



BEA AquaLogic® Service Bus

MQ Transport User Guide

Version: 3.0
Revised: February 2008

Contents

Introduction to the MQ Transport

Messaging Patterns	1-3
Environment Values	1-3
Quality of Service	1-4
MQ Clusters and the MQ Transport	1-5

Using the MQ Transport

Adding MQ Client Libraries to Your Environment	2-1
MQ Connection Resources	2-2
Creating an MQ Connection Resource	2-2
Configuring Proxy Services to Use the MQ Transport	2-4
Transport Headers	2-5
Configuring Transport Headers	2-15
Configuring Business Services to Use the MQ Transport	2-16
Error Handling	2-17
Limitations of the MQ Transport	2-18

Using the WebSphere JMS MQ Interface

Using the WebSphere MQ JMS Interface	3-1
Messaging Types	3-2
Non-Persistent Messaging	3-2
Non-XA Persistent Messaging	3-3
XA Messaging	3-3

Tuning WebSphere MQ 3-3

Introduction to the MQ Transport

ALSB supports access to IBM WebSphere® MQ using the MQ Transport. This transport supports both inbound and outbound connectivity. MQ proxy services can receive messages from WebSphere MQ and MQ business services can route messages to WebSphere MQ.

Key Features

The following key features are supported for the MQ transport:

- Inbound and outbound connectivity. MQ proxy services can receive messages from WebSphere MQ and MQ business services can route messages to WebSphere MQ.
- Access to WebSphere MQ version 5.3 and 6.0. For more information, see [Adding MQ Client Libraries to Your Environment](#).
- Sending and receiving messages of Any XML, Binary, XML, Text and MFL types.
- Processing of all the MQMD (MQ message descriptor) headers. A message descriptor is an attribute representing a property of the message that is either being sent or received.

For a list of MQ headers that can be configured for outbound response and inbound requests, see [Transport Headers](#).

- TCP/IP and bindings mode for connecting to queue managers.

Use the bindings mode to connect to WebSphere MQ that is located on the same physical machine as ALSB. Use TCP/IP to connect to WebSphere MQ that does not reside on the same machine as ALSB.

- One-way and two-way SSL on inbound and outbound transport.

SSL is supported only when TCP/IP is used to connect to WebSphere MQ from ALSB.

Advantages of Using the MQ Transport

Because the WebSphere MQ for Java APIs do not support XA transactions, ALSB does not support XA transactions for the MQ transport. If XA is required, JMS transport with WebSphere MQ JMS interface should be used. For more information, see [Chapter 3, “Using the WebSphere JMS MQ Interface”](#) and [AquaLogic Service Bus Interoperability Solutions for JMS and WebSphere MQ](#).

Using the MQ transport has the following advantages over using the WebSphere MQ JMS interface:

- Faster connection to WebSphere MQ through the MQ transport than through the WebSphere MQ JMS interface.
- Ability to read and generate MQ messages. Using the JMS interface, it is not possible to set certain headers.
- Support for sending and receiving MQ receipt messages.
- No explicit binding of MQ Connection Factory and MQ Queue to WebLogic’s JNDI is required.
- Configuration of resources like a JMS provider, outside of ALSB, is not required.
- Performance improvement because messages are sent directly using the transport instead of channeling them through the JMS transport.

Supported Service Types

The MQ transport is available for the Message Service and XML Service service types.

For more information, see [Business Services: Creating and Managing](#) and [Proxy Services: Creating and Managing](#) in *Using the AquaLogic Service Bus Console*.

Messaging Patterns

The Native MQ transport supports one-way and request-response messaging patterns for both inbound and outbound connectivity. By default, one-way messaging is supported. A proxy or business service supports request-response messaging when you set the `is-response-required` option while configuring the service.

The inbound and outbound transports support the asynchronous request-response pattern using `messageID` or `correlationID` for correlation between the request and the response. You can set the response correlation pattern during service configuration. For more information, see “CorrelationID” and “MessageID” in [Transport Headers](#).

The outbound transport provides an option to auto-generate the correlation ID / messageID or use the one specified by you in the message flow. Select the `Auto-generate Correlation Value` option, to indicate that the correlation ID / message ID should be auto-generated by the transport. If the option is not selected, the value specified by you in the message flow is used.

If the correlation value (`messageID` / `correlationID`) is not auto-generated and if the managed server goes down, the remaining response messages may never get removed when the server is restarted. BEA recommends that the `Expiry` header on the request is configured to a finite value and that the `Report` header contains the `MQC.MQRO_PASS_DISCARD_AND_EXPIRY` option. The `MQC.MQRO_PASS_DISCARD_AND_EXPIRY` option serves as a directive to the receiving client that the message descriptor of the reply should inherit the `Expiry` header value of the request message. This ensures that the response messages are removed by the MQ server after the configured expiry period has elapsed. When the correlation value is automatically generated, the ALSB server is responsible for cleaning up any remaining response messages.

For more information about configuring **Is Response Required**, **Response Correlation Pattern**, **Auto Generate Correlation Value** for a service, see [Configuring Proxy Services to Use the Native MQ Transport](#) and [Configuring Business Services to Use the Native MQ Transport](#).

Environment Values

Environment values are certain predefined fields in the configuration data whose values are very likely to change when you move your configuration from one domain to another (for example, from test to production). For information about updating environment values, see [Finding and Replacing Environment Values](#) in *Using the AquaLogic Service Bus Console*. The following sections provide a list of environment values for the Native MQ transport services and the MQ connection resource.

Services based on the MQ transport support the `work Manager` (Inbound and Outbound) environment value.

MQ connections are sharable resources that can be reused across multiple MQ proxy and business services. MQ Connection resources provide the connection parameters required for connecting to a MQ queue manager. The MQ connection resource supports the following environment and operational values:

- MQ Connection Host Name
- MQ Connection Port
- MQ Queue Manager Name
- MQ Channel Name
- MQ Connection Pool Size
- MQ Connection Timeout

For information about updating environment values, see [Customization](#) in *Using the AquaLogic Service Bus Console*.

Quality of Service

When the inbound transport is MQ and the Quality of Service (QoS) on the outbound transport is *exactly-once*, the resultant QoS will be *at-least-once*. By default, the QoS on the outbound transport is *exactly-once*.

Note: You must create error handling logic (including any retry logic) in the pipeline error handler. For information about configuring error handling, see [Proxy Services: Error Handlers](#) in *Using the AquaLogic Service Bus Console*.

When the outbound is request-response, the QoS is *at-least-once* only if the outbound transport is configured to support *exactly-once* QoS.

For more information about QoS in ALSB messaging, see “Quality of Service” in [Modeling Message Flow in ALSB](#) in the *AquaLogic Service Bus User Guide*.

MQ Clusters and the MQ Transport

The cluster support in WebSphere MQ is that of store-and-forward messaging and not of load-balancing and fail over. The cluster queues in WebSphere MQ are created locally on one of the queue managers and shared with other cluster members that act as remote forwarders to the shared queue.

Requests from the MQ transport are load balanced by sending messages using the load balancing algorithm to the members of the cluster. However, the transport receives messages by accessing only the MQ server node that holds the reference to the local queue.

Introduction to the MQ Transport

Using the MQ Transport

You can use the MQ transport to enable MQ proxy services to get messages from WebSphere MQ and MQ business services to send messages to WebSphere MQ. For inbound and outbound connectivity to WebSphere MQ 5.3 and 6.0 to be available, ensure that the MQ client libraries are available in your ALSB environment. For more information, see [Adding MQ Client Libraries to Your Environment](#).

MQ proxy and business services must first connect to a MQ queue manager before accessing the queue. MQ connection resources provide the connection parameters required for connecting to a MQ queue manager. You can configure your proxy and business services to use these connection resources. For more information, see [MQ Connection Resources, Configuring Proxy Services to Use the MQ Transport](#), and [Configuring Business Services to Use the MQ Transport](#). For information about queue managers, see <http://www.redbooks.ibm.com/redbooks/SG247128/wwhelp/wwhimpl/java/html/wwhelp.htm>

When you configure a proxy service, you can use a Transport Header action to set the header values in messages. For more information, see [Transport Headers](#).

Adding MQ Client Libraries to Your Environment

ALSB is a client for WebSphere MQ and although ALSB supports run-time server compatibility for both MQ 5.3 and 6.0, these MQ libraries are not bundled with the ALSB installer. You need to ensure that the MQ 6.0 client library, `com.ibm.mq.jar`, is available in your environment.

For information about the system requirements for WebSphere MQ, see <http://www-306.ibm.com/software/integration/wmq/requirements/index.html>

To add the MQ client libraries to your environment:

1. Download the MQ client libraries from the IBM web site:
<http://www-306.ibm.com/software/integration/wmq/library/index.html>
2. Stop the domain server.
3. Copy the `com.ibm.mq.jar` file to the
`BEA_HOME\user_projects\domains\\lib` directory.

BEA_HOME is the ALSB installation directory. Replace `<domain_name>` with the actual name of your domain.
4. Restart the domain server.

If you use bindings mode to connect MQ Queue Manager located on the same machine as ALSB, add `<MQ_install_directory>/bin` and `<MQ_install_directory>/java/lib` to the PATH environment variable.

MQ Connection Resources

MQ connections are sharable resources that can be reused across multiple MQ proxy and business services. MQ proxy and business services must connect to a MQ queue manager before accessing the MQ queue. MQ connection resources provide the connection parameters required for connecting to a MQ queue manager.

Each MQ connection resource has a connection pool. Multiple business services and proxy services using the same queue manager share a connection pool. Any business or proxy service using a given MQ connection resource to connect to a given queue manager uses the same connection pool that was created for that MQ connection resource.

For information about managing MQ connection resources, see [MQ Connections](#) in *Using the AquaLogic Service Bus Console*.

Creating an MQ Connection Resource

To create an MQ connection resource:

1. In the Project Explorer, select MQ Connection as the resource type. The Create a New MQ Connection Resource page appears.
2. Enter the name and description of the resource.
3. Select the connection type as `tcp mode` or `bindings mode`.

Use the `tcp` mode when the MQ Queue Manager is not available on the same machine as ALSB. Use `bindings` to connect MQ Queue Manager located on the same machine as ALSB.

4. If the connection type is `bindings` mode, enter the name of the MQ Queue Manager.
5. If the connection type is `tcp` mode,

- a. Enter the host name of the MQ Queue Manager.
- a. Enter the port number of the MQ Queue Manager Listener.
- b. Enter the name of the MQ Queue Manager.
- c. Enter the coded character set identifier (CCSID) to be used for connecting to the MQ Queue Manager for client connection mode.
- d. Enter the Queue Manager server connection channel name.
- e. Select the **SSL** option, if required, to use HTTPS protocol for sending the messages.

After you select this option, select the 2-way SSL Required option to enable 2-way SSL for both client-side and server-side authentication. Clear the 2-way SSL Required option for 1-way SSL for only server-side authentication.

Note: When you select 2-way SSL, you need to provide a reference to a service key provider. A service key provider contains Public Key Infrastructure (PKI) credentials that proxy services use for decrypting inbound SOAP messages and for outbound authentication and digital signatures. For more information about using service providers, see [Service Key Providers](#) in *Using the AquaLogic Service Bus Console*.

- f. Specify the static service account.

A service account provides a user name and password that proxy services and business services use for outbound authentication or authentication to a local or remote resource. For more information about using service accounts, see [Service Accounts](#) in *Using the AquaLogic Service Bus Console*.

6. Select the version of WebSphere MQ as 5.3 or 6.0.
7. Enter the size of the MQ connection pool. Default is 10.
8. Enter the duration after which a connection must timeout. Default is 30 seconds.
9. Click **Save**.

The MQ connection resource is created. You can use it across proxy and business services that use the MQ transport.

Configuring Proxy Services to Use the MQ Transport

When you create a proxy service in ALSB Console, select the transport protocol as `mq` and specify the endpoint configuration for a proxy service that uses the MQ transport in the Transport Configuration page. Specify the URI in `mq://local-queue-name?conn=mq-connection-resource-ref` format where, `local-queue-name` is the name of the local queue configured on the MQ server, `mq-connection-resource-ref` points to the location of the MQ connection resource. For example, if you create a MQ connection resource, `mqConnection` in the `defaultMQ` folder and the queue name is `testQueue`, the URI would be `mq://testQueue?conn=defaultMQ/mqConnection`

Note: During service configuration, select either Message service or XML service as the service type in the General Configuration page. For more information, see [Supported Service Types](#) and [Proxy Services: Creating and Managing](#) in *Using the AquaLogic Service Bus Console*.

Configure the MQ transport for a proxy service with values as described in the following table:

Table 1-1 MQ Proxy Service Configuration

Field	Description
Polling Interval	Time in milliseconds after which the MQ Queue is polled for new messages. Default is 1000 milliseconds.
Is Response Required	Select this option if a response needs to be published to the specified destination.
Response Correlation Pattern	Specify if the response correlation pattern should be based on message ID or correlation ID.

Table 1-1 MQ Proxy Service Configuration

Field	Description
MQ Response URL	<p>The destination to which the response should be published. Specify the URI in <code>mq://local-queue-name?conn=mq-connection-resource-ref</code> format.</p> <p>Note: If the <code>replyToQueueName</code> and <code>replyToQueueManagerName</code> MQMD headers are not set on the incoming request, this response URI is used as the default response destination. However, if a response URI is set in the headers, ALSB connects to the queue manager specified in <code>replyToQueueManagerName</code> and sends the response to the response queue specified in <code>replyToQueueName</code>.</p>
Response Message Type	The message type of the response can be bytes or text.
Client Response Timeout	<p>The duration to wait for the response to be published before closing the connection.</p> <p>Note: You can specify the Response Correlation Pattern, MQ Response URL, Response Message Type, and Client Response Timeout values only if a response is required.</p>
Dispatch Policy	<p>Select a dispatch policy for this endpoint.</p> <p>Dispatch policy refers to the instance of WLS Work Manager that you want to use for the service endpoint. For information about work managers, see Using Work Managers to Optimize Scheduled Work and Create Work Manager in the WebLogic Server Administration Console Online Help.</p>

For more information about configuring proxy services using MQ transport, see MQ Transport Configuration Page in [Proxy Services: Creating and Managing](#) in *Using the AquaLogic Service Bus Console*.

Transport Headers

The various headers used by the MQ transport are listed in [Table 1-2](#). Most of the headers are common to both outbound requests and inbound response. The Reply To Queue Name, Reply To Queue Manager Name, User ID and Version headers can be edited only for the inbound response.

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Accounting Token	<p>Accounting token is part of the identity context of the message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Application ID Data	<p>Application ID data is part of the identity context of the message. This value can be used to provide additional information about the message or its originator.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Application Origin Data	<p>Data about the originating application. This value can be used by the application to provide additional information about the origin of the message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Backout Count	<p>The number of times the message was returned by the MQ Queue, as part of a unit of work, and subsequently backed out.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Character Set	<p>The coded character set identifier of character data in the application message data.</p> <p>Inbound Transport Action: This field is used by the inbound transport to convert data in a specific representation. For request-response messaging, the characterSet header from the request message is copied to the response. When this header is not configured on the incoming request, default value of the MQC.MQCCSI_Q_MGR field is assumed.</p> <p>Outbound Transport Action: This header can be set in the pipeline for the outbound transport. If this header value is not set, the default MQC.MQCCSI_Q_MGR value is assumed.</p>	Both
Correlation ID	<p>The correlation-id of the message that should be retrieved.</p> <p>Inbound Transport Action: For correlationID-based response correlation pattern, the correlationID from the request is echoed on the response. The user can override the correlationID in the response pipeline.</p> <p>Outbound Transport Action: When the Auto-generate correlationID option is selected during service configuration, the outbound transport will automatically generate a correlationID and overwrite the correlationID from the transport header. If this value is not specified, the correlationID specified in the pipeline is used.</p> <p>For one-way messaging, the correlationID specified in the pipeline is used in the (outbound) request.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Encoding	<p>The representation used for numeric values in the application message data.</p> <p>Inbound Transport Action: The inbound transport uses this header to interpret the incoming message data. If this header is not configured in the response pipeline, the default value of <code>MQC.MQENC_NATIVE</code> is used.</p> <p>Outbound Transport Action: If this header is not set in the pipeline for the outbound transport, the default value of <code>MQC.MQENC_NATIVE</code> is used.</p>	Both
Expiry	<p>The expiry time (in tenths of a second) is set by the application that puts the message. After a message's expiry time has elapsed, it is eligible to be discarded by the queue manager.</p> <p>Inbound Transport Action: For request-response messaging, the inbound transport copies the expiry header of the request to the response.</p> <p>Outbound Transport Action: If the corresponding transport header is set in the pipeline, it is copied to the outbound request message.</p> <p>Note: The report header will always contain the <code>MQC.MQRO_PASS_DISCARD_AND_EXPIRY</code> option (in addition to others). This option is a directive to the receiving client that the expiry time of the original message should be copied to the report or reply message.</p>	Both
Feedback	<p>The nature of the feedback report. This value is used with a message of type <code>MQC.MQMT_REPORT</code> to indicate the nature of the report.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Format	<p>Format name of the message data. The format name is used by the sender of the message to indicate the nature of the data in the message to the receiver.</p> <p>Inbound Transport Action: When the field is set to <code>MQC.MQFMT_MD_EXTENSION</code>, the inbound transport will read the extended MQMD object.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Group ID	<p>The value that identifies the message group to which the physical message belongs.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Offset	<p>In a segmented message, offset of data in the physical message from the start of the logical message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Original Length	<p>Original length of a segmented message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Message Flags	<p>Flags that control the segmentation and status of a message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Message ID	<p>ID of the message to be retrieved.</p> <p>Inbound Transport Action: If messageID is not specified in the response pipeline, the messageID header is set to MQC.MQMI_NONE.</p> <p>For messageID-based correlation, the inbound transport copies the messageID from the request to the correlationID header of the response. MessageID-based correlation is assumed when the report header contains the MQC.MQRO_COPY_MSG_ID_TO_CORREL_ID option.</p> <p>Outbound Transport Action: When the Auto-generate messageID option is specified during service configuration, the outbound transport automatically generates the messageID and overwrites the messageID from the transport header. If this value is not specified, the messageID transport header is used.</p> <p>For one-way messaging, the messageID specified in the pipeline is used in the outbound request. If this value is not specified, the messageID is automatically generated by the transport.</p>	Both
Message Sequence Number	<p>Sequence number of a logical message within group.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Message Type	<p>Message type of the message.</p> <p>Inbound Transport Action: The inbound transport reads and processes messages of any type including MQC.MQMT_REQUEST, MQC.MQMT_DATAGRAM, MQC.MQMT_REPLY and MQC.MQMT_REPORT. The inbound transport does not generate report messages.</p> <p>Outbound Transport Action: The outbound transport generates messages of any type including MQC.MQMT_DATAGRAM, MQC.MQMT_REQUEST, MQC.MQMT_REPLY and MQC.MQMT_REPORT. When the messageType header is not configured in the pipeline, the transport generates messages of type MQC.MQMT_DATAGRAM when the messaging pattern is one-way and MQC.MQMT_REQUEST when the messaging pattern is request-reply.</p>	Both
Persistence	<p>The message persistence.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Priority	<p>Priority of the message</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Put Application Name	<p>The name of the application that put the message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Put Application Type	<p>The type of the application that put the message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both
Put Date Time	<p>The time and date when the message was put.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Reply To Queue Name	<p>The name of the queue to which a reply should be sent.</p> <p>The application that issued the get request for the message can send MQC.MQFMT_REPLY and MQC.MQFMT_REPORT messages to this queue.</p> <p>Inbound Transport Action: The inbound transport uses the <code>replyToQueueName</code> as the response queue name when this field is set. If this values is not set, the queue name is derived from the default destination URI.</p> <p>Outbound Transport Action: In request/response message pattern, <code>replyToQueueName</code> set in the message flow is ignored. In one way message pattern, <code>replyToQueueName</code> set in the message flow is used in the outbound messages.</p>	Inbound Response
Reply To Queue Manager Name	<p>The name of the queue manager to which reply or report messages can be sent.</p> <p>Inbound Transport Action: In request/response message pattern, if the inbound message <code>replyToQueueManager</code> header value not match the configured value for the queue manager in the response URI, the response message is dropped and a transport error is logged.</p> <p>Outbound Transport Action: In request/response message pattern, <code>replyToQueueManager</code> set in the message flow is ignored. In one way message pattern, <code>replyToQueueManager</code> set in the message flow is used in the outbound messages.</p>	Inbound Response

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
Report	<p>A report is a message about another message. This field enables the application sending the original message to specify which report messages are required, whether the application message data is to be included in them, and also how the message and correlation ID in the report or reply are to be set. It comprises one or more constants from the MQC class combined by means of the '+' or ' ' operators.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline. For request-response messaging, this header can be configured in the response pipeline.</p> <p>Outbound Transport Action: The transport always sets a combination of the following options in the report field.</p> <p>Set MQC.MQRO_COPY_MSG_ID_TO_CORREL_ID if messageID-based correlation pattern is used and MQC.MQRO_PASS_CORREL_ID if correlationID-based correlation pattern is used. Always set MQC.MQRO_PASS_DISCARD_AND_EXPIRY.</p> <p>Note: These options are set in addition to the options specified on the corresponding transport header in the pipeline.</p>	Both

Table 1-2 Transport Headers

Header	Description	Inbound Response / Outbound Request
User ID	<p>It is part of the identity of the message and identifies the user who originated the message.</p> <p>Inbound Transport Action: No explicit processing is done by the transport. The header is copied to the transport header in the pipeline.</p> <p>Outbound Transport Action: No explicit processing is done by the transport. If the corresponding transport header is set in the pipeline, it is copied to the request message.</p>	Inbound Response
Version	<p>The version number of the message descriptor.</p> <p>Inbound Transport Action: The inbound transport supports both version 1 and version 2 message descriptors.</p> <p>Outbound Transport Action: By default, the outbound transport generates version 2 headers. However, this field can be overridden in the pipeline.</p>	Inbound Response

Configuring Transport Headers

You can configure the transport headers for both inbound and outbound requests in the Message Flow. For information about the transport headers related to the MQ transport, see [Transport Headers](#).

In the message flow, use a Transport Header action to set the header values in messages. For more information, see [Adding Transport Header Actions](#) in *Using the AquaLogic Service Bus Console*.

When the transport header is explicitly set in the message flow, this value overrides the header value except in the following scenarios:

- For the outbound request-response pattern, when the `Auto-generate Correlation value` option is selected for a outbound request with a request-response message pattern, the correlation ID is always generated even if this value is set in the message flow.
- When the report header is set in the message flow, the combination of multiple directives associated with the report header are merged with the default directives.

- When the `replyToQueueManagerName` or `replyToQueueName` headers are set in the message flow for an outbound request with a request/response message pattern, these values are ignored. Instead, these transport header values are derived from the response URI configured for the business service.
- For the inbound response, when the value in the `replyToQueueManagerName` header does not match the queue manager name specified in the response URI, an error message is generated and the response message is not sent.
- The `replyToQueueName` set in the inbound request header overrides the value configured in the reply to URI for the proxy service.
- For a one-way business service, when the message type header is configured to be of type request in the message flow, the `replyToQueueName` header must be specified. If this value is not specified, an error is generated on the MQ server and the message is rolled back.

Configuring Business Services to Use the MQ Transport

When you create a business service from ALSB Console, select the transport protocol as `mq` and specify the endpoint URI in the Transport Configuration page. Specify the URI in `mq://local-queue-name?conn=mq-connection-resource-ref` format where, `local-queue-name` is the name of the local queue configured on the MQ server, `mq-connection-resource-ref` points to the location of the MQ connection resource. For example, if you create a MQ connection resource, `mqConnection` in the `defaultMQ` folder and the queue name is `testQueue`, the URI would be `mq://testQueue?conn=defaultMQ/mqConnection`

Note: Since the only service types that are supported for the MQ transport are Message and XML ensure that you select either Message service or XML service, when you select the service type in the General Configuration page. For more information, see in [Business Services: Creating and Managing](#) in *Using the AquaLogic Service Bus Console*.

To configure the MQ transport for a business service, specify the values as described in the following table:

Field	Description
Is Response Required	Select this option if a response needs to be published to the specified destination.
Polling Interval	Time in seconds after which the MQ Queue is polled for new messages. The default value is 1000 seconds.
Message Type	The message type of the response can be <code>bytes</code> or <code>text</code> .
Response Correlation Pattern	Specify if the response correlation pattern should be based on message ID or correlation ID.
Auto-generate Correlation Value	Select this option to automatically generate the correlation value.
MQ Response URL	The destination to which the response should be published. Specify the URI in <code>mq://local-queue-name?conn=mq-connection-resource-ref</code> format.
Response Timeout	The duration to wait for the response to be published.
Dispatch Policy	Select a dispatch policy for this endpoint. Leave blank to use the default dispatch policy. Dispatch policy refers to the instance of WLS Work Manager that you want to use for the service endpoint. For information about work managers, see Using Work Managers to Optimize Scheduled Work and Create Work Manager in the WebLogic Server Administration Console Online Help.

For more information about configuring business services using MQ transport, see MQ Transport Configuration Page in [Business Services: Creating and Managing](#) in *Using the AquaLogic Service Bus Console*.

Error Handling

You can configure MQ-transport business services to handle application and communications errors as follows:

- Application errors – You can specify whether or not to retry business service endpoint URIs when application errors occur. For more information, see [Retry Application Errors](#) in

“Creating and Configuring Business Services - Transport Configuration page” in *Using the AquaLogic Service Bus Console*.

- Communication errors – You can configure business service URIs to be taken offline when communication errors occur. For more information, see [Enable business service endpoint URIs to be taken offline](#) in Table 23-8 in [Configuring Operational Settings for Business Services](#) and [Viewing Business Services Endpoint URIs Metrics](#) in *Using the AquaLogic Service Bus Console*.

For more information on error messages, see [Index of Error Messages by Range](#).

Limitations of the MQ Transport

The following are the limitations of the MQ transport:

- MQ transport does not support
 - XA for the MQ transport. For more information, see [Using the WebSphere JMS MQ Interface](#).
 - Defining MQ exits.
- You cannot call a request-response proxy service based on MQ proxy service:
 - From a proxy service that has been configured with a route action or dynamic routing and routing table actions).
 - Using the service callout action.
- You cannot call a proxy service with a service callout where the target is a request-response proxy service based on MQ transport.
- You cannot use an indirect call to a request-response MQ proxy service in the ALSB test console.

Using the WebSphere JMS MQ Interface

The WebSphere MQ for Java APIs do not support XA transactions. Consequently, ALSB does not support XA transactions for the MQ transport. If XA is required, you can use the WebSphere MQ JMS interface.

The following sections outline how ALSB connects to WebSphere MQ and presents an overview of some message types used in communication between WebSphere MQ and ALSB.

- [Using the WebSphere MQ JMS Interface](#)
- [Messaging Types](#)
- [Tuning WebSphere MQ](#)

Using the WebSphere MQ JMS Interface

ALSB connects to WebSphere MQ via the WebSphere MQ JMS interface. That is, ALSB is a WebSphere MQ JMS client.

The foreign JMS server in WLS specifies the initial context factory, connection factory, and queue to the WebSphere MQ server. For more information, see [Configuring Foreign Server Resources to Access Third-Party JMS Providers](#) in *Configuring and Managing WebLogic JMS*.

WebSphere MQ JMS supports two transport types:

- BINDINGS
- CLIENT

If the WebSphere MQ JMS client is running on the same physical machine as the queue manager, it is possible to set the transport type to BINDINGS. Otherwise, you can use only the CLIENT type.

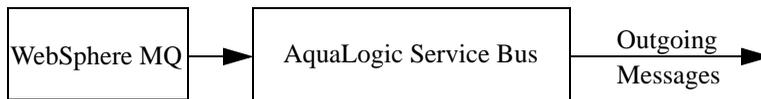
WebSphere MQ can interface with ALSB in two ways:

- ALSB acts as the front-end of WebSphere MQ to accept service requests from other applications and converts them to WebSphere MQ requests. See [Figure 3-1](#).
- WebSphere MQ sends messages to other applications through ALSB. See [Figure 3-2](#).

Figure 3-1 ALSB Front End



Figure 3-2 Messages sent through ALSB



ALSB is configured in the ALSB Console, which is described in [Using the AquaLogic Service Bus Console](#).

Messaging Types

ALSB supports the following messaging types:

- [Non-Persistent Messaging](#)
- [Non-XA Persistent Messaging](#)
- [XA Messaging](#)

Non-Persistent Messaging

If you decide to accept an unreliable delivery, such as some missing requests, you can use non-persistent messages where appropriate. WebSphere MQ logging and WebLogic JMS

message persistence are only performed for persistent messages; therefore, the use of non-persistent messages eliminates any related I/O activity.

Note: Non-persistent message throughput is usually limited by the processor speed of the machine. However, in case of a shortage of physical memory, the server system may consume CPU cycles on a paging I/O.

Non-XA Persistent Messaging

WebSphere MQ persistent message throughput is usually limited by the queue manager and the I/O latency writing to the log.

XA Messaging

To enable support for transactional (XA) access to queues, use:

- **BINDINGS** to access the queue manager when ALSB is co-located with IBM WebSphere MQ
- **CLIENT** when ALSB and IBM WebSphere MQ are on different machines. However, with **CLIENT**, you need a special version of the IBM WebSphere MQ client that supports XA transactions, called the WebSphere MQ Extended Transaction Client.

Tip: For the deployment descriptors to be set appropriately for XA capable resources (JMS, TUXEDO, EJB), you must set the XA attribute on the referenced connection factory before creating a proxy service.

Tuning WebSphere MQ

The following guidelines help you tune WebSphere MQ when you are working with ALSB. For detailed WebSphere MQ information, refer to relevant WebSphere MQ documentation.

- Use the **BINDINGS** transport type if ALSB and the queue manager are deployed on the same machine.
- If you need XA for only a small section of application requests, create a separate connection object and disable XA.
- Distribute active logs across many volumes. If your system is required to handle high persistent message throughput, you must place the log files on a fast Direct Access Storage

Device (DASD) with a minimum of contention from other data set usage. Ideally, you can allocate each of the active logs on separate, low-usage volumes.

- To reduce buffer overflow, tune the buffer pools and pagesets. Buffer overflow results in flushing of the hard disk.
- To avoid broken ALSB JMS connections to MQ queues, increase the number of active channels to more than 100. By default, the number of active channels is 10.