

Web Services SOAP/JMS Protocol Description

by: Phillip C. Adams (phil_adams@us.ibm.com)

August 9, 2004

© Copyright IBM Corporation, 2004

Introduction

The Web services engine which is delivered as part of the WebSphere Application Server product supports the use of a JMS-compliant messaging transport (as an alternative to HTTP) for communicating SOAP messages between clients and servers.

The purpose of this document is to describe the protocol used by the SOAP/JMS client and server components to exchange SOAP request and reply messages.

This description is intended to be used by customers who have a need to supply their own implementations for either the client or server components, and need to ensure that their implementations interoperate properly with the client and server components supplied with the IBM Web services engine.

This document applies to WebSphere Application Server versions 5.0.2, 5.1, and 5.1.1. Any descriptions relating to the current implementations of either the client or server components within the Web services engine apply to all these versions, unless otherwise noted.

Responsibilities of the client

The client component, which is responsible for sending SOAP request messages and receiving SOAP reply messages, must adhere to the following protocol constraints:

1. The client must use either a JMS `TextMessage` (`javax.jms.TextMessage`) or a `BytesMessage` (`javax.jms.BytesMessage`) to transmit the SOAP request message to the server. If the request message contains attachments, then a `BytesMessage` must be used. If the request message does not contain attachments, then the client may use either a `TextMessage` or a `BytesMessage`. Currently, WebSphere's client implementation uses only a `BytesMessage` due to the potential need to transmit attachments, but this may change in a future release.
2. The client must set the following properties on the JMS request message prior to sending it to the destination queue or topic:
 - **contentType** - This property is similar to the `Content-Type` header found in an HTTP message, and is used to describe the *content type* of the message. For a normal, text-only SOAP message (i.e. no attachments present), this should be something like:
`text/xml; charset="UTF-8"`
For a SOAP request message that contains attachments, the **contentType** property must be set to something like:
`multipart/related; type="text/xml"; start="<...content-id of first part...>"`
This example represents a multi-part message, where the first part is of type "text/xml" which will contain the SOAP message. The rest of the parts of the multi-part message will contain the various attachments.
The [HTTP 1.1 specification](#) contains more information about the Content-Type header.
 - **targetService** - This property must be set to the `targetService` property value which is found in the JMS-style endpoint location URL for the request. This value is used by the server component to determine which port component should be the target when dispatching the request.
 - **endpointURL** - This property must be set to the JMS endpoint URL associated with the request.
 - **transportVersion** - This property indicates the version number of the protocol used by the client and

server. This must be set to the value 1 (one). In the future, this property may be used to indicate changes to the way in which the client and server interoperate.

3. If the SOAP request message represents a two-way request, the client component must set the JMS message's *replyTo* property to specify the queue to be used for the reply message. The JMS message's `setJMSReplyTo()` method is used for this.
4. If the SOAP request message represents a one-way request, the client component **must not** set the JMS message's *replyTo* property.
5. The client component must be prepared to handle a reply message which is either a `BytesMessage` or a `TextMessage`, regardless of the type of JMS message used to transmit the SOAP request. Currently, the WebSphere implementation of the server component will respond with the same type of JMS message as was received from the client; however, this might change in the future due to enhancements to the server component.
6. The client component can assume that the correlation id of the reply message will match the message id of the original request message.

Responsibilities of the server

The server component, which is responsible for receiving SOAP request messages and sending SOAP reply messages, should adhere to the following protocol constraints:

1. The server component must be able to receive a `TextMessage` or a `BytesMessage`. Currently, the WebSphere implementation of the server component will respond in kind when sending the reply message back to the client; however, this may change in the future, so it is possible that the type of JMS message used in the reply may depend solely on the type of the SOAP reply message. For example, in the future the server may send a reply using a JMS `TextMessage` if there are no attachments contained in the SOAP reply message.
2. The server component must process the SOAP request properly to produce an appropriate SOAP reply message.
3. The server component must send a reply message back to the client if and only if the JMS request message's *replyTo* property is set.
4. The server component must set the following properties in the JMS reply message prior to sending it to the *replyTo* queue:
 - **contentType** - See the description for this property in the client section above.
 - The **correlation id** of the JMS reply message should be set to the message id of the original JMS request message. This is done by calling the JMS message's `setJMSCorrelationID()` method.
 - **transportVersion** - This property indicates the version number of the protocol used by the client and server. This must be set to the value 1 (one). In the future, this property may be used to indicate changes to the way in which the client and server interoperate.

Examples

SOAP Request - without attachments

The following figure contains the result of calling the JMS message's `toString()` method for a request message without attachments:

```
JMSMessage class: jms_bytes
JMSType: null
JMSDeliveryMode: 2
JMSExpiration: 0
JMSPriority: 4
JMSMessageID: ID:d438eebf04cb124aa25c5821110a134f0000000000000001
JMSTimestamp: 1092110476167
JMSCorrelationID: null
```

```
JMSDestination:    topic://NewsGroupTopic?topicSpace=FvtTopicSpace
JMSReplyTo:        null
JMSRedelivered:    false
JMS_IBM_System_MessageID: 6B6765B36943A18C_11000001
transportVersion:  1
JMSXUserID:
targetService:     NGConsumerJMS
JMSXAppID: Service Integration Bus
endpointURL: jms:/topic?
destination=jms/NewsGroupTopic&connectionFactory=jms/NewsGroupTCF&targetService=NGC
contentType: text/xml; charset=utf-8
3c736f6170656e763a456e76656c6f706520786d6c6e733a736f6170656e763d22687474703a2f2f
736368656d61732e786d6c736f61702e6f72672f736f61702f656e76656c6f70652f2220786d6c6e
...
```

The following figure contains the human-readable version of the payload from the above message:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    <postMessage><ngName xsi:type="xsd:string">news.current.events</ngName>
    <msg xsi:type="xsd:string">This is a sample news item.</msg>
  </postMessage>
</soapenv:Body>
</soapenv:Envelope>
```

SOAP Request - with attachments

The following figure contains the result of calling the JMS message's `toString()` method for a request message with attachments:

```
JMSMessage class: jms_bytes
JMSType:          null
JMSDeliveryMode:  1
JMSExpiration:    1092086312310
JMSPriority:       4
JMSMessageID:     ID:4bb64ed64e7d813d59ba5fec110a134f00000000000000001
JMSTimestamp:     1092086012310
JMSCorrelationID: null
JMSDestination:   queue://Logger_Q
JMSReplyTo:       queue://_Q_6B6765B36943A18C_00000385
JMSRedelivered:   false
JMS_IBM_System_MessageID: 6B6765B36943A18C_10000001
transportVersion: 1
JMSXUserID:
targetService:    WSLoggerJMS
JMSXAppID: Service Integration Bus
endpointURL: jms:/queue?
destination=jms/Logger_Q&connectionFactory=jms/Logger_CF&targetService=WSLoggerJMS
contentType: multipart/related; type="text/xml";
```

```

start="<945414389.1092086011970.IBM.WEBSERVICES@myhost>";      boundary="----
=_Part_0_247953397.1092086011970"
0d0a2d2d2d2d2d2d3d5f506172745f305f32343739353333339372e31303932303836303131393730
0d0a436f6e74656e742d547970653a20746578742f786d6c3b20636861727365743d5554462d380d
...

```

The following figure contains the human-readable version of the payload from the above message:

```

-----=_Part_0_247953397.1092086011970
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: binary
Content-Id: <945414389.1092086011970.IBM.WEBSERVICES@myhost1>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Header>
<p499:InternationalizationContext soapenv:mustUnderstand="0"
xmlns:p499="http://www.ibm.com/webservices/InternationalizationContext">
<Locales>
  <Locale>
    <LanguageCode>en</LanguageCode>
    <CountryCode>US</CountryCode>
  </Locale>
</Locales>
<TimeZoneId>America/Chicago</TimeZoneId>
</p499:InternationalizationContext>
</soapenv:Header>
<soapenv:Body>
  <sendJpegImage/>
</soapenv:Body>
</soapenv:Envelope>
-----=_Part_0_247953397.1092086011970
Content-Type: image/jpeg
Content-Transfer-Encoding: binary
Content-Id: <jpegImageRequest=81380956150.1092086011880.IBM.WEBSERVICES@myhost1>
<...contents of jpeg image file...>

```

SOAP Response

The following figure contains the result of calling the JMS message's `toString()` method for a SOAP reply message:

```

JMSMessage class: jms_bytes
JMSType: null
JMSDeliveryMode: 2
JMSExpiration: 0
JMSPriority: 4
JMSMessageID: null
JMSTimestamp: 0
JMSCorrelationID: ID:cdddb857f078a266eb9a972f110a134f0000000000000001
JMSDestination: null
JMSReplyTo: null
JMSRedelivered: false
contentType: multipart/related; type="text/xml";
start="<961368106530.1092112854745.IBM.WEBSERVICES@yackerjr>";      boundary="----
=_Part_0_1655006754.1092112854745"
0d0a2d2d2d2d2d2d3d5f506172745f305f313635353030363735342e313039323131323835343734
350d0a436f6e74656e742d547970653a20746578742f786d6c3b20636861727365743d5554462d38
...

```

Related references

- [HTTP 1.1 specification](#)
- [Java Message Service 1.0.2 API specification](#)
- [SOAP 1.1 specification](#)

Trademark attributions and copyrights

© IBM Corporation 1994-2004.

All rights reserved.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp. IBM Web site pages may contain other proprietary notices and copyright information which should be observed.

[\(Back to the top\)](#)