Configuring Project Components for Oracle® Java CAPS Application Adapters

ORACLE®

Contents

Configuring JAVA CAPS Project Components for Application Adapters

What You Need To Do

The following section provides instructions on how to configure Java CAPS project components for application adapters:

- "Configuring Adapter Connectivity Map Properties" on page 5

What You Need To Know

The following section provides information about the Java CAPS project properties for application adapters:

- "Oracle Applications Adapter Properties" on page 7
- "PeopleSoft Adapter Connectivity Map Properties" on page 8
- "SAP BAPI Adapter Connectivity Map Properties" on page 10
- "Siebel EAI Adapter Connectivity Map Properties" on page 15
- "SWIFT Alliance Gateway Adapter Connectivity Map Properties" on page 17
- "WebSphere MQ Adapter Connectivity Map Properties" on page 30

Configuring Adapter Connectivity Map Properties

Modifications to Connectivity Map configuration properties are made from the Adapter Properties Editor. A description of each property is displayed in the Description pane when it is selected. This provides a brief explanation of the required settings or options. The Comments pane provides an area to record notes and information regarding the currently selected property. These comments are saved when you close the editor.

▼ To Add an Adapter to the Connectivity Map

1   Open the Connectivity Map opened in the Connectivity Map Editor.

2   In the Connectivity Map Editor toolbar, click the External Applications icon.

A list of all available External Applications appears.

3   Click the External Application you want to add.

An icon for the application appears on the toolbar to the right of the External Applications icon.

4   Select the new icon, and drag and drop it to the Connectivity Map editor canvas.

The External Application appears on the canvas and in the project tree.

5   Drag a connection between the External Application and a Service, such as a Java Collaboration Definition (JCD) or Business Process.

When you connect an External Application to a Service, the Connectivity Map Editor automatically assigns the appropriate adapter to the link (see the figure below). Each adapter is supplied with a template containing its default configuration properties, which are accessible from the Connectivity Map.
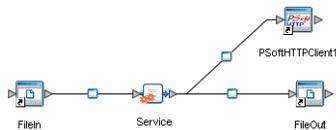
▼ To Configure Adapter Connectivity Map Properties

1   From the Connectivity Map, double click the adapter icon located in the link between the associated External Application and the Service.

2   If a dialog box appears requesting that you select the type of connection, make your selection and then click OK.

This dialog box appears when there is more than one type of Adapter for the selected External Application. For example, some Adapters have a transactional mode and non-transactional mode.

The Properties Editor appears.

3   From the Properties Editor, click on any folder to display the default configuration properties for that section.

4   Click on any property field to make it editable.

5 If an ellipsis appears next to a field, you can click the ellipsis button to open an editor for the field.

This is useful for long field values.

6 Make any necessary changes to the property values.

7 Click OK to close the Properties Editor and save the settings.

Note – The available properties for the Applications Adapters are described in the following sections.

Oracle Applications Adapter Properties

The following sections provide information about Oracle Applications Adapter Connectivity Map properties:

- "Outbound Oracle Applications Adapter with no XA Support" on page 7
- "Outbound Adapter with XA Support" on page 7

Note that there are no properties to configure for inbound Oracle Applications Adapters.

Outbound Oracle Applications Adapter with no XA Support

The JDBC Connector Settings section of the Oracle Applications Connectivity Map properties contains the properties listed in the following table.

TABLE 1 Outbound Connectivity Map JDBC Connector Settings

Name	Description	Required Value
Description	Specifies the description for the database.	A valid string. The default is Oracle thin driver Connection Pool Datasource.
ClassName	Specifies the Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface.	A valid class name. The default value is `oracle.jdbc.pool.OracleConnectionPoolDataSource`.

Outbound Adapter with XA Support

The JDBC Connector Settings section of the Oracle Applications Connectivity Map properties contains the properties listed in the following table.

TABLE 2 Outbound with XA Support JDBC Connector Settings

Name	Description	Required Value
Description	Specifies the description for the database.	A valid string. The default is Oracle thin driver XA Datasource.
ClassName	Specifies the Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface.	A valid class name. The default value is `oracle.jdbc.xa.client.OracleXADataSource`.

PeopleSoft Adapter Connectivity Map Properties

This task describes the Connectivity Map properties of the PeopleSoft Adapter. The PeopleSoft Adapter includes two component adapters:

- PeopleSoft HTTP Client Adapter: The PeopleSoft HTTP Client Adapter includes both Connectivity Map and Environment properties and must be configured from both locations. Only the outbound Adapter required configuration in the Connectivity Map.

- PeopleSoft HTTP Server Adapter: The PeopleSoft HTTP Server Adapter includes only Connectivity Map properties. There are no Environment properties.

Once a component adapter and the Environment (with the external systems) have been created, the properties can be modified for your specific system.

The PeopleSoft Adapter properties are organized into the following sections:

PeopleSoft HTTP Client Outbound Connectivity Map Properties

The PeopleSoft HTTP Client Adapter properties, accessed from the Connectivity Map, are organized into the HTTP Settings category.

Note – SSL and proxy configurations (HTTPS) are not supported.

HTTP Settings

The HTTP Settings section of the PeopleSoft HTTP Client Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 3   Connectivity Map - HTTP Settings

Name	Description	Required Value
Allow cookies	Specifies whether cookies sent from servers are allowed to be stored and sent on subsequent requests. If cookies are not allowed, sessions are not supported.	True or False. True indicates that cookies sent from the server are allowed. **False** indicates that Cookies are not accepted. The configured default is **True**.
Accept type	Specifies the default Accept-Type header value included when a request is sent to the server; for example, text/html, text/plain, text/xml.	An Accept-type header value. The configured default is text/xml.

Caution – Calling the clear() method in the Collaboration Editor clears all HTTP properties. Once the properties have been cleared, you must manually rebuild the header and payload sections of the Request message in the Transformation Designer.

PeopleSoft HTTP Server Inbound Connectivity Map Properties

The PeopleSoft HTTP Server Adapter properties, accessed from the Connectivity Map, are organized under the HTTP Server External Configuration category.

Note – The PeopleSoft HTTP Server Adapter does not have any Environment properties.

HTTP Server External Configuration

The HTTP Server External Configuration section of the PeopleSoft HTTP Server Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 4 Connectivity Map - HTTP Server External Configuration

Name	Description	Required Value
servlet-url	Specifies the last path component of the HTTP server servlet URL. This URL is the one the client uses to access the server. The property value must be the servlet name, for example: HttpServerServlet. The total URL is made up of several components, including the Project deployment name and the value entered for this property. The servlet name must match the Primary URL property on the PeopleSoft server. For example, a complete servlet URL appears as follows: http://<hostname>:18001/<deploymentname>_servlet_<servletname>/<servletname> Where: hostname is the name of the machine where your current Application Server is running. 18001 is the port number. deploymentname is the name of your Project's Deployment Profile concatenated with **_servlet**. servletname is the name of the PeopleSoft servlet. This is the value needed for this property. **Note –** The servlet-url parameter does not support LDAP values.	Enter the name of the PeopleSoft Servlet. This is the last path component of the HTTP server servlet URL.

SAP BAPI Adapter Connectivity Map Properties

When you connect an External Application to a Collaboration, Java CAPS IDE automatically assigns the appropriate Adapter to the link. Each Adapter is supplied with a template containing default configuration properties that are accessible on the Connectivity Map. The following figure illustrates the SAP BAPI properties.

FIGURE 1 Outbound Adapter Properties
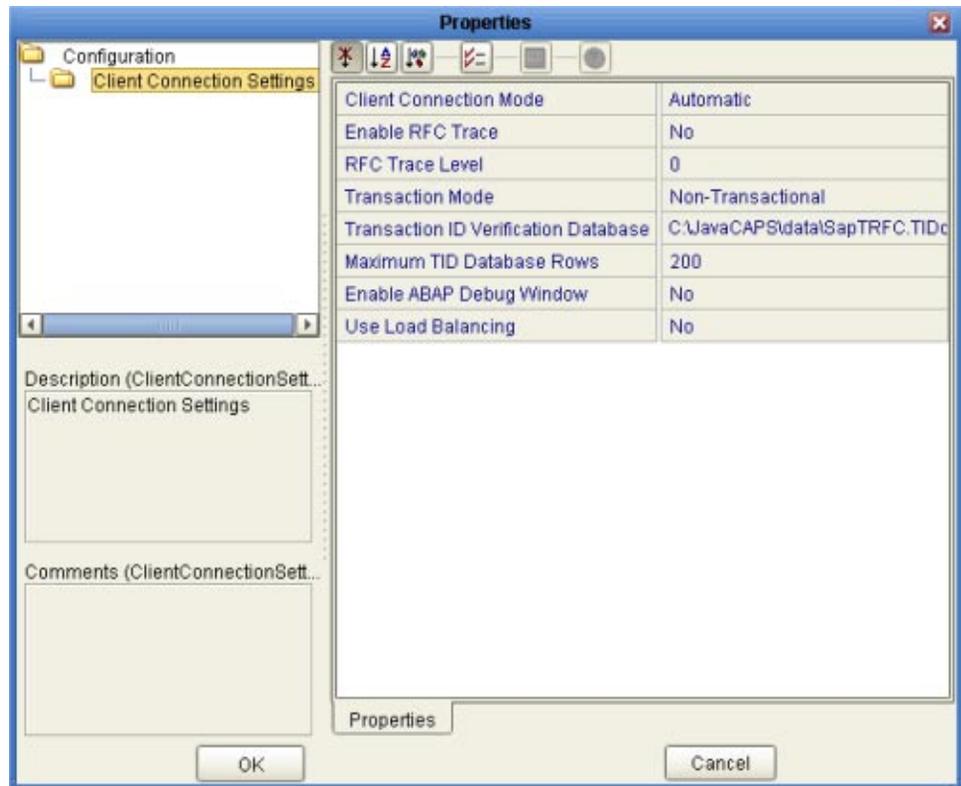
Inbound SAP BAPI Adapter Properties

The Inbound Adapter Properties include parameters required to receive data from SAP R/3 into Java CAPS. The following server connection settings are configured in the Inbound Adapter Properties window.

Server Connection Settings

The following Server Connection Settings are used by the external database:

TABLE 5   Inbound Adapter—Server Connection Settings

Name	Description	Required Value
Enable RFC Trace	You enable RFC tracing with the `Enable RFC Trace` property. The trace file contains RFC API calls, and data sent to and received from the SAP R/3 host. The trace file is `rfcnumber.trc`, for example, `rfc00310_0156.trc`.	Yes or No. The default mode is No; the RFC tracing is disabled. Setting the Enable RFC Trace parameter to Yes creates both the JCo and RFC Trace logs. Both are created in the same location under: `appserver\domains\domain1\ config` The JCoTrace log provides Java Runtime, version, and path information. It also provides a manifest. If Enable RFC Trace is set to No, then no trace file is generated.
RFC Trace Level	Trace level specifies the complexity of the information in the trace file. 0 provides minimal trace logging and 5 provides the maximum trace logging of diagnostic information in the trace file.	An integer value from 0 to 5. The default number is 0. The Enable RFC Trace level only affects the JCo trace level. It has no effect on the RFC trace level.
Number of RFC Servers to create	Specify the number of RFC servers to create. The created RFC servers facilitate parallel processing when receiving multiple requests from SAP R/3.	A valid integer value. The default number is **1**.
Transaction Mode	Specifies the transaction mode. The transaction mode specifies whether tRFC is enforced. With tRFC, transactions have unique TIDs and are processed only once by this Adapter.	A transaction mode of Non-Transactional or Transactional RFC(tRFC) The default mode is Non-Transactional.
Transaction ID Verification Database	Specifies the location of the Transaction ID Verification database. Specify the name of the file-based database which persists the TIDs. Provide the path to the database file that records the disposition of all transactions outgoing from this Adapter. The database records whether transactions are: ■ C (committed) ■ U (unprocessed or rolled-back) ■ R (reserved or pending)	A valid path to the database file. For example, the default location is: `C:\JavaCAPS6\data\SapTRFC.TIDdb`

Outbound SAP BAPI Adapter Properties

The Outbound Adapter Properties include parameters required to communicate from Java CAPS to SAP/R3. The following server connection settings are configured in the Outbound Adapter Properties window.

Client Connection Settings

The following Client Connection Settings are used by the outbound adapter:

TABLE 6   Outbound Adapter—Client Connection Settings

Name	Description	Required Value
Client Connection Mode	Determines the type of client connection to use when logging onto SAP R/3.	Automatic or Manual. The default mode is Automatic.
Enable RFC Trace	You enable RFC tracing with the Enable RFC Trace property. The trace file is rfc*number*.trc, for example, rfc00310_0156.trc.	Yes or No. The default mode is No; the RFC tracing is disabled. Setting the Enable RFC Trace parameter to Yes creates both the JCo and RFC Trace logs. Both are created in the same location under: appserver\domains\domain1\ config The JCoTrace log provides Java Runtime, version, and path information. It also provides a manifest. If Enable RFC Trace is set to No, then no trace file is generated.
RFC Trace Level	Trace level specifies the complexity of the information in the trace file. 0 provides minimal trace logging and 5 provides the maximum trace logging of diagnostic information in the trace file.	Integer value from 0 to 5. The default number is **0**. The Enable RFC Trace level only affects the JCo trace level. It has no effect on the RFC trace level.

TABLE 6 Outbound Adapter—Client Connection Settings   *(Continued)*

Name	Description	Required Value
Transaction Mode	Specifies the transaction mode. ■ **Non-Transactional**: Actions performed by BAPI call are committed immediately and automatically by SAP R/3 (auto commit). In this mode, use the execute( ) method in the OTD. ■ **Transactional RFC (tRFC)**: Adapter communicates with SAP R/3 using unique transaction IDs (TID) to avoid message repeats. Use the **executeAsynchronous(eid)** method in the OTD in this mode. ■ **VIA COMMIT/ROLLBACK BAPI**: Performs a single phase commit, where actions performed by BAPI calls are committed or rolled back by calling BAPI_TRANSACTION_COMMIT or BAPI_TRANSACTION_ROLLBACK. In this mode you must use the commit and rollback methods on the BAPI/RFC OTD.	A transaction mode of Non-Transactional or Transactional RFC(tRFC), or VIA COMMIT/ROLLBACK BAPI. The default mode is NON-TRANSACTIONAL.
Transaction ID Verification Database	Specifies the location of the Transaction ID Verification database. Specify the name of the file-based database which persists the TIDs. Provide the path to the database file that records the disposition of all transactions outgoing from this Adapter. The database records whether transactions are: ■ C (committed) ■ U (unprocessed or rolled-back) ■ R (reserved or pending)	A valid path to the database file. For example, the default location could be: C:\JavaCAPS6\data\SapTRFC.TIDdb
Maximum TID Database Rows	Specifies the maximum amount of rows for the Transaction ID (TID) database for outbound adapters. Set this property only if tRFC is used. This property specifies the maximum number of rows in the outbound TID database that are kept before the oldest rows are purged and their corresponding TIDs confirmed on SAP R/3. Confirmation allows SAP R/3 to remove those TIDs from its TID tracking database and reduce resource consumption.	At least 1 row. The default is 200 rows.

TABLE 6   Outbound Adapter—Client Connection Settings   *(Continued)*

Name	Description	Required Value
Enable ABAP Debug Window	Enables the ABAP debugging window. Enabling the `Enable ABAP Debug Window` property opens the ABAP debugging window on the Application Server. The window shows the debug information for the RFC-enabled ABAP application that is called by SAP R/3. This property only works if the SAPGUI software is installed on the Application Server.	Yes or No. The default mode is No; the ABAP Debug window is disabled.
Use Load Balancing	Enables load balancing for outbound adapters. This property allows you to take advantage of the workload balancing provided by SAP R/3. SAP R/3 provides workload balancing to automatically route requests to the SAP application server within a group of servers that has the best response time determined at that moment by an SAP message server. If you disable load balancing, use the `System number` property.	Yes or No. The default mode is No; load balancing is disabled by default.

Siebel EAI Adapter Connectivity Map Properties

All adapters contain a unique set of default configuration parameters. After the adapters are established and a Siebel EAI External System is created in the Project's Environment, the adapter parameters are modified for your specific system. The adapter property settings define how the properties are used to interact with the Siebel application.

- "Servlet Configuration (Inbound)" on page 15
- "Siebel EAI Settings (Outbound)" on page 16
- "HTTP Settings (Outbound)" on page 16

Servlet Configuration (Inbound)

Details for Siebel EAI settings are listed in the table below.

TABLE 7   Inbound Adapter—Servlet Configuration

Name	Description	Required Value
eWay URL Mapping	A pattern that determines the set of URIs that the servlet handles. This value must follow the rules specified in Section 10 of the Servlet API specification.	The default value that the servlet responds to is `/eaiservice/*`.

TABLE 7 Inbound Adapter—Servlet Configuration *(Continued)*

Name	Description	Required Value
Request Timeout	The length of time in milliseconds for the adapter to wait for a service invocation to complete before abandoning the request. If Immediate Acknowledgement (below) is set to false, this property is disregarded.	A number equal to or greater than 0. Zero indicates that the adapter will wait indefinitely.
Immediate Acknowledgement	An indicator of whether the adapter is responsible for composing responses to each request. When set to true, the adapter responds to each request itself instead of waiting for the outcome of the service invocation. When set to false, the responsibility is left to the service.	True or False.

Siebel EAI Settings (Outbound)

Details for Siebel EAI settings are listed in the table below.

TABLE 8 Outbound Adapter—Siebel EAI Settings

Name	Description	Required Value
SWEExtSource	Specifies the service that the Siebel Web Engine calls.	A valid string. The default is SEEBEYOND_HTTP_UPDATE. Other values may include: ■ **SEEBEYOND_HTTP_DELETE** ■ **SEEBEYOND_HTTP_EXECUTE** ■ **SEEBEYOND_HTTP_QUERY** ■ **SEEBEYOND_HTTP_UPDATE**
SWEExtCmd	Specifies the external command.	A valid entry. The default is Execute.

HTTP Settings (Outbound)

Details for HTTP settings are listed in the table below.

TABLE 9 Outbound Adapter—HTTP Settings

Name	Description	Required Value
Allow cookies	Specifies whether or not cookies sent from servers is stored and sent on subsequent requests. If cookies are not allowed, then sessions are not supported.	True or False. The default is True.
Content type	Specifies the default Content-Type header value to include when sending a request to the server.	
Accept type	Specifies the parameters for the Accept Type request header.	Any valid string. For example: ■ text/html ■ text/plain ■ text/xml
Encoding	Specifies the default encoding used when reading or writing textual data.	ASCII.

SWIFT Alliance Gateway Adapter Connectivity Map Properties

All adapters contain a set of properties that are unique to that adapter type. After the adapters are created and a SWIFTAlliance Gateway External System is added to the Project's Environment, the adapter parameters can be modified for your specific system.

Note – Modifying individual OTD configuration settings can override the default adapter OTD configuration settings. For more information on SWIFTAlliance Gateway configuration properties in the context of SWIFTAlliance Gateway, refer to the SWIFTAlliance Gateway and SNL user documentation provided by SWIFT.

Selecting SWIFTAlliance Gateway as the External Application

To create an SWIFTAlliance Gateway adapter, you must first create an SAG (SWIFTAlliance Gateway) External Application in your Connectivity Map. SWIFTAlliance Gateway adapters are located between the SAG External Application and a Service. Services are containers for Collaborations, Business Processes, and so forth.

▼ To Create the SWIFT AG External Application

1 From the Connectivity Map toolbar, click the External Applications icon.

2 **Select the SAG External Application from the menu (see the figure below). The selected SAG External Application icon appears on the Connectivity Map toolbar.**
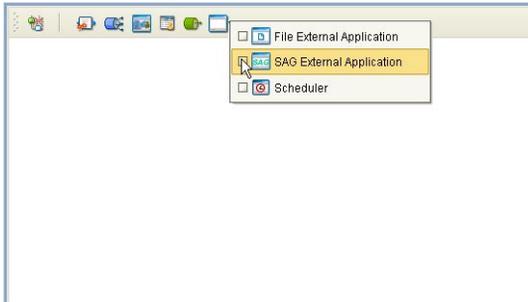
3 **Drag the new SAG External Application from the toolbar onto the Connectivity Map canvas. This represents an external SWIFTAlliance Gateway system.**

From the Connectivity Map, you can associate (bind) the External Application with the Service to establish an adapter (see the figure below).

FIGURE 2 Adapter Location

When SAG is selected as the External Application, it automatically applies the default SWIFT Alliance Gateway Adapter properties, provided by the OTD, to the adapter that connects it to the Service. You can then modify these properties for your specific system using the Properties Editor.

SWIFT AG Adapter Connectivity Map Properties

The SWIFTAlliance Gateway Adapter Connectivity Map Properties include the following parameter sections:

- "SWIFT AG Envelope Properties" on page 19
- "SWIFT AG Primitive Control Properties" on page 19
- "SWIFT AG RemoteApi Base Settings" on page 20
- "SWIFT AG InterAct Client Properties" on page 21
- "SWIFT AG InterAct Client > Store and Forward Properties" on page 22
- "SWIFT AG FileAct Client Properties" on page 23
- "SWIFT AG FileAct Client > Store and Forward Properties" on page 26
- "SWIFT AG FileAct Client > Get File Properties" on page 27
- "SWIFT AG FileAct Client > Put File Properties" on page 28
- "SWIFT AG FileAct Client > SnF Fetch File Properties" on page 29
- "SWIFT AG Connection Establishment Properties" on page 29

SWIFT AG Envelope Properties

The Envelope section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 10   Connectivity Map - Envelope

Name	Description	Required Value
Application ID	Specifies the name of the Application Interface Message Partner. The Message Partner basically identifies an application that sends and receives messages on behalf of a user.	The name of the Application Interface Message Partner. This property is mandatory
Context ID	Specifies the cryptographic mode for Relaxed SNL Protocol: ■ Empty (blank) indicates automatic mode. ■ Advanced: indicates advanced mode. Relaxed SNL Protocol does not support manual cryptographic mode.	Advanced or blank indicating the selected Context ID. The configured default is Advanced
Msg Format	Specifies the format of the current message. Each of the names of the required values denotes a SWIFT Alliance Gateway message format.	The default value is Sag:RelaxedSNL. **Note –** This property is "grayed out" to indicate that the property is not configurable.
Sender	Specifies the name of the SWIFT Alliance Gateway Sender. This is an SAG operator.	The name of the sender.
Sender Auth	Specifies the password of the SAG operator.	The SAG operator password.

SWIFT AG Primitive Control Properties

The Primitive Control section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 11 Connectivity Map - Primitive Control

Name	Description	Required Value

Include XML Attributes in SNL Primitive	Specifies the construction/marshalling of SNL primitives.
	
	When the SWIFT AG Adapter constructs/marshals the SNL primitives, this flag indicates whether the adapter includes the XML attributes in the primitives. For example:
	■ For element SwInt:Requestor, including XML attributes, may be as follows:
	`<SwInt:Requestor type="Sw.Gbl.DN"`
	`version="4.0.0">o=swift,o=swift`
	`</SwInt:Requestor>`
	
	■ If you do not include XML attributes, it may be as follows:
	
	`<SwInt:Requestor>o=swift,o=swift </SwInt:Requestor>`
	`Note that all XML attributes in SNL`
	`Primitives are defined in SNL`
	`Specification.`
		Select True or False. True indicates that the adapter includes the XML attributes in the primitives.
		
		The configured default is False.

SWIFT AG RemoteApi Base Settings

The RemoteApi Base settings section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 12 Connectivity Map - RemoteApi Base settings

Name	Description	Required Value
Client Handle Timeout	Specifies the maximum time (in milliseconds) allowed between a request message and its corresponding response message.	An integer indicating the maximum time allowed between a request message and its corresponding response message in milliseconds (for example, 300000 equals 5 minutes). The configured default is 300000.
Read Blocking Timeout	Specifies the maximum time (in milliseconds) to block a read operation	An integer indicating the maximum time in milliseconds to block a read operation (for example, 60000 equals 1 minute). The configured default is 60000.

SWIFT AG InterAct Client Properties

This section maps to Primitive SwInt:ExchangeRequest. For parameters SwXXX under this section, refer to the SNL specification for more detailed descriptions. The InterAct Client section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 13 Connectivity Map - InterAct Client

Name	Description	Required Value
User DN	Specifies the SwSec_UserDN, full distinguished name used to identify and authenticate the principal. Size is limited to 100 bytes.	The User DN (distinguished name) ending with o=<BIC8>,o=swift For example: `cn=dandrews,o=bankn14b,o=swift`
Requestor DN	Specifies the SwInt_Requestor, used to identify the Application entity name. Size is limited to 100 bytes.	The Requestor DN (distinguished name) ending with o=<SWIFTNet institution BIC-8>,o=swift For example: `ou=management,o=bankn14b,o=swift`
Responder DN	Specifies the SwInt_Responder, used to identify the intended responder. Size is limited to 100 bytes.	The Responder DN (distingusihed name) ending with o=<SWIFTNet institution BIC-8>,o=swift For example: `cn=management,o=swift,o=swift`
Service Name	Specifies the SwInt_Service, the Service Name containing the SWIFTNet service used. The size is limited to 30 bytes.	The Service Name. For example, swift.cte, swift.generic.ia!x, swift.generic.iast!x, and so forth.
Request Type	Specifies the SwInt_RequestType, used to identify the message type of the XML message using the standard message code. The size is limited to 30 bytes.	The Request Type. For example: camt.005.001.02 (GetTransaction)
User Reference	Specifies the SwInt_RequestRef. This is used to associate a request message with subsequent response or error messages. The size is limited to 30 bytes.	A user reference of 30 bytes or less.
Signed?	Specifies whether the request contains Crypto operations to be performed. Only the last Crypto block is analyzed. SwInt_RequestCrypto (digital signature).	Select True or False. True indicates that the request contains Crypto operations to be performed. The configured default is FALSE.

TABLE 13   Connectivity Map - InterAct Client   *(Continued)*

Name	Description	Required Value
Priority	Specifies the SwInt_Priority. The priority of delivery. In the future, SWIFTNet may implement this priority through schemes such as top-queuing, dedicated server processes, network transport priority. In the current implementation, differentiation on priority may be used for store-and-forward delivery.	Normal or Urgent. The configured default is Normal.
Non-Repudiation?	Specifies whether SwInt_NRIndicator, nonrepudiation support is being requested.	Select True or False. True indicates that nonrepudiation is requested. The configured default is FALSE.
Delivery Notification Queue Name	Specifies the SwInt_NotifQueue, store-and-forward delivery mode. When a value is present, it indicates SnF delivery mode and the queue where SnF delivery notifications are received. In the case of non-delivery (Rejected or Failed message), an SnF failed notification is always generated in this notification queue. In the case of delivery (Accepted or Duplicated message) an SnF delivery notification is optionally generated in this queue.	For store-and-forward delivery mode, enter the name of the delivery notification queue. Size is limited to 30 bytes. For example: ptsauszz_generic!x. (see Sw:DeliveryNotif)
Ask Positive Delivery Notification?	Specifies Sw_DeliveryNotif, delivery notification. This is for store-and-forward delivery mode only. Indicates whether a delivery notification is required in case of successful delivery (Accepted or Duplicated).	Select True or False. True indicates that a delivery notification is required. The configured default is FALSE.

SWIFT AG InterAct Client > Store and Forward Properties

This section maps to the parameter Sw:AcquireSnFRequest of the Primitive Sw:ExchangeSnFRequest. For parameters SwXXX under this section, refer to the SNL specification for more detailed descriptions. The InterAct Client > Store and Forward section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

Note – For more information on SWIFT Alliance Gateway configuration properties in the context of SWIFT Alliance Gateway, refer to the SWIFT Alliance Gateway and SNL user documentation provided by SWIFT.

Connectivity Map - InterAct Client > Store and Forward

Name	Description	Required Value
Queue Name To Acquire	Specifies the name of the queue (SwInt_Queue).	The User SwInt_Queue For example: ptsauszz_generic!x Size is limited to 30 bytes.
Force Acquire?	Specifies the Sw_ForceAcquire. Indicates whether an acquisition request must be accepted in case the queue is already acquired.	Select TRUE or FALSE. True indicates that an acquisition request must be accepted. The configured default is TRUE.
Session Mode	Specifies the usage mode of the queue. This property is purposely disabled.	The default setting is **Pull**. Pull is the correct setting for *client* mode **Note** – This property is purposely disabled.
Order By	Specifies the Sw_OrderBy. Indicates the order priority in which the messages are retrieved on the queue. The options are: ■ InterAct ■ Blank: indicates FIFO (first in, first out). ■ FileAct ■ Urgent	Select InterAct, blank, FileAct, or Urgent. The configured default is InterAct.
Recovery Mode?	Specifies the Sw_RecoveryMode. Indicates whether the session must be opened in recovery mode. Messages that already have an output sequence number are sent first, before considering the selected order.	Select TRUE or FALSE. True indicates that the session must be opened in recovery mode. The configured default is FALSE.

SWIFT AG FileAct Client Properties

This section maps to the Primitive Sw:ExchangeFileRequest. For parameters SwXXX under this section, refer to the SNL specification for more detailed descriptions. The FileAct Client section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 15   Connectivity Map - FileAct Client

Name	Description	Required Value
User DN	Specifies the SwSec_UserDN. Used to identify and authenticate the principal. Size is limited to 100 bytes.	The User DN (distinguished name) ending with o=<BIC8>,o=swift For example: cn=dandrews,o=bankn14b,o=swift

TABLE 15 Connectivity Map - FileAct Client   *(Continued)*

Name	Description	Required Value
Requestor DN	Specifies the SwInt_Requestor, the Application entity name. Size is limited to 100 bytes.	The Requestor DN (distinguished name) ending with o=<SWIFTNet institution BIC-8>,o=swift For example: `ou=management,o=bankn14b,o=swift`
Responder DN	Specifies the SwInt_Responder, the name of the intended responder. Size is limited to 100 bytes.	The Responder DN (distingusihed name) ending with o=<SWIFTNet institution BIC-8>,o=swift For example: `cn=management,o=swift,o=swift`
Service Name	Specifies the SwInt_Service, the Service Name containing the SWIFTNet service used. The size is limited to 30 bytes.	The Service Name. For example, swift.cte, swift.generic.fa!x, swift.generic.fast!x, and so forth.
Request Type	Specifies the SwInt_RequestType, used to identify the message type of the XML message using the standard message code. The size is limited to 30 bytes.	The Request Type. For example: camt.005.001.02 (GetTransaction)
User Reference	Specifies the SwInt_RequestRef. This is used to associate a request message with subsequent response or error messages. The size is limited to 30 bytes.	A user reference of 30 bytes or less.
Signed?	Specifies whether the request contains Crypto operations to be performed. Only the last Crypto block is analyzed. SwInt_RequestCrypto (digital signature).	Select True or False. True indicates that the request contains Crypto operations to be performed. The configured default is FALSE.
Priority	Specifies the SwInt_Priority. The priority of delivery. In the future, SWIFTNet may implement this priority through schemes such as top-queuing, dedicated server processes, network transport priority. In the current implementation, differentiation on priority may be used for store-and-forward delivery.	Normal or Urgent. The configured default is Normal.
Non-Repudiation?	Specifies whether SwInt_NRIndicator, nonrepudiation support is being requested.	Select True or False. True indicates that nonrepudiation is requested. The configured default is FALSE.

TABLE 15 Connectivity Map - FileAct Client   *(Continued)*

Name	Description	Required Value
Delivery Notification Queue Name	Specifies the SwInt_NotifQueue, store-and-forward delivery mode. When a value is present, it indicates SnF delivery mode and the queue where SnF delivery notifications are received. In the case of non-delivery (Rejected or Failed message), an SnF failed notification is always generated in this notification queue. In the case of delivery (Accepted or Duplicated message) an SnF delivery notification is optionally generated in this queue.	For store-and-forward delivery mode, enter the name of the delivery notification queue. Size is limited to 30 bytes. For example: ptsauszz_generic!x. (see Sw:DeliveryNotif)
Ask Positive Delivery Notification?	Specifies Sw_DeliveryNotif, delivery notification. This is for store-and-forward delivery mode only. Indicates whether a delivery notification is required in case of successful delivery (Accepted or Duplicated).	Select True or False. True indicates that a delivery notification is required. The configured default is FALSE.

TABLE 15   Connectivity Map - FileAct Client   *(Continued)*

Name	Description	Required Value

Remote File Handler TransferEP	Specifies the Sw_TransferEP, the Transfer EndPoint name used for the remote file handler. Value options:
	■ Blank value: indicates that the file transfer does not use the remote file handler.
	
	■ Transfer EndPoint name: When this value is present, the file transfer uses the remote file handler.
	Before the file transfer starts you need to start the remote file handler process with the specified Transfer EndPoint name on your system. The command to start the remote file handler (swfa_handler) requires the command-line arguments:
	`swfa_handler`
	`<HostName>:<PortNumber>[:ssl]`
	`<TransferEndpoint> [<Process ID>]`
	Here are some examples:
	`swfa_handler snlhost:48003:ssl`
	`MyUniqueEndpoint 23450`
	`swfa_handler snlhost:48003`
	`MyUniqueEndpoint 23450`
	`swfa_handler snlhost:48003`
	`MyUniqueEndpoint`
	As for the syntax details and operational guidelines of remote file handler, refer to the SWIFTNet Service Design Guide or consult your system person.
		The Transfer EndPoint name used for the remote file handler.
		
		Size is limited to 30 bytes.

Block File Transfer?	Specifies BlockFileTransfer. This indicates whether the function call will finish when a final file status is returned. This may be useful for the large file transfers. The final status values for file transfer are: Completed, Duplicated, Aborted, Failed, Rejected, and Unknown.	Select True or False. True indicates that the function call will finish when a final file status is returned. The configured default is True.

SWIFT AG FileAct Client > Store and Forward Properties

This section maps to parameter Sw:AcquireSnFRequest of Primitive Sw:ExchangeSnFRequest. For parameters SwXXX under this section, refer to the SNL specification for more detailed descriptions. The InterAct Client > Store and Forward section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 16   Connectivity Map - FileAct Client > Store and Forward

Name	Description	Required Value
Queue Name To Acquire	Specifies the SwInt_Queue. For example, ptsauszz_generic!x. Size is limited to 30 bytes.	The User SwInt_Queue For example: ptsauszz_generic!x
Force Acquire?	Specifies the Sw_ForceAcquire. Indicates whether an acquisition request must be accepted if a queue is already acquired.	Select TRUE or FALSE. True indicates that an acquisition request must be accepted. The configured default is TRUE.
Session Mode	Specifies the usage mode of the queue. This property is purposely disabled.	The default setting is Pull. Pull is the correct setting for *client* mode **Note** – This property is purposely disabled.
Order By	Specifies the Sw_OrderBy. Indicates the order priority in which the messages are retrieved on the queue. The options are: ■ InterAct ■ Blank: indicates FIFO (first in, first out). ■ FileAct ■ Urgent	Select InterAct, blank, FileAct, or Urgent. The configured default is FileAct.
Recovery Mode?	Specifies the Sw_RecoveryMode. Indicates whether the session must be opened in recovery mode. Messages that already have an output sequence number are sent first, before considering the selected order.	Select True or False. True indicates that the session must be opened in recovery mode. The configured default is False.

SWIFT AG FileAct Client > Get File Properties

This section maps to the parameter Sw:GetFileRequest of the Primitive Sw:ExchangeFileRequest. For parameters SwXXX under this section, refer to the SNL Specification for more detailed descriptions. The FileAct Client > Get File section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 17   Connectivity Map - FileAct Client > Get File

Name	Description	Required Value
Transfer Description	Specifies the Sw_TransferDescription, user information about the file transfer. Free text. Size is limited to 256 bytes.	The transfer description.

TABLE 17 Connectivity Map - FileAct Client > Get File   *(Continued)*

Name	Description	Required Value
Transfer Info	Specifies the Sw_TransferInfo, user information about the file transfer. Structured data that can be analyzed by the server. Size is limited to 256 bytes.	The transfer information.
Logical File Name	Specifies the Sw_LogicalName, the logical name of the file to get. This name is communicated to the server application. By default, this name is the physical name without path. Size is limited to 254 bytes.	The logical file name.
Physical File Name	Specifies the Sw_PhysicalName, the full physical name where the file to get must be stored. If the file already exists it is overwritten. Size is limited to 254 bytes.	The physical file name.
Maximum File Size	Specifies the Sw_MaxSize, the maximum accepted file size. When a value is present, the maximum size is a field that is communicated from the client application to the server application. This property is optional. Size range is 0 to 250 Megabytes.	The maximum file size.

SWIFT AG FileAct Client > Put File Properties

This section maps to parameter Sw:PutFileRequest of the Primitive Sw:ExchangeFileRequest. For parameters SwXXX under this section, refer to the SNL Specification for more detailed descriptions. The FileAct Client > Put File section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 18 Connectivity Map - FileAct Client > Put File

Name	Description	Required Value
Transfer Description	Specifies the Sw_TransferDescription, This is information about the file transfer. Free text. Size is limited to 256 bytes.	The transfer description.
Transfer Info	Specifies the Sw_TransferInfo, information about the file transfer. This is structured data that can be analysed by the server. Size is limited to 256 bytes.	The transfer information.
Logical File Name	Specifies the Sw_LogicalName, the logical name of the file to put. This name is communicated to the server application. By default, this name is the physical name without path. Size is limited to 254 bytes.	The logical file name.
Physical File Name	Specifies the Sw_PhysicalName, the full physical name of the file to put. Size is limited to 254 bytes.	The physical file name.

TABLE 18 Connectivity Map - FileAct Client > Put File   *(Continued)*

Name	Description	Required Value
File Description	Specifies the Sw_FileDescription. This is user information about the file. Free text. Size is limited to 256 bytes.	The file description.
File Info	Specifies the Sw_FileInfo, user information about the file. This is structured data that can be analysed by the server. Size is limited to 256 bytes.	The file information.

SWIFT AG FileAct Client > SnF Fetch File Properties

This section maps to Primitive Sw:FetchFileRequest. For parameters SwXXX under this section, refer to the SNL Specification for more detailed descriptions. The FileAct Client > SnF Fetch File section of the SWIFT AG Adapter Connectivity Map properties contains the top-level parameters displayed in the table below.

TABLE 19 Connectivity Map - FileAct Client > SnF Fetch File

Name	Description	Required Value
Physical File Name	Specifies the Sw_PhysicalName, the full physical name where the file is stored locally. Size is limited to 254 bytes.	The physical file name.

SWIFT AG Connection Establishment Properties

The Connection Establishment section of the SWIFT AG Adapter Connectivity Map properties defines configuration parameters used to control the connection establishment. It contains the top-level parameters displayed in the table below.

TABLE 20 Connectivity Map - Connection Establishment

Name	Description	Required Value
Always Create New Connection	Specifies whether to ALWAYS try to create a new connection for a connection establishment request. The options are: ■ **True**: a new connection is always created without trying to match an existing connection. ■ **False**: an attempt to match an existing connection (managed by container) is made.	Select True or False. The configured default is False.

TABLE 20 Connectivity Map - Connection Establishment   *(Continued)*

Name	Description	Required Value
Auto Disconnect Connection	Specifies whether the adapter closes the connection automatically after the work is finished on the connection. ■ **True**: the connection is not re-used. ■ **False**: the connection is returned to the pool for reuse.	Select True or False. The configured default is False.

WebSphere MQ Adapter Connectivity Map Properties

The WebSphere MQ Adapter's Properties are organized as follows:

Note – Creating customized individual OTD configuration settings can override the default Adapter OTD configuration settings.

Inbound WebSphere MQ Adapter Connectivity Map Properties

The inbound WebSphere MQ Adapter parameters, accessed from the Connectivity Map, are organized into the following sections:

Inbound WebSphere MQ Adapter Settings

The Inbound WebSphere MQ Adapter Settings section of the WebSphere MQ Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 21 Connectivity Map - Inbound WebSphere MQ Adapter Settings

Name	Description	Required Value
Queue Name	Specifies the name of the queue from which messages are picked up (subscribed). **Note –** Inbound mode Adapters will not function if a non-local queue is specified. Non-local queues include alias queues and local queue definitions to remote queues. This limitation applies only applies to polling/receive mode Adapters, not outbound Adapters used to retrieve messages.	The name of the WebSphere MQ queue. The configured default is default.
Maximum Message Size	Specifies the maximum message size that the Adapter is able to get from a queue. A value of zero (0) instructs the Adapter to use the MQ-provided default size. If you specify a non-zero value, and a message on the queue is larger than this value, one of the following will occur: If the MQC.MQGMO_ACCEPT_TRUNCATED_MSG property is set to True, the Adapter processes as much of the message from the queue as possible, producing a truncated message. If the MQC.MQGMO_ACCEPT_TRUNCATED_MSG property is set to False, the Adapter leaves the message on the queue and raises an MQException with a completion code of MQCC_WARNING, and with a reason code of MQRC_TRUNCATED_MSG_FAILED.	A number indicating the maximum message size in bytes. The configured default is 0 (use the MQ defined default size).
Schedule Interval	Specifies the polling interval in milliseconds at which the subscribed queue is polled for messages. This is the duration of the pause, in milliseconds, between attempts to get messages from the queue.	The number of milliseconds at which the queue is polled. The configured default is 5000 (or 5 seconds).
Security Exit	Specifies the optional, package qualified name of a user-defined class that implements the com.ibm.mq.MQSecurityExit interface. WebSphere MQ invokes an instance of the class whenever the Adapter attempts to connect to the queue manager. The named class must include a default constructor. This parameter is only used for client connections. Security Exits are not applicable to bindings connections. For more information about Security Exits, see the IBM document, "WebSphere MQ Using Java" (CSQZAW09) regarding MQSecurityExit.	The name of the user-defined class. This property also requires an assigned value for the Security Exit Jar Classpath property.

TABLE 21 Connectivity Map - Inbound WebSphere MQ Adapter Settings *(Continued)*

Name	Description	Required Value
Security Exit JAR Classpath	Specifies the absolute path to the JAR file that contains the named Security Exit. This property is required if the Security Exit is specified. The specified JAR is packaged into the application (EAR) file that is generated during Project activation. If the specified JAR cannot be accessed or found, the activation will fail. If this property value is left blank, you must ensure that a JAR file containing the Security Exit is made accessible to the runtime Environment prior deploying the Project (for example, by manually copying the JAR file into the Integration Server's lib directory prior to or during the Deployment Process). For more information about Security Exits, see the IBM document, "WebSphere MQ Using Java" (CSQZAW09) regarding MQSecurityExit.	The absolute path for the JAR file that contains the named Security Exit class.

Inbound WebSphere MQ Adapter Settings > GetMessageOptions > matchOptions

The Inbound Adapter Settings > GetMessageOptions > matchOptions section of the WebSphere MQ Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 22 Connectivity Map - Inbound WebSphere MQ Adapter Settings > GetMessageOptions > matchOptions

Name	Description	Required Value
correlationId	Specifies the correlation identifier of the message to be retrieved. Normally the queue manager returns the first message with a message identifier and correlation identifier that matches the identifiers specified.	The correlation identifier of the message.
groupId	Specifies the byte string that identifies the message group to which the physical message belongs.	A byte string that indicates the message group.

TABLE 22 Connectivity Map - Inbound WebSphere MQ Adapter Settings > GetMessageOptions > matchOptions   *(Continued)*

Name	Description	Required Value
messageId	For an MQGET call, this field specifies the message identifier of the message to be retrieved. Normally, the queue manager returns the first message with a message identifier and correlation identifier that matches those identifiers specified. For an MQPUT call, this specifies the message identifier to use.	The message identifier.
messageSequenceNumber	Specifies the sequence number of a logical message within a group.	The sequence number of the logical message within a group. The configured default is **1**.
MQMO_MATCH_CORREL_ID	Specifies that the retrieved message must have a correlation identifier that matches the value of the correlationId parameter. The values are: ■ True: Indicates that the message must have a matching correlation identifier. ■ False: Indicates that the correlation identifier is ignored and any correlation identifier will be accepted. This match is in addition to any other matches that may apply (for example, the message identifier).	True or False. The configured default is False.
MQMO_MATCH_GROUP_ID	Specifies that the retrieved message must have a group identifier that matches the value of the groupId parameter. The values are: ■ True: Indicates that the message must have a matching group identifier. ■ False: Indicates that the group identifier is ignored and any group identifier is accepted. This match is in addition to any other matches that may apply (for example, the correlation identifier).	True or False. The configured default is False.

TABLE 22 Connectivity Map - Inbound WebSphere MQ Adapter Settings > GetMessageOptions > matchOptions   *(Continued)*

Name	Description	Required Value
MQMO_MATCH_MSG_ID	Specifies that the retrieved message must have a message identifier that matches the value of the messageId parameter. The values are: ■ True: Indicates that the message must have a matching message identifier. ■ False: Indicates that the message identifier is ignored and any message identifier is accepted. This match is in addition to any other matches that may apply (for example, the correlation identifier).	True or False. The configured default is False.
MQMO_MATCH_MSG_ SEQ_NUMBER	Specifies that the retrieved message must have a message sequence number that matches the value of the messageSequenceNumber parameter. The values are: ■ True: Indicates that the message must have a matching message sequence number. ■ False: Indicates that the message sequence number is ignored and any message sequence number is accepted. This match is in addition to any other matches that may apply (for example, the group identifier).	True or False. The configured default is False.
MQMO_NONE	Specifies that no matches are to be used in selecting the message to be returned. All messages on the queue are eligible for retrieval (subject to some MQGMO_ options).	True or False. The configured default is True.

Inbound WebSphere MQ Adapter Settings > GetMessageOptions > options

The Inbound Adapter Settings > GetMessageOptions > options section of the WebSphere MQ Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 23   Connectivity Map - Inbound Adapter Settings > GetMessageOptions > options

Name	Description	Required Value
MQGMO_ACCEPT_ TRUNCATED_MSG	Specifies whether a truncated message is accepted as a complete message. If the message buffer is too small to hold the complete message, this option allows the MQGET call to fill the buffer with as much as it can hold and complete its processing. Without this option, in the given situation, the MQGET call will still be filled to capacity, but the processing will not be considered completed. The values are: ■ **True**: Indicates that a truncated message is accepted as a complete message. ■ **False**: Indicates that a truncated message is not considered as a complete message.	True or False. The configured default is True.
MQGMO_FAIL_IF_ QUIESCING	Forces the MQGET call to fail if the queue manager is in the quiescing state. The values are: ■ **True**: Indicates that calling MQGET fails if the queue manager is in the quiescing state. ■ **False**: Indicates that calling MQGET does not fail if the queue manager is in the quiescing state.	True or False. The configured default is True.
MQGMO_SYNCPOINT	Forces the MQGET call to get the message under syncpoint control; the message is marked as being unavailable to other applications, but it is deleted from the queue only when the unit of work is committed. The message is made available again if the unit of work is backed out. The values are: ■ **True**: Indicates that calling MQGET gets the message under syncpoint control. ■ **False**: Indicates that MQGET, when called does not get the message under syncpoint control.	True or False. The configured default is False.

TABLE 23   Connectivity Map - Inbound Adapter Settings > GetMessageOptions > options   *(Continued)*

Name	Description	Required Value
MQGMO_SYNCPOINT_IF_PERSISTENT	Forces the MQGET call to get the message under syncpoint control if the message is persistent. The values are: ■ **True**: Indicates that calling MQGET gets the message under syncpoint control if the message is persistent. ■ **False**: Indicates that MQGET, when called does not get the message under syncpoint control if the message is persistent.	True or False. The configured default is False.
MQGMO_COMPLETE_MSG	Specifies that only a complete logical message can be returned by calling MQGET. If the logical message is segmented, the queue manager reassembles the segments and returns the complete logical message to the application; the fact that the logical message was segmented is not apparent to the Adapter. The values are: ■ **True**: Indicates that only a complete logical message can be returned by calling MQGET. ■ **False**: Indicates that a complete logical message is not required.	True or False. The configured default is False.
MQGMO_WAIT	Specifies that an MQ GET call waits (block/suspend) until a message becomes available in the queue. The values are: ■ **True**: Indicates that an MQ GET call waits until a message becomes available in the queue. ■ **False**: Indicates that an MQ GET call does not wait until a message becomes available in the queue.	True or False. The configured default is False.

TABLE 23   Connectivity Map - Inbound Adapter Settings > GetMessageOptions > options   *(Continued)*

Name	Description	Required Value
waitInterval	Specifies how long (in milliseconds) an MQ GET call waits for a message to become available in the queue. This parameter is used in conjunction with MQGMO_WAIT. If MQGMO_WAIT is set to false, waitInterval is not used. Specifying a negative value indicates that the wait will last indefinitely. Setting this value to a negative number causes the polling Adapter to execute MQ GET calls with a wait interval of MQWI_UNLIMITED. With this type of get call, the Adapter will block indefinitely until a suitable message is available. If the Integration Server (in association with the Application Server) is commanded to shut down or restart while the Adapter is still blocked, the Integration Server will not be able to proceed until the Adapter is unblocked by the availability of a suitable MQ message. The same limitation affects the non-polling use of the Adapter. The WebSphere MQ Adapter's OTD GMO structure exposes a method named setUnlimitedWait() to Java Collaborations that, when used, sets the waitInterval to the value MQWI_UNLIMITED. If using setUnlimitedWait() causes the Adapter to block indefinitely during a subsequent get call, the Integration Server will be unable to shut down until the Adapter is unblocked.	A number indicating the period of time, in milliseconds, that an MQ GET call waits for a message to become available in the queue. The configured default is 0.

Outbound WebSphere MQ Adapter XA Connectivity Map Properties

The outbound WebSphere MQ Adapter XA parameters, accessed from the Connectivity Map, are contained in the following section.

Outbound WebSphere MQ Adapter (XA) Settings

The Outbound Adapter (XA) Settings section of the WebSphere MQ Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 24   Connectivity Map - Outbound Adapter Settings

Name	Description	Required Value
Queue Name	Specifies the name of queue to which the message is published. This parameter is optional. The queue name may also be specified manually in the Business Process or Collaboration that effects the put.	The queue name to which the message is retrieved or published.
Security Exit	Specifies the optional, package qualified name of a user-defined class that implements the com.ibm.mq.MQSecurityExit interface. WebSphere MQ invokes an instance of the class whenever the Adapter attempts to connect to the queue manager. The named class must include a default constructor. This parameter is only used for client connections. Security Exits are not applicable to bindings connections. For more information about Security Exits, see the IBM document, "WebSphere MQ Using Java" (CSQZAW09) regarding MQSecurityExit.	The name of the user-defined class. This property also requires an assigned value for the *Security Exit JAR Classpath* property.

TABLE 24   Connectivity Map - Outbound Adapter Settings   *(Continued)*

Name	Description	Required Value
Security Exit JAR Classpath	Specifies the absolute path to the JAR file that contains the named Security Exit. This property is required if the Security Exit is specified. The specified JAR is packaged into the application (EAR) file that is generated during Project activation. If the specified JAR cannot be accessed or found, the activation will fail. If this property value is left blank, you must ensure that a JAR file containing the Security Exit is made accessible to the runtime Environment prior deploying the Project (for example, by manually copying the JAR file into the Integration Server's lib directory prior to or during the Deployment Process). For more information about Security Exits, see the IBM document, "WebSphere MQ Using Java" (CSQZAW09) regarding MQSecurityExit.	The absolute path for the JAR file that contains the named Security Exit class.

Outbound WebSphere MQ Adapter (XA) Settings > Queue Open Options

The Outbound Adapter Settings > Queue Open Options section of the WebSphere MQ Adapter Connectivity Map properties contains the top-level parameters displayed in the following table. These properties apply specifically to the MQOPEN calls.

TABLE 25   Connectivity Map - Outbound Adapter Settings > Queue Open Options

Name	Description	Required Value
MQOO_ALTERNATE_ USER_AUTHORITY	Specifies whether an alternate user identifier is used to check the authorization for the open. **True** indicates that and alternate user identifier is used.	True or False. The configured default is False.
MQOO_BIND_AS_Q_DEF	Specifies whether the binding used for the queue handle is taken from the DefBind queue attribute. **True** indicates that the binding used is taken from DefBind queue attribute.	True or False. The configured default is False.

TABLE 25   Connectivity Map - Outbound Adapter Settings > Queue Open Options   *(Continued)*

Name	Description	Required Value
MQOO_BIND_NOT_FIXED	Specifies whether the local queue manager binds the queue handle to a particular instance of the destination queue, when the object being opened is a cluster queue. **True** indicates that the local queue manager *will* bind to a specific destination. This option is ignored when specified for a queue that is not a cluster queue.	True or False. The configured default is False.
MQOO_BIND_ON_OPEN	Specifies whether the local queue manager binds the queue handle to a particular instance of the destination queue, when the object being opened is a cluster queue. **True** indicates that the local queue manager *will not* bind to a specific destination. This option is ignored when specified for a queue that is not a cluster queue.	True or False. The configured default is False.
MQOO_BROWSE	Specifies whether the queue is opened to browse messages. **True** indicates that the queue is open for use with MQGET calls with the following options: **MQGMO_BROWSE_FIRST** **MQGMO_BROWSE_NEXT** **MQGMO_BROWSE_MSG_UNDER_CURSOR**	True or False. The configured default is False.
MQOO_FAIL_IF_QUIESCING	Specifies whether the MQOPEN call fails when the queue manager is in quiescing state. Used to control authorization checking. **True** indicates that the MQOPEN call will fail if queue manager is quiescing.	True or False. The configured default is False.
MQOO_INPUT_AS_Q_DEF	Specifies whether the queue is opened to browse messages using the queue-defined default. **True** indicates that the queue is open for use with subsequent MQGET calls. **Note –** The value of this parameter is ignored when the Adapter is operating in automatic connection mode, because the Adapter must be capable of both receiving and sending messages.	True or False. The configured default is True.

TABLE 25   Connectivity Map - Outbound Adapter Settings > Queue Open Options   *(Continued)*

Name	Description	Required Value
MQOO_INPUT_ EXCLUSIVE	Specifies whether the queue is opened to get messages with exclusive access. **True** indicates that the queue is open for use with subsequent MQGET calls. Calls will fail with reason code MQRC_OBJECT_IN_USE if the queue is currently used (open) by this or another application for input of any type.	True or False. The configured default is False.
MQOO_INPUT_SHARED	Specifies whether the queue is opened to get messages with shared access. **True** indicates that the queue is open for use with subsequent MQGET calls. Calls will succeed, even when the queue is currently used (open) by this or another application for input of any type.	True or False. The configured default is False.
MQOO_INQUIRE	Specifies whether the object is opened to inquire attributes. **True** indicates that the queue, name list, process definition, or queue manager is open for use with subsequent MQINQ calls.	True or False. The configured default is True.
MQOO_OUTPUT	Specifies whether the object is opened to put messages. **True** indicates that MQOPEN call can succeed, even if the InhibitPut queue attribute is set to MQQA_PUT_INHIBITED (though subsequent MQPUT calls will fail). **Note** – The value of this parameter is ignored when the Adapter is operating in automatic connection mode, because the Adapter must be capable of both receiving and sending messages.	True or False. The configured default is True.

TABLE 25 Connectivity Map - Outbound Adapter Settings > Queue Open Options   *(Continued)*

Name	Description	Required Value
MQOO_PASS_ALL_ CONTEXT	Specifies whether to allow all context to pass. **True** indicates that the MQPMO_PASS_ALL_CONTEXT option is specified in the PutMsgOpts parameter when a message is put on a queue, and gives the message identity and origin context information from an input queue opened with the MQOO_SAVE_ALL_CONTEXT option. **True** also indicates that MQOO_PASS_IDENTITY_CONTEXT is implied and does not need to be specified.	True or False. The configured default is False.
MQOO_PASS_IDENTITY_ CONTEXT	Specifies whether to allow identity context to pass. **True** indicates that the MQPMO_PASS_IDENTITY_CONTEXT option to be specified in the PutMsgOpts parameter when a message is put on a queue. This gives the message the identity context information from an input queue opened with the MQOO_SAVE_ALL_CONTEXT option. True indicates that the MQOO_OUTPUT option must be specified.	True or False. The configured default is False.
MQOO_RESOLVE_NAMES	Specifies MQOO_RESOLVE_NAMES. Select **True** if you want to use the resolved queue manager name and resolved queue name attributes of the ImqQueue class.	True or False. The configured default is False.
MQOO_SAVE_ALL_ CONTEXT	Specifies whether to save context when message is retrieved. True indicates that context is saved. Context information is associated with this queue handle and set from the context of any message retrieved using this handle.	true or false. The configured default is False.
MQOO_SET	Specifies whether the queue is open to set attributes. True indicates that the queue is open to set attributes and for use with subsequent MQSET calls.	True or False. The configured default is False.

TABLE 25 Connectivity Map - Outbound Adapter Settings > Queue Open Options   *(Continued)*

Name	Description	Required Value
MQOO_SET_ALL_ CONTEXT	Specifies whether to allow all context to be set. **True** indicates that all context can be set. The MQPMO_SET_ALL_CONTEXT option is specified in the PutMsgOpts parameter when a message is put on a queue. Gives the identity and origin context information contained in the MsgDesc parameter specified on the MQPUT or MQPUT1 call to the message.	True or False. The configured default is False.
MQOO_SET_IDENTITY_ CONTEXT	Specifies whether to allow identity context to be set. **True** indicates that identity context can be set. The MQPMO_SET_IDENTITY_CONTEXT option can be specified in the PutMsgOpts parameter when a message is put on a queue. Gives the identity and origin context information contained in the MsgDesc parameter specified on the MQPUT or MQPUT1 call to the message.	True or False. The configured default is False.

Outbound WebSphere MQ Adapter Connectivity Map Properties

The outbound WebSphere MQ Adapter parameters, accessed from the Connectivity Map, are contained in the following section.

- "Outbound WebSphere MQ Adapter Settings" on page 43
- "Outbound WebSphere MQ Adapter Settings > Queue Open Options" on page 44

Outbound WebSphere MQ Adapter Settings

The Outbound Adapter Settings section of the WebSphere MQ Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 26 Connectivity Map - Outbound Adapter Settings > GetMessageOptions > options

Name	Description	Required Value
Queue Name	Specifies the name of queue to which the message is published. This parameter is optional. The queue name may also be specified manually in the Business Process or Collaboration that effects the put.	The queue name to which the message is retrieved or published.

TABLE 26 Connectivity Map - Outbound Adapter Settings > GetMessageOptions > options *(Continued)*

Name	Description	Required Value
Security Exit	Specifies the optional, package qualified name of a user-defined class that implements the com.ibm.mq.MQSecurityExit interface. WebSphere MQ invokes an instance of the class whenever the Adapter attempts to connect to the queue manager. The named class must include a default constructor. This parameter is only used for client connections. Security Exits are not applicable to bindings connections. For more information about Security Exits, see the IBM document, "WebSphere MQ Using Java" (CSQZAW09) regarding MQSecurityExit.	The name of the user-defined class. This property also requires an assigned value for the `Security Exit JAR Classpath` property.
Security Exit JAR Classpath	Specifies the absolute path to the JAR file that contains the named Security Exit. This property is required if the Security Exit is specified. The specified JAR is packaged into the application (EAR) file that is generated during Project activation. If the specified JAR cannot be accessed or found, the activation will fail. If this property value is left blank, you must ensure that a JAR file containing the Security Exit is made accessible to the runtime Environment prior deploying the Project (for example, by manually copying the JAR file into the Integration Server's lib directory prior to or during the Deployment Process). For more information about Security Exits, see the IBM document, "WebSphere MQ Using Java" (CSQZAW09) regarding MQSecurityExit.	The absolute path for the JAR file that contains the named Security Exit class.

Outbound WebSphere MQ Adapter Settings > Queue Open Options

The Outbound Adapter Settings > Queue Open Options section of the WebSphere MQ Adapter Connectivity Map properties contains the top-level parameters displayed in the following table. These properties apply specifically to the MQOPEN calls.

TABLE 27 Connectivity Map - Outbound Adapter Settings > Queue Open Options

Name	Description	Required Value
MQOO_ALTERNATE_ USER_AUTHORITY	Specifies whether an alternate user identifier is used to check the authorization for the open. **True** indicates that and alternate user identifier is used.	True or False. The configured default is False.
MQOO_BIND_AS_Q_DEF	Specifies whether the binding used for the queue handle is taken from the DefBind queue attribute. True indicates that the binding used is taken from DefBind queue attribute.	True or False. The configured default is False.
MQOO_BIND_NOT_FIXED	Specifies whether the local queue manager binds the queue handle to a particular instance of the destination queue, when the object being opened is a cluster queue. **True** indicates that the local queue manager will bind to a specific destination. This option is ignored when specified for a queue that is not a cluster queue.	True or False. The configured default is False.
MQOO_BIND_ON_OPEN	Specifies whether the local queue manager binds the queue handle to a particular instance of the destination queue, when the object being opened is a cluster queue. **True** indicates that the local queue manager *will not* bind to a specific destination. This option is ignored when specified for a queue that is not a cluster queue.	True or False. The configured default is False.
MQOO_BROWSE	Specifies whether the queue is opened to browse messages. **True** indicates that the queue is open for use with MQGET calls with the following options: **MQGMO_BROWSE_FIRST** **MQGMO_BROWSE_NEXT** **MQGMO_BROWSE_MSG_UNDER_ CURSOR**	True or False. The configured default is False.
MQOO_FAIL_IF_ QUIESCING	Specifies whether the MQOPEN call fails when the queue manager is in quiescing state. Used to control authorization checking. **True** indicates that the MQOPEN call will fail if queue manager is quiescing.	True or False. The configured default is False.

TABLE 27 Connectivity Map - Outbound Adapter Settings > Queue Open Options   *(Continued)*

Name	Description	Required Value
MQOO_INPUT_AS_Q_DEF	Specifies whether the queue is opened to browse messages using the queue-defined default. **True** indicates that the queue is open for use with subsequent MQGET calls. **Note** – The value of this parameter is ignored when the Adapter is operating in automatic connection mode, because the Adapter must be capable of both receiving and sending messages.	True or False. The configured default is True.
MQOO_INPUT_ EXCLUSIVE	Specifies whether the queue is opened to get messages with exclusive access. **True** indicates that the queue is open for use with subsequent MQGET calls. Calls will fail with reason code MQRC_OBJECT_IN_USE if the queue is currently used (open) by this or another application for input of any type.	True or False. The configured default is False.
MQOO_INPUT_SHARED	Specifies whether the queue is opened to get messages with shared access. **True** indicates that the queue is open for use with subsequent MQGET calls. Calls will succeed, even when the queue is currently used (open) by this or another application for input of any type.	True or False. The configured default is False.
MQOO_INQUIRE	Specifies whether the object is opened to inquire attributes. **True** indicates that the queue, name list, process definition, or queue manager is open for use with subsequent MQINQ calls.	True or False. The configured default is True.
MQOO_OUTPUT	Specifies whether the object is opened to put messages. **True** indicates that MQOPEN call can succeed, even if the InhibitPut queue attribute is set to MQQA_PUT_INHIBITED (though subsequent MQPUT calls will fail). **Note** – The value of this parameter is ignored when the Adapter is operating in automatic connection mode, because the Adapter must be capable of both receiving and sending messages.	True or False. The configured default is True.

TABLE 27 Connectivity Map - Outbound Adapter Settings > Queue Open Options   *(Continued)*

Name	Description	Required Value
MQOO_PASS_ALL_ CONTEXT	Specifies whether to allow all context to pass. **True** indicates that the MQPMO_PASS_ALL_CONTEXT option is specified in the PutMsgOpts parameter when a message is put on a queue, and gives the message identity and origin context information from an input queue opened with the MQOO_SAVE_ALL_CONTEXT option. **True** also indicates that MQOO_PASS_IDENTITY_CONTEXT is implied and does not need to be specified.	True or False. The configured default is False.
MQOO_PASS_IDENTITY_ CONTEXT	Specifies whether to allow identity context to pass. **True** indicates that the MQPMO_PASS_IDENTITY_CONTEXT option to be specified in the PutMsgOpts parameter when a message is put on a queue. This gives the message the identity context information from an input queue opened with the MQOO_SAVE_ALL_CONTEXT option. True indicates that the MQOO_OUTPUT option must be specified.	True or False. The configured default is False.
MQOO_RESOLVE_NAMES	Specifies MQOO_RESOLVE_NAMES. Select **True** if you want to use the resolved queue manager name and resolved queue name attributes of the ImqQueue class.	True or False. The configured default is False.
MQOO_SAVE_ALL_ CONTEXT	Specifies whether to save context when message is retrieved. True indicates that context is saved. Context information is associated with this queue handle and set from the context of any message retrieved using this handle.	True or False. The configured default is False.
MQOO_SET	Specifies whether the queue is open to set attributes. True indicates that the queue is open to set attributes and for use with subsequent MQSET calls.	True or False. The configured default is False.

TABLE 27   Connectivity Map - Outbound Adapter Settings > Queue Open Options   *(Continued)*

Name	Description	Required Value
MQOO_SET_ALL_ CONTEXT	Specifies whether to allow all context to be set. **True** indicates that all context can be set. The MQPMO_SET_ALL_CONTEXT option is specified in the PutMsgOpts parameter when a message is put on a queue. Gives the identity and origin context information contained in the MsgDesc parameter specified on the MQPUT or MQPUT1 call to the message.	True or False. The configured default is False.
MQOO_SET_IDENTITY_ CONTEXT	Specifies whether to allow identity context to be set. **True** indicates that identity context can be set. The MQPMO_SET_IDENTITY_CONTEXT option can be specified in the PutMsgOpts parameter when a message is put on a queue. Gives the identity and origin context information contained in the MsgDesc parameter specified on the MQPUT or MQPUT1 call to the message.	True or False. The configured default is False.

Running XA Transactions in Client (Outbound) Mode

The outbound WebSphere MQ Adapter supports XA transactions with WebSphere MQ Manager servers running on Solaris, AIX, HP-UX, Linux or Windows (not supported for OS/400 or z/OS). If your Oracle Java Composite Application Platform Suite is installed on a different computer than your WebSphere MQ server, XA mode requires that you first install the WebSphere MQ base client, and then the WebSphere MQ Extended Transactional client, on the Application Server.

The transaction manager of the WebSphere MQ Adapter in XA mode, runs in the Application Server. The Application Server requires two JAR files, `com.ibm.mq.jar` and `com.ibm.mqetclient.jar`, to be added to the Application Server classpath.

The IBM document WebSphere MQ External Transactional Clients, provides information on distributed XA transactions and limitations, such as WebSphere MQ API that cannot be issued in XA mode.

According to IBM, when using the WebSphere MQ Extended Transactional client, a client application can be connected to only one queue manager at a time within a single thread. This restriction applies to the WebSphere MQ Extended Transactional client. (The WebSphere MQ base client can be connected to more than one queue manager concurrently within a single thread.) For the WebSphere MQ Adapter this means, in one deployment, you are only allowed to have one outbound XA mode Adapter connecting to a WebSphere MQ external system. You

cannot have multiple XA outbound connections to different WebSphere MQ external systems and expect the Integration Server Transactional manager to handle XA transaction for multiple WebSphere MQ queue managers.