. Click the interface in the assembly editor. (It shows the module properties if you don't do the extra click.)
4. Click the **Properties** tab. (You can also right-click the interface in the 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 139

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 WebSphere Integration Developer. The properties reside in the method binding of the import.



---

## Chapter 6. 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 WebSphere Integration Developer, the integrated test environment features runtime support for WebSphere Process Server 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 WebSphere Integration Developer, 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 WebSphere Process Server or WebSphere Enterprise Service Bus as EAR files using the administrative console or command-line tools.

---

### Deploying the module for testing

In WebSphere Integration Developer, 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.

#### About this task

### 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 WebSphere Process Server 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 WebSphere Process Server 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, and 8.0)
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 WebSphere Integration Developer, copy the files to the `${WAS_INSTALL_ROOT}/runtimes/bi_v62/lib/ext` directory.
  - In a production environment, copy the files to the `${WAS_INSTALL_ROOT}/lib/ext` directory of WebSphere Process Server 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 third party 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 WebSphere Integration Developer. 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 WebSphere Integration Developer, 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. This draws a wire from the export component to the new component, as shown in the following figure:
  - 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**.

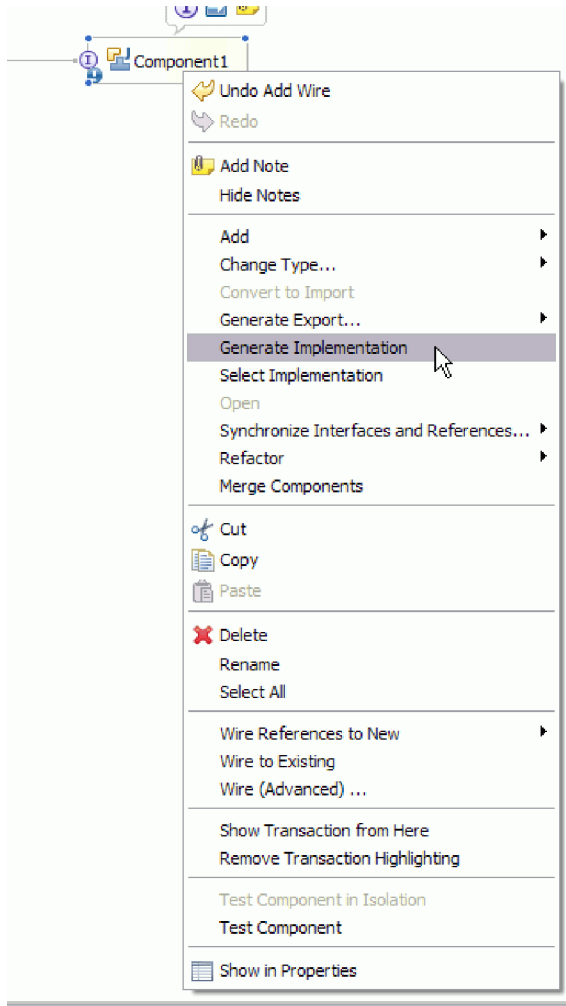


Figure 27. Generating a Java implementation

- 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 WebSphere Integration Developer, 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, you need to generate and wire a *target component* to which the adapter will send 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 server's settings.
  - d. Click **Finish** to publish the server.
2. Add the module to the server
  - a. Switch to the servers view. In WebSphere Integration Developer, select **Windows** → **Show View** → **Servers**
  - a. Start the server. In the Servers tab in the lower-right pane of the WebSphere Integration Developer screen, right-click on the server, and then select **Start**.
3. When the server status is *Started*, right-click on 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 WebSphere Integration Developer integration test client.

### Before you begin

You need to add the module to the server first.

### About this task

Testing a module 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.

### Procedure

1. Select the module you want to test, right-click on it, and select **Test** → **Test Module**.
2. For information on testing a module using the test client, see the *Testing modules and components* topic in the WebSphere Integration Developer 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 WebSphere Process Server or WebSphere Enterprise Service Bus in a production environment is a two-step process. First, you export the module in WebSphere Integration Developer as an enterprise archive (EAR) file. Second, you deploy the EAR file using the WebSphere Process Server or WebSphere Enterprise Service Bus administrative console.

### About this task

## Adding external software dependencies on the server

The adapter needs the Siebel JAR files installed on the WebSphere Process Server 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 WebSphere Process Server 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 8. 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, and 8.0)
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 WebSphere Integration Developer, copy the files to the `#{WAS_INSTALL_ROOT}/runtimes/bi_v62/lib/ext` directory.
  - In a production environment, copy the files to the `#{WAS_INSTALL_ROOT}/lib/ext` directory of WebSphere Process Server 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

will 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 have set **Deploy connector project** to **On server for use by multiple adapters** in the Service Generation and Deployment Configuration 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 runtime.

### Procedure

1. Start the administrative console.
2. Click **Resources** → **Resource Adapters** → **Resource adapters**.
3. In the Resource adapters page, click **Install RAR**.

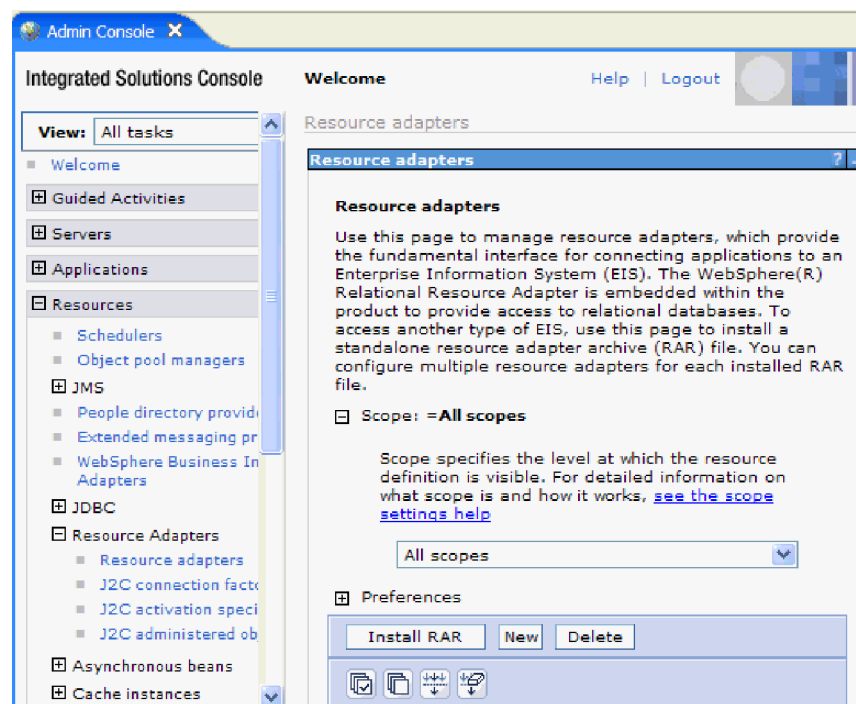


Figure 28. The Install RAR button on the Resource adapters page

4. 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:

`WID_installation_directory/ResourceAdapters/adapter_name/deploy/adapter.rar`

5. Click **Next**.
6. In the Resource adapters page, optionally change the name of the adapter and add a description.
7. Click **OK**.
8. 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 WebSphere Integration Developer, 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 WebSphere Process Server 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 WebSphere Integration Developer 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. Optionally, if you want to export the source files, select **Export source files**. 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 an existing file**.
8. Click **Finish**.

#### Results

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

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



## About this task

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

## Procedure

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



Figure 29. Preparing for the application installation window

3. Click **Browse** to locate your EAR file and click **Next**. The EAR file name is the name of the module followed by "App."
4. Optional: If you are deploying to a clustered environment, complete the following steps.
  - a. On the **Step 2: Mapping modules to servers** window, select the module.
  - b. Select the name of the server cluster.
  - c. Click **Apply**.
5. Click **Next** to open the Summary. Verify that all settings are correct and click **Finish**.
6. Optional: If you are using an authentication alias, complete the following steps:
  - a. Expand **Security** and select **Business Integration Authentication Aliases**.
  - b. Select the authentication alias that you want to configure. You must have administrator or operator authority to make changes to authentication alias configurations.
  - c. Optional: If it is not already filled in, type the **User name**.
  - d. If it is not already filled in, type the **Password**.
  - e. If it is not already filled in, 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 7. 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 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).

#### Related reference

“Inbound configuration properties” on page 140

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 WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

“Outbound configuration properties” on page 121

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 WebSphere Process Server or WebSphere Enterprise Service Bus using WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

### 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 WebSphere Process Server or WebSphere Enterprise Service Bus.

#### About this task

Custom properties are default configuration properties shared by all WebSphere adapters.

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

#### Procedure

1. Start the administrative console.
2. Under **Applications**, select **Enterprise Applications**.
3. From the **Enterprise Applications** list, click the name of the adapter module whose properties you want to change. The Configuration page is displayed.

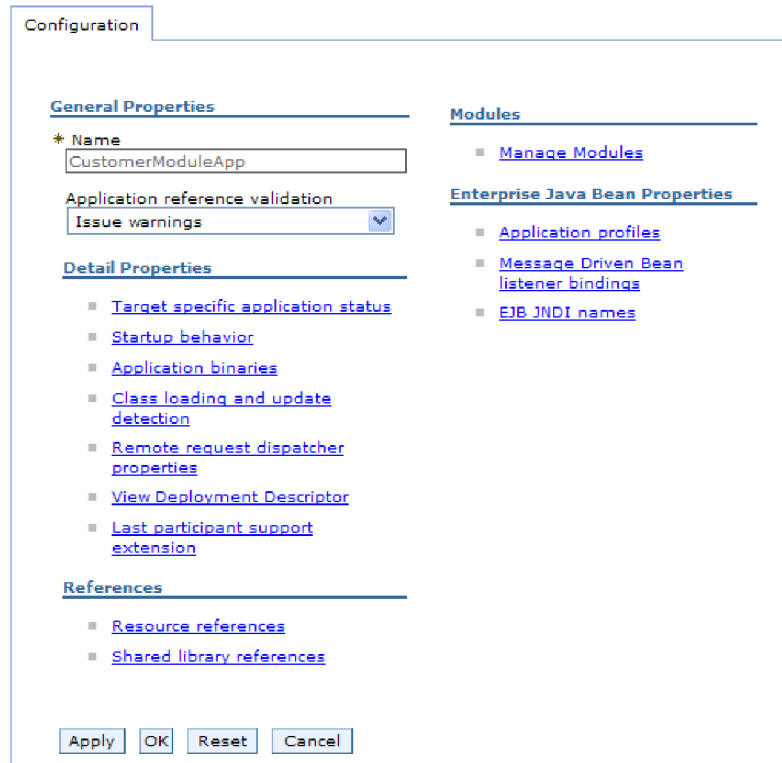


Figure 30. The Manage Modules selection in the Configuration tab

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

**Note:** See “Resource adapter properties” on page 129 for more information about these properties.

- a. Click the name of the property. The **Configuration** page for the selected property is displayed. For example, if you click the **logNumberOfFiles** property, you see the page shown in the figure below.

The screenshot shows a configuration dialog box titled "Configuration" with a tab labeled "Configuration". Under the "General Properties" section, there are several fields and controls:

- \* Scope:** A text box containing "widNode".
- Required:** An unchecked checkbox.
- Name:** A text box containing "logNumberOfFiles".
- Value:** A text box containing "1".
- Description:** An empty text area with scroll bars.
- Type:** A dropdown menu showing "java.lang.String".

At the bottom of the dialog, there are four buttons: "Apply", "OK", "Reset", and "Cancel".

Figure 31. The Configuration tab for the logNumberOfFiles property

- b. Change the contents of the **Value** field or type a value, if the field is empty. You can change the number in the **Value** field and add a description of the property.
  - c. Click **OK**.
9. Click the **Save** link in the **Messages** box at the top of the window.

### Results

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

#### Related reference

“Resource adapter properties” on page 129

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

## 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 WebSphere Process Server 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. Start the administrative console.
2. Under **Applications**, select **Enterprise Applications**.
3. In the **Enterprise Applications** list, click the name of the adapter module whose properties you want to change.
4. Under **Modules**, click **Manage Modules**.

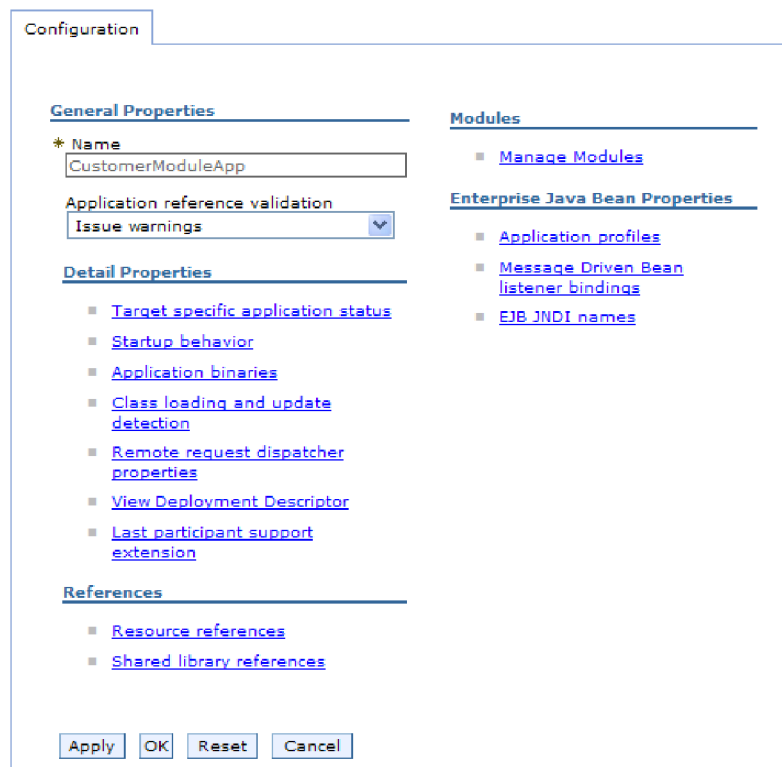


Figure 32. The Manage Modules selection in the Configuration tab

5. Click **IBM WebSphere Adapter for Siebel Business Applications**.
6. In the **Additional Properties** list, click **Resource Adapter**.
7. On the next page, from the **Additional Properties** list, click **J2C connection factories**.
8. Click the name of the connection factory associated with your adapter module.
9. In the **Additional Properties** list, click **Custom properties**.

Custom properties are those J2C connection factory properties that are unique to 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 134 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. Click the **Save** link in the **Messages** box at the top of the window.

## Results

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

### Related reference

“Managed connection factory properties” on page 134

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 WebSphere Process Server 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. Start the administrative console.
2. Under **Applications**, select **Enterprise Applications**.
3. From the **Enterprise Applications** list, click the name of the adapter module whose properties you want to change.
4. Under **Modules**, click **Manage Modules**.

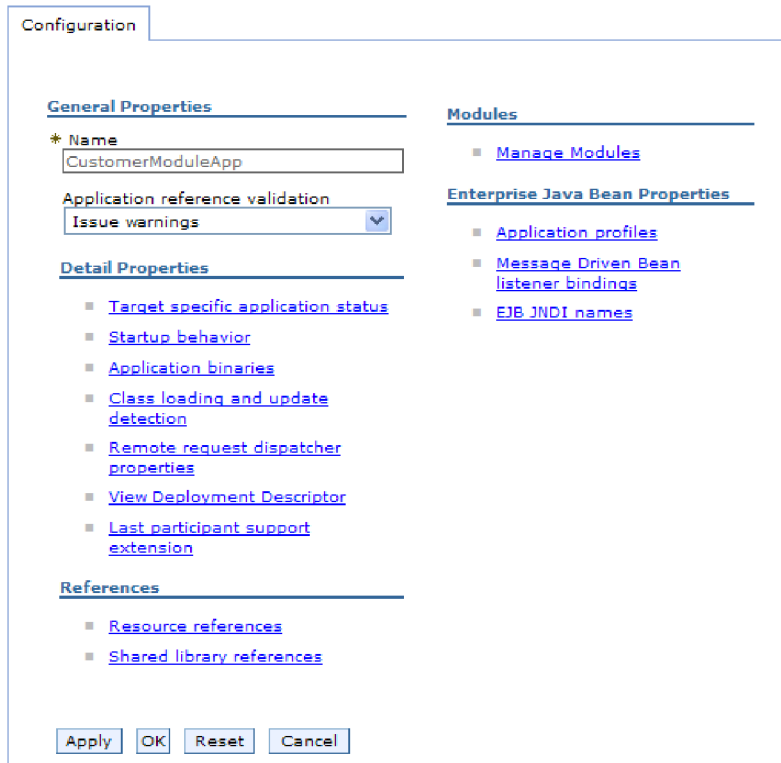


Figure 33. The Manage Modules selection in the Configuration tab

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

**Note:** See "Activation specification properties" on page 153 for more information about these properties.

**Note:** See "Activation specification properties" on page 153 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. Click the **Save** link in the **Messages** box at the top of the window.

## Results

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

### Related reference



“Activation specification properties” on page 153

Activation specification properties hold the inbound event processing configuration information for a message endpoint. You can set 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, you use the administrative console of the runtime environment. You provide general information about the adapter and then set resource adapter properties (which are used for general adapter operation). If the adapter will be used for outbound operations, you create a connection factory and then set properties for it. If the adapter will be used for inbound operations, you create an activation specification and then set 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 WebSphere Process Server 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 WebSphere Process Server or WebSphere Enterprise Service Bus.

#### About this task

Custom properties are default configuration properties shared by all WebSphere adapters.

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

#### Procedure

1. Start the administrative console.
2. Click **Resources** → **Resource Adapters** → **Resource adapters**.
3. In the Resource adapters page, click **IBM WebSphere Adapter for Siebel Business Applications**.
4. In the **Additional Properties** list, click **Custom properties**.
5. For each property you want to change, perform the following steps.

**Note:** See “Resource adapter properties” on page 129 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.

For example, if you click **logNumberOfFiles**, you see the following page:

The image shows a configuration dialog box titled "Configuration" with a tab labeled "Configuration". Inside, there is a section titled "General Properties". The form contains the following fields and controls:

- \* Scope:** A text input field containing "widNode".
- Required:** An unchecked checkbox.
- Name:** A text input field containing "logNumberOfFiles".
- Value:** A text input field containing "1".
- Description:** A text area with a vertical scrollbar, currently empty.
- Type:** A dropdown menu showing "java.lang.String".
- Buttons:** "Apply", "OK", "Reset", and "Cancel" buttons are located at the bottom.

Figure 34. The Configuration tab for the logNumberOfFiles property

You can change the number in the **Value** field and add a description of the property.

- c. Click **OK**.
6. Click **Save** in the **Messages** box at the top of the page.

### Results

The resource adapter properties associated with your adapter are changed.

#### Related reference

“Resource adapter properties” on page 129

The resource adapter properties control the general operation of the adapter.

You set the resource adapter properties using the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

## 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 WebSphere Process Server 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 WebSphere Process Server 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. Start the administrative console.
2. Click **Resources** → **Resource Adapters** → **Resource adapters**.
3. In the Resource adapters page, click **IBM WebSphere Adapter for Siebel Business Applications**.
4. In the **Additional Properties** list, click **J2C connection factories**.
5. If you are going to use an existing connection factory, skip ahead to select from the list of existing connection factories.

**Note:** If you selected **Use predefined connection properties** when you used 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 could type AdapterCF.
- c. Type a value for **JNDI name**. For example, you could type com/eis/AdapterCF.
- d. Select an authentication alias from the **Component-managed authentication alias** list.
- e. Click **OK**.
- f. Click **Save** in the **Messages** box at the top of the page.

The newly created connection factory is displayed.

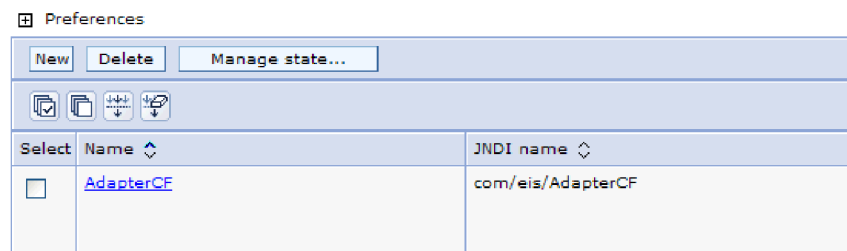


Figure 35. User defined connection factories for use with the resource adapter

6. In the list of connection factories, click the one you want to use.
7. In the **Additional Properties** list, click **Custom properties**.  
Custom properties are those J2C connection factory properties that are unique to Adapter for Siebel Business Applications. Connection pool and advanced connection factory properties are properties you configure if you are developing your own adapter.
8. For each property you want to change, perform the following steps.

**Note:** See “Managed connection factory properties” on page 134 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**.
9. After you have finished setting properties, click **Apply**.
  10. Click **Save** in the **Messages** box at the top of the window.

## Results

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

### Related reference

“Managed connection factory properties” on page 134

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 WebSphere Process Server 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 WebSphere Process Server 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. Start the administrative console.
2. Click **Resources** → **Resource Adapters** → **Resource adapters**.
3. In the Resource adapters page, click **IBM WebSphere Adapter for Siebel Business Applications**.
4. In the **Additional Properties** list, click **J2C activation specifications**.
5. If you are going to use an existing activation specification, skip ahead to select from an existing list of activation specifications.

**Note:** If you selected **Use predefined connection properties** when you used the external service wizard to configure the adapter module, you do not need to 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 could type AdapterAS.
  - c. Type a value for **JNDI name**. For example, you could type com/eis/AdapterAS.
  - d. 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.
6. In the list of activation specifications, click the one you want to use.
  7. In the Additional Properties list, click **J2C activation specification custom properties**.
  8. For each property you want to set, perform the following steps.

**Note:** See “Activation specification properties” on page 153 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**.
9. After you have finished setting properties, click **Apply**.
  10. Click **Save** in the **Messages** box at the top of the page.

## Results

The activation specification properties associated with your adapter are set.

### Related reference

“Activation specification properties” on page 153

Activation specification properties hold the inbound event processing configuration information for a message endpoint. You can set 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. On the administrative console, click **Applications** → **Enterprise Applications**.

**Note:** The administrative console is labeled “Integrated Solutions Console”.

2. Select the check box of the application that you want to start. The application name is the name of the EAR file you installed, without the .EAR file extension.

3. 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. On the administrative console, click **Applications** → **Enterprise Applications**.

**Note:** The administrative console is labeled “Integrated Solutions Console”.

2. Select the check box of the application that you want to stop. The application name is the name of the EAR file you installed, without the .EAR file extension.
3. Click **Stop**.

### Results

The status of the application changes to Stopped, and a message stating that the application has stopped displays 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 the 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 WebSphere Process Server.

### 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 can 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 WebSphere Application Server web site: <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 from which to gather 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, refer to “Enabling tracing with the Common Event Infrastructure (CEI)” on page 100.

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.

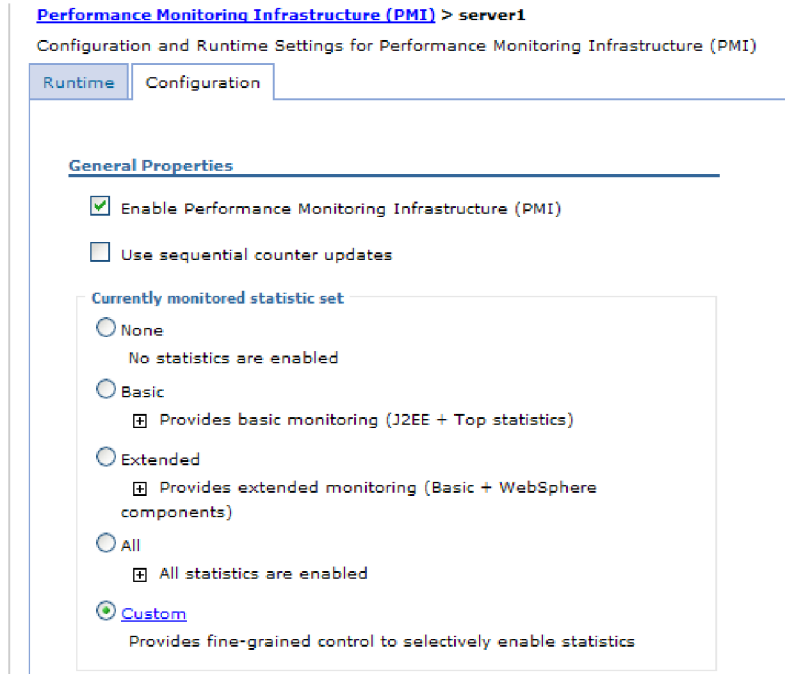


Figure 36. Enabling Performance Monitoring Infrastructure

- 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.



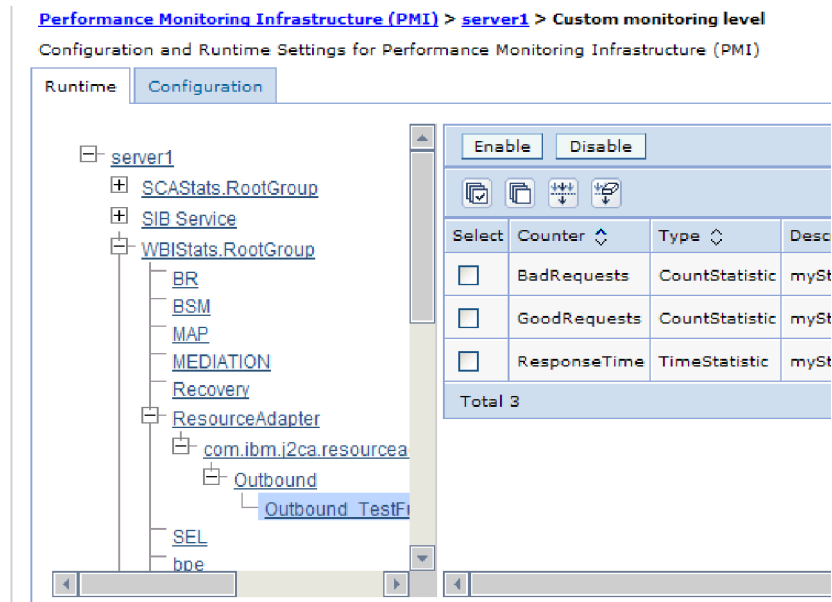


Figure 37. Runtime tab used for configuring PMI

- 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 WebSphere Process Server.

## Before you begin

Configure Performance Monitoring Infrastructure for your adapter.

## Procedure

1. In the administrative console, expand **Monitoring and Tuning**, expand **Performance Viewer**, then select **Current Activity**.
2. In the list of servers, click the name of your server.
3. Under your server name, expand **Performance Modules**.

4. Click **WBIStatsRootGroup**.
5. Click **ResourceAdapter** and the name of your adapter module.
6. If there is more than one process, select the check boxes for the processes whose statistics you want to view.

## Results

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

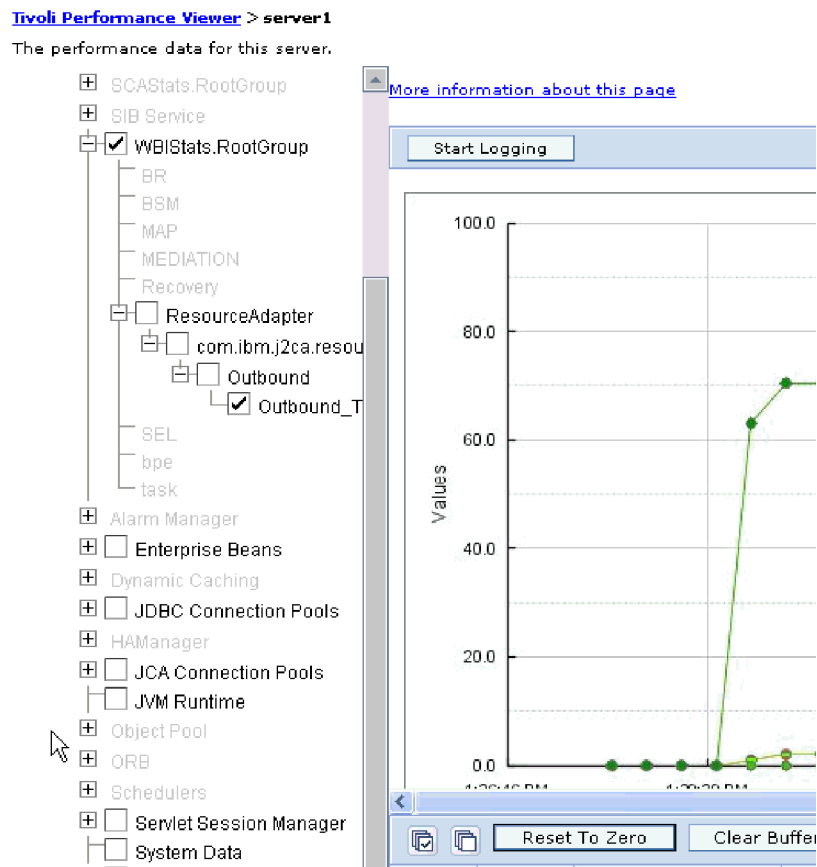


Figure 38. Adapter performance statistics, using graph view

## Enabling tracing with the Common Event Infrastructure (CEI)

The adapter can use the Common Event Infrastructure, 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

#### Procedure

1. In the administrative console, click **Troubleshooting**.
2. Click **Logs and Trace**.
3. In the list of servers, click the name of your server.

4. In the **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, choose one of the following:
    - 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.



---

## Chapter 8. Troubleshooting and support

Common troubleshooting techniques and self-help information help you identify and solve problems quickly.

### Related reference

“Adapter messages” on page 164

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

---

### Support for the 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, make sure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate 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 log and trace information to a particular instance of an adapter. To illustrate how the length of the adapter ID property affects the filtering of log and trace files, suppose 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. For example, suppose you set the adapter ID properties of two instances to Instance01 and Instance02. You will not be able to separately 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, to prevent 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 about the adapter ID property, see “Adapter ID (AdapterID)” on page 130. For more information about the Log and Trace Analyzer, see <http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/org.eclipse.hyades.log.ui.doc.user/concepts/cltaviews.htm>.

---

## 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.

### About this task

## Configuring logging properties

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

### About this task

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

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

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

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

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

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

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

### Procedure

1. In the navigation pane of the administrative console, click **Servers** → **Application Servers**.
2. Click the name of the server that you want to work with.
3. Under **Troubleshooting**, click **Logs and trace**.
4. Click **Change Log Detail Levels**.
5. Specify when you want the change to take effect:

- For a static change to the configuration, click the **Configuration** tab.
  - For a dynamic change to the configuration, click the **Runtime** tab.
6. 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 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 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 process server is written to the SystemOut.log and trace.log files, respectively.

### 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. This method allows you to make changes that take effect right away.

---

## 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 WebSphere Process Server 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 WebSphere Process Server or WebSphere Enterprise Service Bus documentation.

## 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 WebSphere Process Server and WebSphere Enterprise Service Bus support other types of faults, the adapter generates only business faults, which are called simply *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 39.

WBIFault	
message	string
errorCode	string
primaryKeySet	PrimaryKeyPairType []

Figure 39. The structure of the WBIFault business object

Some faults contain the matchCount attribute, to provide additional information about the error. For others, WBIFault contains all the information needed to handle the fault.

The wizard creates the following fault business objects:

- DuplicateRecordFault

When processing a Create operation, the adapter throws this fault if a duplicate record is found.

- MatchesExceededLimitFault

When processing a RetrieveAll operation, the adapter throws 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 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 throws this fault. This fault can occur for the Create, Delete, Update, Retrieve, ApplyChanges and Exists operations.

- RecordNotFoundFault

When retrieving data, the adapter throws this fault if the record is not found in the database for the keys specified. This fault can occur for the Delete, Update, Retrieve, and RetrieveAll operations.

## Configuring the module for fault processing

Before you can configure your module to support business faults, you must have used the external service wizard to configure your module.

To enable fault processing, you must modify the .import and WSDL files for your module. You can configure faults at either the binding level or the method level. If the changes are made at binding level, they apply to all methods in the import. If the changes are made at the method binding level, you can configure a different fault for each method.

Table 9 lists the fault name and fault binding class for each fault. Use the fault name and fault binding class when you configure the module.

Table 9. The fault name and fault binding class for each fault

Fault name	Associated fault binding class
DUPLICATE_RECORD	com.ibm.j2ca.extension.emd.runtime.WBIFaultDataBindingImpl
MATCHES_EXCEEDED_LIMIT	com.ibm.j2ca.extension.emd.runtime.MatchingFaultDataBinding
MISSING_DATA	com.ibm.j2ca.extension.emd.runtime.WBIFaultDataBindingImpl
RECORD_NOT_FOUND	com.ibm.j2ca.extension.emd.runtime.WBIFaultDataBindingImpl

1. Edit the .import file to configure the fault at either the binding or the method level.
  - To configure the faults at the binding level:
    - a. In the binding section, add the faultSelector attribute and the name of the fault selector. The name of the fault selector is com.ibm.j2ca.extension.emd.runtime.WBIFaultSelectorImpl.
    - b. For each fault that you want to enable, add a <faultBinding> element. In the element, specify the fault name and the fault data binding class name from Table 9.

The following .import file shows the RECORD\_NOT\_FOUND fault configured for all methods. **Bold face type** indicates changes made to enable fault handling.

```
<esbBinding xsi:type="eis:EISImportBinding"
  dataBindingType="com.ibm.j2ca.siebel.emd.runtime.SiebelDataBindingGenerator"
  faultSelector="com.ibm.j2ca.extension.emd.runtime.WBIFaultSelectorImpl">
  <resourceAdapter
    name="SiebelOutApp.IBM WebSphere Adapter for Siebel Business Applications"
    type="com.ibm.j2ca.siebel.SiebelResourceAdapter">
    <properties/>
  </resourceAdapter>
  <faultBinding fault="RECORD_NOT_FOUND"
    faultBindingType="com.ibm.j2ca.extension.emd.runtime.WBIFaultDataBindingImpl"/>
```

- To configure the faults at the method level:
  - a. In method binding section for the method you want to associate with the fault, add the name of the fault selector. The value for fault selector is com.ibm.j2ca.extension.emd.runtime.WBIFaultSelectorImpl.

- b. Add the fault binding elements in the method binding section. Use the fault name and the corresponding fault data binding class name from Table 9 on page 108.

The following .import file shows the RECORD\_NOT\_FOUND fault configured for the retrieveAccount method. **Bold face type** indicates changes made to enable fault handling.

```
<methodBinding
  inDataBindingType="com.ibm.xmlns.prod.wbi.j2ca.siebel.accountbg.AccountBGDataBinding"
  method="retrieveAccount"
  outDataBindingType="com.ibm.xmlns.prod.wbi.j2ca.siebel.accountbg.AccountBGDataBinding"
  faultSelector="com.ibm.j2ca.extension.emd.runtime.WBIFaultSelectorImpl">
  <interaction>
    <properties>
      <functionName>Retrieve</functionName>
    </properties>
  </interaction>
  <faultBinding fault="RECORD_NOT_FOUND"
  faultBindingType="com.ibm.j2ca.extension.emd.runtime.WBIFaultDataBindingImpl"/>
</methodBinding>
```

2. Determine the target namespaces for your faults. For each fault that you want to enable, determine the namespace as follows:

- a. Open the fault schema (XSD file) in a text editor.
- b. Locate the target namespace. The target namespace is shown in **bold face type** in the following portion of a fault schema:

```
<?xml version="1.0" encoding="UTF-8" ?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://com/ibm/j2ca/fault/afcfault"
  xmlns:basefault="http://com/ibm/j2ca/fault">
<import namespace="http://com/ibm/j2ca/fault" schemaLocation="WBIFault.xsd"/>
```

. . .

The faults can all have the same target namespace or they can have different target namespaces.

3. Edit the WSDL file to declare the faults for the service. A sample WSDL file with these changes made is shown at the end of the list.

- a. In the <definitions> element, add a namespace for each fault namespace, using the information you obtained from the fault schema files. If all your fault schemas have the same targetNamespace, add only one alias. If they have different targetNamespaces, add an alias for each unique namespace.
- b. Create an <xsd:import> element to import the schema for each fault you want to enable.
- c. Declare import statements for each fault type. Make sure that you are using the correct alias defined in step 3a to resolve the complex type in type=*alias:faultBOName.xsd*.
- d. Declare the message tags for each of the fault types.
- e. Add the fault declaration to each method where faults should be handled.

The following WSDL file defines the MULTIPLE\_MATCHING\_RECORDS and RECORD\_NOT\_FOUND faults. **Bold face type** indicates changes made to enable fault handling.

```
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:AccountBG="http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/accountbg"
  xmlns:intf="http://SiebelOut/SiebelOutboundInterface"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```

Step 3a on  
page 109

```
xmlns:fault="http://com/ibm/j2ca/fault/afcfault"
name="SiebelOutboundInterface.wsdl"
targetNamespace="http://SiebelOut/SiebelOutboundInterface">
<types>
  <xsd:schema
    xmlns:tns="http://SiebelOut/SiebelOutboundInterface"
    xmlns:xsd1="http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/accountbg"
    elementFormDefault="qualified"
    targetNamespace="http:// SiebelOut/SiebelOutboundInterface"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:import
      namespace="http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/accountbg"
      schemaLocation="AccountBG.xsd"/>
```

Step 3b on  
page 109

```
<xsd:import namespace="http://com/ibm/j2ca/fault/afcfault"
  schemaLocation="RecordNotFoundFault.xsd"/>
```

Step 3c on  
page 109

```
...
  <xsd:element name="recordNotFoundFaultX">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="recordNotFoundFaultElement"
          type="fault:RecordNotFoundFault"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
</types>
```

Step 3d on  
page 109

```
...
<message name="recordNotFoundFault">
  <part element="intf:recordNotFoundFaultX" name="recordNotFoundFaultPart"/>
</message>
<portType name="SiebelOutboundInterface">
  name="retrieveCUSTOMERResponse"/>
```

Step 3e on  
page 109

```
...
<fault message="intf:multipleMatchingRecordsFault"
  name="multipleMatchingRecordsFaultFault" />
<fault message="intf:recordNotFoundFault"
  name="recordNotFoundFaultFault" />
</operation>
</portType>
</definitions>
```

---

## Solutions to some common problems

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

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

## **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 will be 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 should 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.

## **The adapter generates multiple container attributes in the business object for simple link relationships**

### **Problem**

WebSphere Adapter for Siebel Business Applications generates multiple container attributes 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, if 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 instead of one: BusinessAddress, BusinessAddress', and BusinessAddress3. You can confirm that SimpleLink application-specific information is set against the three container attributes.

### **Solution and Workaround**

Although the extra container attributes should not affect the transaction if they are left untouched, you can remove the unwanted attributes by using the business object editor in WebSphere Integration Developer. Refer to the WebSphere Integration Developer Information Center for details about using the business object editor.

---

## XAResourceNotAvailableException

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

### Symptom:

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

```
com.ibm.ws.Transaction.XAResourceNotAvailableException
```

### Problem:

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

### Solution:

To correct this problem, use the following procedure:

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

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

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

```
server_install_directory\profiles\profile_name\tranlog\node_name\wps\  
server_name\transaction\tranlog
```

3. Start the process server.

---

## Self-help resources

Use the 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 Web site

The WebSphere Adapters software support Web site at <http://www.ibm.com/software/integration/wbiadapters/support/> provides links to many resources to help you learn about, use, and troubleshoot WebSphere Adapters, including the following types of

- Flashes (alerts about the product)
- Technical information including the product information center, manuals, IBM Redbooks®, and white papers
- Educational offerings
- Technotes

## Recommended fixes

A list of recommended fixes you should 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 the 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

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 9. 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 should 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 Business Integration perspective of WebSphere Integration Developer 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 10. 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 11. 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 12. 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 13. 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 objects and components. The following table describes the metadata elements located at the container level for Siebel business objects and Siebel business components.

Table 14. 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.
KeepRelations	Specifies whether or not the child objects needs 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.

Table 14. Container attribute-level metadata elements generated against Siebel business objects and Siebel business components (continued)

Metadata element	Description
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.

## Supported operations

An operation is the action that an adapter can perform on the Siebel 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 15. 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 children
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 for business objects representing Siebel business services, integration components, business objects, and business components

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 16. 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" <b>Note:</b> 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 Input/Output complex type arguments> <ul style="list-style-type: none"> <li>If there are no Input or Input/Output arguments, the names of all the output arguments are: &lt;Prefix&gt;&lt;Business Service Name&gt;&lt;Method Name&gt;&lt;Names of all the integration objects selected for the output complex type arguments&gt;</li> <li>If the method contains no complex arguments in the method, the naming convention is: &lt;Prefix&gt;&lt;Business Service Name&gt;&lt;Method Name&gt;</li> </ul>
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 should set the 'Folder' property with a unique value. This will avoid overwriting 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 'EASSiebelAdapter.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 17. 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> <b>Note:</b> 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) <b>Note:</b> 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) <b>Note:</b> 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 18. 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>. <b>Note:</b> 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> <b>Note:</b> 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 <b>Note:</b> 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 should set the 'Folder' property with a unique value. This will avoid overwriting 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, '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 19. 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>&lt;Prefix&gt;+&lt;Name of the Siebel Business Component&gt;+BG (with business graph specified) and                      &lt;Prefix&gt;+&lt;Name of the Siebel Business Component&gt; (without business graph specified)</p> <p><b>Note:</b> 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>

## Outbound configuration properties

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 WebSphere Process Server or WebSphere Enterprise Service Bus using WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

### Related tasks

“Changing configuration properties for embedded adapters” on page 85

To change 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 deployment properties and generating the service” on page 51

To generate the module, which is the artifact that is deployed on WebSphere Process Server or WebSphere Enterprise Service Bus, you create the module, include the adapter in the module, and specify an alias used to authenticate the caller to the Siebel Business Applications.

“Setting deployment properties and generating the service” on page 66

To generate the module, which is the artifact that is deployed on WebSphere Process Server or WebSphere Enterprise Service Bus, you create the module, include the adapter in the module, and specify an alias used to authenticate the caller to the Siebel Business Applications.

## 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 will process the field using its assigned default value, and that default value will also be displayed on the administrative console.</p> <p>Possible values are <b>Yes</b> and <b>No</b>.</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"> <li>• Yes, when the EventQueryType property is set to Dynamic</li> <li>• Yes, for Oracle databases</li> </ul>
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 the following:</p> <ul style="list-style-type: none"> <li>• Boolean</li> <li>• String</li> <li>• Integer</li> </ul>
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"> <li>• Must be uppercase</li> <li>• Must be 8 characters in length</li> </ul> <p>For versions of Rational Application Developer for WebSphere Software later than 6.40, the password:</p> <ul style="list-style-type: none"> <li>• Is not case sensitive</li> <li>• Can be up to 40 characters in length.</li> </ul> <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), Codepage 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 <b>Yes</b> and <b>No</b>.</p>
Bidi supported	<p>Indicates whether the property is supported in bidirectional (bidi) processing. Bidirectional processing pertains to the task of processing data that contains both left-to-right (Hebrew or Arabic, for example) and right-to-left (a URL or file path, for example) semantic content within the same file.</p> <p>Valid values are <b>Yes</b> and <b>No</b>.</p>



## Connection properties for the wizard

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

Once a connection between the external service wizard and the Siebel server is established, the external service wizard is able to access the metadata it needs from the Siebel 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 on how to read the property detail tables in the sections that follow, see “Guide to information about properties” on page 121.

**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 20. 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.
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).
Language code	The language code used to log on to the Siebel 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

Table 20. External service wizard properties (continued)

Property name in the wizard	Description
Siebel server view mode	Specifies the Siebel server mode and controls the kind of data 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 server, and the value is set to true, the adapter takes advantage of the load balancing feature to connect to the Siebel server more efficiently.
User name	The user name that is used to log into the Siebel application

### Adapter style (AdapterStyle)

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

Table 21. 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 22. Business object namespace details

Required	Yes
Default	<a href="http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel">http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel</a>
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	<a href="http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/IBMSiebelAccountInsertAccount">http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/IBMSiebelAccountInsertAccount</a>
Globalized	No
Bidi supported	No

### Connection URL (ConnectionURL)

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

Table 23. 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"> <li>• For Siebel 7.5: siebel://&lt;IP_address&gt;:2320/siebel/SSEObjMgr_ENU/sebldev1 .</li> <li>• For Siebel 7.8: siebel://&lt;IP_address&gt;/Sieb78/SSEObjMgr_enu .</li> <li>• For Siebel 8: siebel://&lt;IP_address&gt;:2321/SBA_80/SSEObjMgr_enu .</li> </ul>
Globalized	Yes
Bidi supported	Yes

### Delimiter for keys in the event store (DelimiterForKeysInTheEventStore)

Table 24. 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 don't 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 25. 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 26. 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. <b>Note:</b> The adapter removes special characters from the shorter business object names. Alpha-numeric 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 server.

Table 27. 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 server.
Globalized	No
Bidi supported	No

## Method name (MethodName)

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

Table 28. 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 29. 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 30. 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	Let's say, for 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 31. Siebel business object name for event store details

Required	Yes
Default	IBM_EVENT
Property type	String
Usage	After clicking on 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 32. 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 server view mode and controls the data that can be retrieved and what actions can be performed on it.

Table 33. Siebel server view mode details

Required	Yes
Default	3
Property type	Integer
Usage	This property displays after clicking on 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 34. Type of Siebel objects to discover details

Required	Yes
Possible values	Siebel business objects and 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 server uses resonate support.

Table 35. 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 on 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 will take advantage of the load balancing feature to connect to the Siebel 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 into the Siebel application.

Table 36. 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 using the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

The following properties for logging and tracing are no longer required in version 6.1.0, 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 on how to read the property detail tables in the sections to follow, see "Guide to information about properties" on page 121.

Table 37. 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 strings of X's instead of user data in 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 server, and the value is set to true, the adapter will take advantage of the load balancing feature to connect to the Siebel server more efficiently
Siebel server view mode	viewMode	Specifies the Siebel view mode and controls the kind of data to be retrieved and what actions can 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 38. Adapter ID details

Required	Yes
Default	001
Property type	String



Table 38. Adapter ID details (continued)

Usage	<p>This property identifies the adapter instance in 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, make sure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate 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 log and trace information to a particular instance of an adapter.</p> <p>To illustrate how the length of the adapter ID property affects the filtering of log and trace files, suppose 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. For example, suppose you set the adapter ID properties of two instances to Instance01 and Instance02. You will not be able to separately 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, can be set both at the resource adapter level and the managed connection factory level. After using 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 WebSphere Integration Developer assembly editor or the administrative console to reset these properties, make sure 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) property

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 39. 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, can be set both at the resource adapter level and the managed connection factory level. After using 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 WebSphere Integration Developer assembly editor or the administrative console to reset these properties, make sure you set them consistently, to prevent inconsistent marking of the log and trace entries.</p>
Globalized	No

Table 39. 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 40. 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 41. Log file maximum size details

Required	No
Default	0
Property type	Integer
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 42. 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 43. 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 server uses resonate support.

Table 44. Resonate support details

Required	No
Possible values	True False
Default	True
Property type	Boolean
Usage	If the check box is selected, the value for Resonate Support is set to true, and the adapter takes advantage of the load balancing feature to connect to the Siebel server more efficiently. If the check box is not selected, the value for Resonate Support is set to false.
Globalized	No

### Siebel server view mode (SiebelServerViewMode)

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

Table 45. 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 46. 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

Table 46. Trace file maximum size details (continued)

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

### Trace file name (TraceFilename)

This property specifies the full path of the trace file.

Table 47. 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 48. 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 WebSphere® Integration Developer 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 on how to read the property detail tables in the sections that follow, see “Guide to information about properties” on page 121.

**Note:** The external service wizard refers to these properties as managed connection factory properties and the WebSphere Process Server administrative console refers to them as (J2C) connection factory properties.

Table 49. 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
Disguise user data as "XXX" in log and trace files	HideConfidentialTrace	Specifies whether to disguise potentially sensitive information by writing strings of X's instead of user data in log and trace files
Language code	languageCode	The language code used to log on to the Siebel server
Password	password	The password for the corresponding user name
Prefix	prefix	The prefix for the business object name
Resonate support	resonateSupport	Specifies that if resonate support is installed on the Siebel server, and the value is set to true, the adapter takes advantage of the load balancing feature to connect to the Siebel server more efficiently.
User name	userName	The user name that is used to log into 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 50. Adapter ID details

Required	Yes
Default	001
Property type	String

Table 50. Adapter ID details (continued)

Usage	<p>This property identifies the adapter instance in 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, make sure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate 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 log and trace information to a particular instance of an adapter.</p> <p>To illustrate how the length of the adapter ID property affects the filtering of log and trace files, suppose 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. For example, suppose you set the adapter ID properties of two instances to Instance01 and Instance02. You will not be able to separately 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, can be set both at the resource adapter level and the managed connection factory level. After using 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 WebSphere Integration Developer assembly editor or the administrative console to reset these properties, make sure you set them consistently, to prevent inconsistent marking of the log and trace entries.</p>
Globalized	Yes
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. For Siebel 7.0.5 to 7.5x : siebel://&lt;IP ADDRESS&gt;/siebel/SSEObjMgr_ENU/sebldev1. For Siebel 7.8 : siebel://&lt;IP ADDRESS&gt;:2321/Sieb78/SSEObjMgr_enu. For Siebel 8 : siebel://&lt;IP ADDRESS&gt;:2321/SBA_80/SSEObjMgr_enu. The default port number, is 2320. In the examples above (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) property

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
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, can be set both at the resource adapter level and the managed connection factory level. After using 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 WebSphere Integration Developer assembly editor or the administrative console to reset these properties, make sure you set them consistently, to prevent inconsistent marking of the log and trace entries.
Globalized	No
Bidi supported	No

### Language code (LanguageCode)

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

Table 53. 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). This is used to log on to the Siebel server.
Globalized	No
Bidi supported	No

### Password (Password)

This property specifies the password for the corresponding user name.

Table 54. Password details

Required	Yes
Default	No default value
Property type	String
Example	sadmin
Usage	This property will display after clicking on 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.
Globalized	Yes
Bidi supported	Yes

## Prefix (Prefix)

This property specifies the prefix for the business object name.

Table 55. 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	If you use the prefix, IBM, generate a business object for the EAI Siebel Adapter and the Insert method, and 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

## Resonate support (ResonateSupport)

This property indicates whether the Siebel server uses resonate support.

Table 56. Resonate support details

Required	No
Possible values	True False
Default	True
Property type	Boolean
Usage	If the check box is selected, it will be set to true, and the adapter will take advantage of the load balancing feature to connect to the Siebel 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 into the Siebel application.

Table 57. User name details

Required	Yes
Possible values	None
Default	No default value
Property type	String
Usage	This property will display after clicking on 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.
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 58. View mode details

Required	Yes
Default	Although the adapter supports values 1 - 9, the default value is 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

## 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 WebSphere Integration Developer. The properties reside in the method binding of the import.

Table 59 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 121.

Table 59. Interaction specification property for the Adapter for Siebel Business Applications

Property name		Description
In the wizard	In the assembly editor	
“Maximum records (MaximumRecords)”	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 60. 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 error MatchesExceededLimitException. The adapter uses this property to help avoid out-of-memory issues.
Property type	Integer
Globalized	No

Table 60. Maximum records details (continued)

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

**Related tasks**

Chapter 5, “Changing interaction specification properties using the assembly editor,” on page 73

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

## 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 WebSphere Integration Developer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

**Related tasks**

“Changing configuration properties for embedded adapters” on page 85

To change 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 deployment properties and generating the service” on page 51

To generate the module, which is the artifact that is deployed on WebSphere Process Server or WebSphere Enterprise Service Bus, you create the module, include the adapter in the module, and specify an alias used to authenticate the caller to the Siebel Business Applications.

“Setting deployment properties and generating the service” on page 66

To generate the module, which is the artifact that is deployed on WebSphere Process Server or WebSphere Enterprise Service Bus, you create the module, include the adapter in the module, and specify an alias used to authenticate the caller to the Siebel Business Applications.

## 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 will process the field using its assigned default value, and that default value will also be displayed on the administrative console.</p> <p>Possible values are <b>Yes</b> and <b>No</b>.</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"> <li>• Yes, when the EventQueryType property is set to Dynamic</li> <li>• Yes, for Oracle databases</li> </ul>
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 the following:</p> <ul style="list-style-type: none"> <li>• Boolean</li> <li>• String</li> <li>• Integer</li> </ul>
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"> <li>• Must be uppercase</li> <li>• Must be 8 characters in length</li> </ul> <p>For versions of Rational Application Developer for WebSphere Software later than 6.40, the password:</p> <ul style="list-style-type: none"> <li>• Is not case sensitive</li> <li>• Can be up to 40 characters in length.</li> </ul> <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), Codepage 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 <b>Yes</b> and <b>No</b>.</p>
Bidi supported	<p>Indicates whether the property is supported in bidirectional (bidi) processing. Bidirectional processing pertains to the task of processing data that contains both left-to-right (Hebrew or Arabic, for example) and right-to-left (a URL or file path, for example) semantic content within the same file.</p> <p>Valid values are <b>Yes</b> and <b>No</b>.</p>

## Connection properties for the wizard

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

Once a connection between the external service wizard and the Siebel server is established, the external service wizard is able to access the metadata it needs from the Siebel 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 on how to read the property detail tables in the sections that follow, see “Guide to information about properties” on page 121.

**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 61. 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.
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).
Language code	The language code used to log on to the Siebel 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

Table 61. External service wizard properties (continued)

Property name in the wizard	Description
Siebel server view mode	Specifies the Siebel server mode and controls the kind of data 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 server, and the value is set to true, the adapter takes advantage of the load balancing feature to connect to the Siebel server more efficiently.
User name	The user name that is used to log into the Siebel application

### Adapter style (AdapterStyle)

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

Table 62. 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 63. Business object namespace details

Required	Yes
Default	<a href="http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel">http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel</a>
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	<a href="http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/IBMSiebelAccountInsertAccount">http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/IBMSiebelAccountInsertAccount</a>
Globalized	No
Bidi supported	No

### Connection URL (ConnectionURL)

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

Table 64. 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"> <li>• For Siebel 7.5: siebel://&lt;IP_address&gt;:2320/siebel/SSEObjMgr_ENU/sebldev1 .</li> <li>• For Siebel 7.8: siebel://&lt;IP_address&gt;/Sieb78/SSEObjMgr_enu .</li> <li>• For Siebel 8: siebel://&lt;IP_address&gt;:2321/SBA_80/SSEObjMgr_enu .</li> </ul>
Globalized	Yes
Bidi supported	Yes

### Delimiter for keys in the event store (DelimiterForKeysInTheEventStore)

Table 65. 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 don't 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 66. 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 67. 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. <b>Note:</b> The adapter removes special characters from the shorter business object names. Alpha-numeric 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 server.

Table 68. 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 server.
Globalized	No
Bidi supported	No

## Method name (MethodName)

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

Table 69. 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 70. 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 71. 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	Let's say, for 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 72. Siebel business object name for event store details

Required	Yes
Default	IBM_EVENT
Property type	String
Usage	After clicking on 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 73. 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 server view mode and controls the data that can be retrieved and what actions can be performed on it.

Table 74. Siebel server view mode details

Required	Yes
Default	3
Property type	Integer
Usage	This property displays after clicking on 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 75. Type of Siebel objects to discover details

Required	Yes
Possible values	Siebel business objects and 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 server uses resonate support.

Table 76. 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 on 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 will take advantage of the load balancing feature to connect to the Siebel 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 into the Siebel application.

Table 77. 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 using the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

The following properties for logging and tracing are no longer required in version 6.1.0, 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 on how to read the property detail tables in the sections to follow, see "Guide to information about properties" on page 121.

Table 78. 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 strings of X's instead of user data in 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 server, and the value is set to true, the adapter will take advantage of the load balancing feature to connect to the Siebel server more efficiently
Siebel server view mode	viewMode	Specifies the Siebel view mode and controls the kind of data to be retrieved and what actions can 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 79. Adapter ID details

Required	Yes
Default	001
Property type	String

Table 79. Adapter ID details (continued)

Usage	<p>This property identifies the adapter instance in 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, make sure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate 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 log and trace information to a particular instance of an adapter.</p> <p>To illustrate how the length of the adapter ID property affects the filtering of log and trace files, suppose 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. For example, suppose you set the adapter ID properties of two instances to Instance01 and Instance02. You will not be able to separately 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, can be set both at the resource adapter level and the managed connection factory level. After using 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 WebSphere Integration Developer assembly editor or the administrative console to reset these properties, make sure 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) property

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 80. 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, can be set both at the resource adapter level and the managed connection factory level. After using 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 WebSphere Integration Developer assembly editor or the administrative console to reset these properties, make sure you set them consistently, to prevent inconsistent marking of the log and trace entries.</p>
Globalized	No

Table 80. 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 81. 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 82. Log file maximum size details

Required	No
Default	0
Property type	Integer
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 83. 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 84. 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 server uses resonate support.

Table 85. Resonate support details

Required	No
Possible values	True False
Default	True
Property type	Boolean
Usage	If the check box is selected, the value for Resonate Support is set to true, and the adapter takes advantage of the load balancing feature to connect to the Siebel server more efficiently. If the check box is not selected, the value for Resonate Support is set to false.
Globalized	No

### Siebel server view mode (SiebelServerViewMode)

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

Table 86. 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 87. 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

Table 87. Trace file maximum size details (continued)

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

### Trace file name (TraceFilename)

This property specifies the full path of the trace file.

Table 88. 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 89. 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

## Activation specification properties

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

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

The following table lists the activation specification properties for inbound communication. You set the activation specification properties using the external service wizard and can change them using the WebSphere® Integration Developer 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 121.

Table 90. Activation specification properties

Property name		Description
In the wizard	In the administrative console	
Connection URL	connectString	The connection URL needed to connect to the Siebel application
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
Retry limit for failed events (FailedEventRetryLimit)	FailedEventRetryLimit	The number of times the adapter attempts to redeliver an event before marking the event as failed
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 server
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 that 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
Retry EIS connection on startup	RetryConnectionOnStartup	Controls whether the adapter retries the connection to the Siebel application if it cannot connect at startup
Number of times to retry the system connection	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 interval if connection fails	RetryInterval	The length of time that the adapter waits between attempts to establish a new 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 into the Siebel application

### Connection URL (ConnectionURL)

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

Table 91. Connection URL details

Required	Yes
Default	No default value



Table 91. Connection URL details (continued)

Property type	String
Usage	Protocol://machinename:port/enterprisename/object manager/server name. 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. In the examples above (for Siebel versions 7.8 and 8), another port (2321) has been specified.
Globalized	Yes
Bidi supported	Yes

### Type of delivery (DeliveryType)

This property specifies the order in which events are delivered by the adapter to the export.

Table 92. Type of delivery details

Required	No
Possible values	ORDERED UNORDERED
Default	ORDERED
Property type	String
Usage	The following values are supported: <ul style="list-style-type: none"> <li>• ORDERED: The adapter delivers events to the export one at a time.</li> <li>• UNORDERED: The adapter delivers all events to the export at once.</li> </ul>
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 93. 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 assured-only event delivery (AssuredOnceDelivery)

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

Table 94. Ensure assured-only event delivery details

Required	Yes
Possible values	True False
Default	True
Property type	Boolean
Usage	<p>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.</p> <p>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.</p> <p>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.</p>
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 95. Event component name details

Required	Yes
Default	IBM2 (for Siebel version 7.x) and IBM Event (for Siebel version 8)
Property type	String
Usage	<p>The default is IBM2, for Siebel version 7.x and IBM Event, for Siebel version 8. If you select one of these defaults to configure the event business component on the Siebel server, it will be 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.</p>
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 96. Event types to process details

Required	No
Possible values	A comma-delimited (,) list of business object types
Default	null
Property type	String

Table 96. Event types to process details (continued)

Usage	Events are filtered by business object type. 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.
Example	To receive only events relating to the Customer and Order business objects, specify this value: Customer,Order  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.
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 97. Retry limit for failed events details

Required	No
Possible values	Integers
Default	5
Property type	Integer
Usage	Use this property to control how many times the adapter tries to send an event before marking it as failed. It accepts the following values:  <b>Default</b> If this property is not set, the adapter tries five additional times before marking the event as failed.  <b>0</b> 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.  <b>&gt; 0</b> For integers greater than zero, the adapter retries the specified number of times before marking the event as failed.  <b>&lt; 0</b> For negative integers, the adapter does not retry failed events.
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 98. 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

Table 98. Interval between polling periods details (continued)

Usage	The poll period is established at a fixed rate, which means that if running the poll cycle is delayed for any reason (for example, if a prior poll cycle takes longer than expected to complete) the next poll cycle will occur immediately to make up for the lost time caused by the delay.
Globalized	No
Bidi supported	No

### Language code (LanguageCode)

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

Table 99. 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 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 100. Maximum connections details

Required	No
Default	1
Property type	Integer
Usage	Only positive values are valid. The adapter considers any positive entry less than 1 to be equal to 1. Typing a negative value or 1 for this property may result in runtime errors.
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 101. Maximum events in polling period details

Required	Yes
Default	10
Property type	Integer
Usage	The value must be greater than 0. If this value is increased, more events are processed per polling period and the adapter may perform less efficiently. If this value is decreased, fewer events are processed per polling period and the adapter's performance may improve slightly.
Globalized	No

Table 101. Maximum events in polling period details (continued)

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

### Minimum connections (MinimumConnections)

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

Table 102. 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. Typing a negative value or 1 for this property may result in run time errors.
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 103. 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	Yes
Bidi supported	No

### Password (Password)

This property specifies the password for the corresponding user name.

Table 104. Password details

Required	Yes
Default	No default value
Property type	String
Usage	This property will display after clicking on 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.
Example	admin

Table 104. Password details (continued)

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 105. 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"> <li>• 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.</li> <li>• 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>.</li> </ul>
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 establish a new connection.

Table 106. Retry interval details

Required	Yes
Default	2000
Unit of measure	Milliseconds
Property type	Integer
Usage	Only positive values are valid. When the adapter encounters an error related to the inbound connection, this property specifies the length of time the adapter waits before trying to establish a new connection.
Globalized	Yes
Bidi supported	No

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

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

Table 107. 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 into the Siebel application.

Table 108. User name details

Required	Yes
Default	No default value
Property type	String
Usage	This property will display after clicking on 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.
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, WebSphere Integration Developer, and WebSphere Process Server, 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. WebSphere Process Server 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

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

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

Table 109. Bidirectional format attributes

Letter position	Purpose	Values	Description	Default setting
1	Order schema	I	Implicit (Logical)	I
		V	Visual	
2	Direction	L	Left-to-Right	L
		R	Right-to-Left	
		C	Contextual Left-to-Right	
		D	Contextual Right-to-Left	



Table 109. Bidirectional format attributes (continued)

Letter position	Purpose	Values	Description	Default setting
3	Symmetric Swapping	Y	Symmetric swapping is on	Y
		N	Symmetric swapping is off	
4	Text Shaping	S	Text is shaped	N
		N	Text is not shaped (Nominal)	
		I	Initial shaping	
		M	Middle shaping	
		F	Final shaping	
		B	Isolated shaping	
5	Numeric Shaping	H	National (Hindi)	N
		C	Contextual shaping	
		N	Numbers are not shaped (Nominal)	

### Bidirectional properties that identify data for transformation

To identify business data subject to transformation, set the `BiDiContextEIS` property. Do this by specifying values for each of the five bidirectional format attributes (listed in the preceding table) for the property. The `BiDiContextEIS` property can be set for the managed connection factory and the activation specification.

To identify event persistence data subject to transformation, set the `BiDiFormatEP` property. Do this by specifying values for each of the five bidirectional format attributes (listed in the preceding table) for the property. 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. Do this by using the business object editor within WebSphere Integration Developer to add the properties as application-specific elements of a business object.

#### Related reference

“Activation specification properties” on page 153

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

“Managed connection factory properties” on page 134

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. Once these properties are set,

bidirectional script data is correctly processed and displayed in WebSphere Integration Developer and WebSphere Process Server 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/v6r2mx/topic/com.ibm.wbit.620.help.messages.doc/messages.html>

The displayed Web page shows a list of message prefixes. Click a message prefix to see all the messages with that prefix:

- Messages with the prefix 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 the WebSphere Adapter for Siebel Business Applications.

## Samples and tutorials

To help you use WebSphere Adapters, samples and tutorials are available from the Business Process Management Samples and Tutorials Web site. You can access the samples and tutorials in either of the following ways:

- From the welcome page that opens when you start WebSphere Integration Developer. To see samples and tutorials for WebSphere Adapter for Siebel Business Applications, click **Retrieve**. Then browse the displayed categories to make your selections.
- At this location on the Web: <http://publib.boulder.ibm.com/bpcsamp/index.html>.

## 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

- WebSphere Business Process Management, version 6.2, information center, which includes WebSphere Process Server, WebSphere Enterprise Service Bus, and WebSphere Integration Developer information: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r2mx/index.jsp>
- WebSphere Adapters, version 6.1.x, information center: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.wbit.612.help.adapter.emd.ui.doc/topics/tcreatecmps.html>
- WebSphere Business Integration Adapters information center: [http://publib.boulder.ibm.com/infocenter/wbihelp/v6rxmx/index.jsp?topic=com.ibm.wbi\\_adapters.doc/welcome\\_adapters.htm](http://publib.boulder.ibm.com/infocenter/wbihelp/v6rxmx/index.jsp?topic=com.ibm.wbi_adapters.doc/welcome_adapters.htm)

## developerWorks® resources

- WebSphere Adapter Toolkit
- WebSphere business integration zone

## Support and assistance

- WebSphere Adapters technical support: <http://www.ibm.com/software/integration/wbiadapters/support/>
- WebSphere Adapters technotes: <http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>. In the **Product category** list, select the name of the adapter and click **Go**.



---

## 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 (<sup>®</sup> or <sup>™</sup>), 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
  - list of 153
  - setting in administrative console 89, 94
  - setting with external service wizard 153
- adapter application
  - starting 95
  - stopping 96
- adapter architecture 3
- Adapter for Siebel Business Applications
  - administering 85
  - overview 1
- Adapter for Siebel Business Applications module
  - exporting as EAR file 82
  - installing EAR file on server 82
  - starting 95
  - stopping 96
- adapter messages 164
- adapter performance 96
- adapter technotes 165
- AdapterID property 129, 148
- AdapterStyle property 123, 142
- alias, authentication 33
- application-specific information
  - business-object-level metadata 115
  - container attribute-level metadata 115
  - property-level metadata 115
  - siebel business components 115
  - siebel business objects 115
  - siebel business services 115
- ApplyChanges operation 4
- AssuredOnceDelivery property 153
- authentication
  - description 16
  - external service wizard 17
  - run time 17
- authentication alias 17, 33

## B

- backward compatibility
  - project interchange files 24
  - projects 24
- bidirectional data transformation properties
  - activation specification properties 164
  - connection properties used in the external service wizard 164
  - list of 164
  - managed connection properties 164
- business faults 107
- business graph 4, 7, 8
- business object information 115
- business objects
  - application-specific information 115
  - configuring for inbound processing 64
  - configuring for outbound processing 49
  - creating
    - structure 12
  - naming conventions 118
  - selecting for inbound processing 58

- business objects (*continued*)
  - selecting for outbound processing 43
  - supported data operations 117
- business-object-level metadata 115
- BusinessObjectNameSpace property 123, 142

## C

- CEI (Common Event Infrastructure) 100
- clustered environment
  - deploying in 20
  - description 20
  - inbound processes 21
  - outbound processes 21
- Common Event Infrastructure (CEI) 100
- compatibility matrix 2
- confidential data, disguising 15
- confidential tracing 15
- configuration overview 30
- configuring
  - logging 104
  - Performance Monitoring Infrastructure (PMI) 97
  - tracing 104
- connection properties, external service wizard 40, 55
- ConnectionURL property 123, 134, 142, 153
- container attribute-level metadata 116
- Create operation 5
- custom properties
  - activation specification 89, 94
  - managed connection factory 87, 92
  - resource adapter 85, 91

## D

- data operations 117
- debugging
  - self-help resources 113
  - XAResourceNotAvailableException exception 113
- Delete operation 6
- DelimiterForKeysInTheEventStore property 123, 142
- DeliveryType property 153
- deployment
  - environments 75
  - options 18
  - to production environment 80
  - to test environment 75
- developerWorks 165
- developerWorks resources, WebSphere Adapters 165

## E

- EAR file
  - exporting 82
  - installing on server 82
- education, WebSphere Adapters 165
- embedded adapter
  - activation specification properties, setting 89
  - considerations for using 19
  - description 18
  - managed connection factory properties, setting 87

- embedded adapter (*continued*)
  - resource adapter properties, setting 85
- embedded deployment 76
- enableHASupport property 21, 129, 148
- event business components 8
- event notification 3
- event record 8
- event store
  - event business components 8
  - event record 8
  - event triggers 8
  - polling 8
- event table
  - creating 31
- event triggers 8
- EventComponentName property 153
- EventDelimiter property 129, 148
- EventTypeFilter property 153
- exceptions
  - XAResourceNotAvailableException 113
- Exists operation 6
- exporting module as EAR file 82
- external dependencies, adding 38, 75, 76
- external service wizard
  - authentication in 17
  - overview 13
  - properties, connection 123, 142
  - setting connection properties 40
  - starting 35

## F

- faults
  - description 107
- FFDC (first-failure data capture) 106
- files
  - SystemOut.log log file 105
  - trace.log trace file 105
- FilterFutureEvents property 153
- first-failure data capture (FFDC) 106
- Folder property 123, 142

## G

- GenerateBusinessObjectsWithShorterNames property 123, 142
- globalization and bidirectional data transformation
  - bidirectional properties that identify data for transformation 162
  - bidirectional script data formats 162
  - bidirectional script data transformation 162

## H

- hardware and software requirements 2
- hardware requirements 2
- high-availability environment
  - deploying in 20
  - description 20
  - inbound processes 21
  - outbound processes 21

## I

- IBM WebSphere Adapter Toolkit 165
- implementation, Java 77

- inbound configuration properties 140
- inbound processing
  - business graph 8
  - generating the service 66
  - inbound process 66
  - overview 3, 8
  - selecting business objects 58
  - setting deployment properties 66
- installing EAR file 82
- interaction specification properties
  - changing 73
  - list of 139
  - setting in assembly editor 139
  - setting with external service wizard 139

## J

- JAR file, adding external 38, 75, 76
- Java 2 security 17
- Java implementation 77

## L

- LanguageCode property 123, 134, 142, 153
- Log Analyzer 104
- Log and Trace Analyzer, support for 103
- log and trace files 103
- log files
  - changing file name 105
  - disabling 104
  - enabling 104
  - level of detail 104
  - location 106
- LogFileMaxSize property 129, 148
- LogFilename property 129, 148
- logging
  - configuring properties with administrative console 104
- LogNumberOfFiles property 129, 148

## M

- managed (J2C) connection factory properties
  - list of 134
  - setting in administrative console 87, 92, 134
  - setting with external service wizard 134
- matrix, compatibility 2
- MaximumConnections property 153
- MaximumRecords property 139
- messages, adapter 164
- metadata 115
  - attribute-level 116
  - business-object-level 115
  - property-level 116
- MethodName property 123, 142
- migration considerations 22
- MinimumConnections property 153
- monitoring performance 96

## N

- naming conventions for business objects
  - business graph 118
  - siebel business components 118
  - siebel business services 118
  - siebel integration components 118

## O

- outbound configuration properties 121
- outbound operations 4
  - ApplyChanges 4
  - Create 5
  - Delete 6
  - Exists 6
  - Retrieve 7
  - RetrieveAll 7
  - Update 8
- outbound processing
  - ApplyChanges operation 3
  - Create operation 3
  - Delete operation 3
  - Exists operation 3
  - generating the service 51
  - outbound process 51
  - overview 3
  - Retrieve operation 3
  - RetrieveAll operation 3
  - selecting business objects 43
  - setting deployment properties 51
  - supported outbound operations
    - business object hierarchy 3
  - Update operation 3

## P

- package files for adapters 105
- Password property 123, 134, 142, 153
- Performance Monitoring Infrastructure (PMI)
  - configuring 97
  - description 96
  - viewing performance statistics 99
- performance statistics 99
- PMI (Performance Monitoring Infrastructure)
  - configuring 97
  - description 96
  - viewing performance statistics 99
- polling 8
- PollPeriod property 153
- PollQuantity property 153
- Prefix property 134
- PrefixForBusinessObjectNames property 123, 142
- problem determination
  - self-help resources 113
  - solutions to common problems 110
  - XAResourceNotAvailableException exception 113
- project interchange (PI) file
  - updating without migrating 24
- project, creating 35
- properties
  - activation specification 89, 94
  - configuration properties
    - inbound 140
    - outbound 121
  - external service connection 123, 142
  - inbound configuration 140
  - managed (J2C) connection factory 87, 92
  - outbound configuration 121
  - resource adapter 85, 91
- property-level metadata 116

## R

- RAR (resource adapter archive) file
  - description 81
  - installing on server 81
- record, event 8
- Redbooks, WebSphere Adapters 165
- related information 165
- related products, information 165
- requirements, hardware and software 2
- ResonateSupport property 129, 134, 148
- resource adapter archive (RAR) file
  - description 81
  - installing on server 81
- resource adapter properties
  - list of 129, 148
  - setting in administrative console 85, 91
  - setting with external service wizard 129, 148
- Retrieve operation 7
- RetrieveAll operation 7
- Retry limit property 159
- RetryInterval property 153
- RetryLimit property 153
- road map for configuring the module 29
- runtime environment
  - authentication in 17
  - deploying EAR file to 80

## S

- samples 27
- security
  - disguising sensitive data 15
- security, Java 2 17
- self-help resources 113
- sensitive data, disguising 15
- setting connection properties 40, 55
- SiebelBusinessObjectNameForEventStore property 123, 142
- SiebelRepositoryName property 123, 142
- SiebelServerViewMode property 123, 129, 142, 148
- software dependencies, adding external 38, 75, 76
- software requirements 2
- solutions to common problems
  - adapter does not generate an exception 110
  - adapter does not generate artifacts properly 110
  - adapter generates multiple container attributes 110
  - adapter may time out 110
- stand-alone adapter
  - activation specification properties, setting 94
  - considerations for using 20
  - description 18
  - managed connection factory properties, setting 92
  - resource adapter properties, setting 91
- starting adapter applications 95
- stopping adapter applications 96
- StopPollingOnError property 153
- support
  - overview 103
  - self-help resources 113
  - technical 165
- supported data operations 117
- SystemOut.log file 105

## T

- target component 77
- technical support 165

- technotes 2, 113, 165
- technotes, WebSphere Adapters 165
- test environment
  - adding module to 78
  - deploying to 75, 78
  - testing modules 79
- trace files
  - changing file name 105
  - disabling 104
  - enabling 104
  - level of detail 104
  - location 106
- trace.log file 105
- TraceFileMaxSize property 129, 148
- TraceFilename property 129, 148
- TraceNumberOfFiles property 129, 148
- tracing
  - configuring properties with administrative console 104
- triggers, event 8
- troubleshooting
  - overview 103
  - self-help resources 113
  - XAResourceNotAvailableException exception 113
- tutorials 27
- TypeOfSiebelObjectsToDiscover property 123, 142

## U

- Update operation 8
- UseResonateSupportForLoadBalancingOnSiebelServer property 123, 142
- UserName property 123, 134, 142, 153

## V

- ViewMode property 134

## W

- WebSphere Adapters, version 6.0, information 165
- WebSphere Adapters, version 6.0.2.x, information 165
- WebSphere Application Server information 165
- WebSphere Business Integration Adapters information 165
- WebSphere Business Process Management, version 6.1.x, information 165
- WebSphere Enterprise Service Bus information 165
- WebSphere Extended Deployment 20
- WebSphere Integration Developer 35
  - information 165
  - test environment 75
- WebSphere Process Server information 165
- WebSphere Process Server or WebSphere Enterprise Service Bus
  - deploying to 80
- wiring components 77

## X

- XAResourceNotAvailableException 113





Printed in USA