

JMS Message in Detail

A JMS message is composed of the following parts

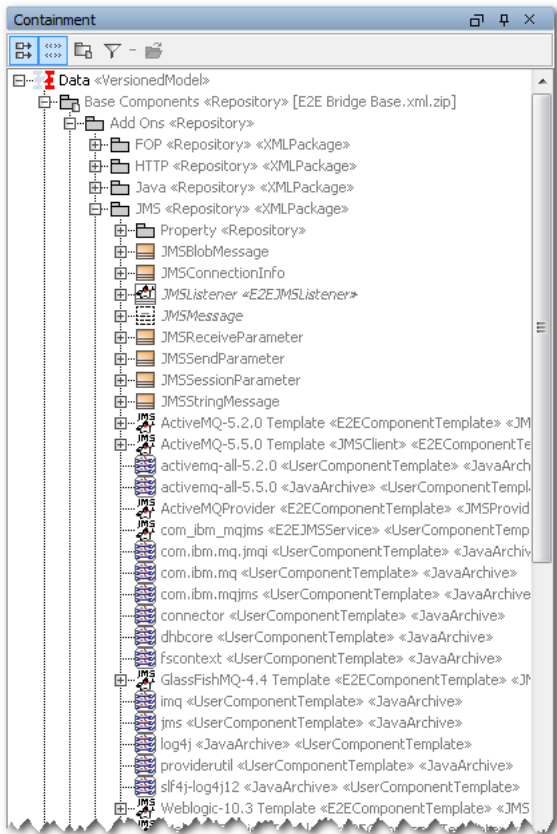
Name	Description	Class
Header	All JMS messages support the same set of header fields. Header fields contain values used by both clients and providers to identify and route messages.	JMSMessage
Properties	In addition to the standard header fields, JMS messages provide a built-in facility for adding optional header fields to a message.	JMSMessage
Body	JMS defines several types of message body, which cover the majority of messaging styles currently in use.	JMSBlobMessage and JMSStringMessage

On this Page:

- [JMS Message Header Fields](#)
- [JMS Message Properties](#)
- [JMS Message Body](#)

The folder Data / Base Components / Add-Ons / JMS in the containment tree contains UML representations of all relevant JMS message types. E2E Builder links these base components during the creation of a model from the E2E Bridge template (see figure below).

Figure: JMS Base Components



JMS Message Header Fields

Figure: Attributes of the Base Component JMSMessage

JMSMessage
+booleanProperties : JMSBooleanProperty [0..*] +byteProperties : JMSByteProperty [0..*] +doubleProperties : JMSDoubleProperty [0..*] +floatProperties : JMSFloatProperty [0..*] +integerProperties : JMSIntegerProperty [0..*] +JMSCorrelationID : String +JMSDestination : String +JMSExpiration : Integer +JMSMessageID : String +JMSPersistentDeliveryMode : Boolean = true +JMSPriority : Integer = 4 +JMSRedelivered : Boolean +JMSReplyTo : String +JMSTimestamp : Integer +JMSType : String +longProperties : JMSLongProperty [0..*] +shortProperties : JMSShortProperty [0..*] +stringProperties : JMSStringProperty [0..*]

As illustrated in figure above, the JMS adapter provides the following header fields:

Name	Type	Description	Allowed Values	
JMSCorrelationID	String	A client can use this header field to link one message with another. A typical use is to link a response message with its request message. You can override JMSCorrelationID with a custom value.		
JMSPersistentDeliveryMode	Boolean	This header field contains the delivery mode specified when the message is sent.	true	The JMS provider has to take extra care to ensure that the message is not lost in transit due to a JMS provider failure (default). The JMS provider must deliver the message <i>once and only once</i> .
			false	The JMS provider may lose the message due to a JMS provider failure.
JMSDestination	String	This header field contains the destination, to which the message is sent.		
JMSExpiration	Integer	When a message is sent, its expiration time is calculated as the sum of the time-to-live value specified on the send method and the current GMT value. On return from the send method, the JMSExpiration header field of the message contains this value. When a message is received, its JMSExpiration header field contains the same value.	0	The JMS does not expire.
			>0	The JMS message expires after the given period of time.
JMSMessageID	String	This header field contains a value that uniquely identifies each message sent by a provider. A JMSMessageID is a string value, which should function as a unique key for identifying messages in a historical repository. A provider defines the exact scope of uniqueness. Combined with JMSCorrelationID , this header field is used to link messages, and thus to implement synchronous messaging.		
JMSPriority	Integer	This header field contains the message's priority. JMS defines a ten-level priority value, with 0 as the lowest priority and 9 as the highest.	0-4	The JMS message has a <i>normal</i> priority (4 is the default value).

			5 -9	The JMS message has an <i>expedited</i> priority.
JMSRedelivered	Boolean	If a client receives a message with the JMSRedelivered indicator set to true, it is likely, but not guaranteed, that this message was delivered, but not acknowledged, in the past. In general, a provider must set this message header field to true whenever he redelivers a message.	true	The JMS message has been redelivered. This is an indication to the consuming application that the message may have been delivered in the past and that the application should take extra precautions to prevent duplicate processing.
			false	The JMS message has not been redelivered.
JMSReplyTo	String	This header field contains the name of a destination, to which the recipient of the message should send a reply. The client sending the message specifies the JMSReplyTo field value.		
JMSTimestamp	Integer	This header field contains the time a client passes the message to a provider to be sent. The time is not the actual transmission time of the message; the actual send may occur later due to transactions or other client side queuing of messages.		
JMSType	String	This header field contains a message type identifier supplied by a client when the message is sent. Some messaging systems require that a message type definition is created for each application message and that each message specifies its type. In order to work with such JMS providers, JMS clients should assign a value to JMSType whether the application makes use of it or not. The assignment of a value ensures that the field is set properly for those providers that require a JMSType value.		
JMSCorrelationIDAsBytes	Blob	Contains the JMSCorrelationID in Blob format.		

Most JMS headers are automatically assigned. Their value is set by the JMS provider when the message is delivered. Nevertheless, some headers can be overwritten by the developer. These fields are marked in bold in the table above.

JMS Message Properties

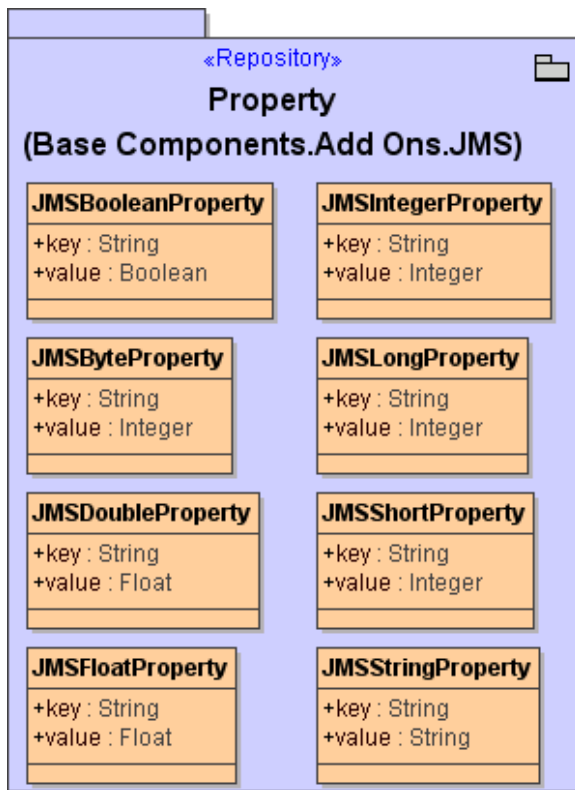
In addition to the header fields described before, the JMS Message interface contains a built-in facility for supporting property values. In effect, this facility provides a mechanism for adding optional header fields to a message.

As you can see in the figure below, the JMSMessage class contains a number of attributes, which can hold arrays of properties. Subfolder **Property** in **Base Components/Add-Ons/JMS** contains all available types of properties supported by the Bridge.

Properties and header fields allow a JMS client to have a JMS provider select messages on its behalf using application-specific criteria. This is communicated to the JMS Provider via message selectors (refer to the official Java Message Service Specification for selector statement syntax). The xUML examples include a message selector example.

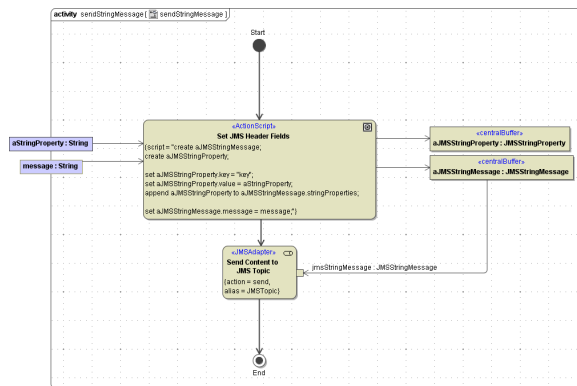
JMS Properties in the Bridge are objects composed of a key-value-pair of attributes. The key attribute is of type String, contains the property name, and must obey the rules for a message selector identifier. The native Java types of the attribute value can be of type Boolean, Byte, Double, Float, Integer, Long, Short, or String. They are mapped to Bridge base types.

Figure: JMS Properties



For best performance, message properties should only be used if the header of a message needs to be customized. The primary reason for using message properties is to support customized message selection.

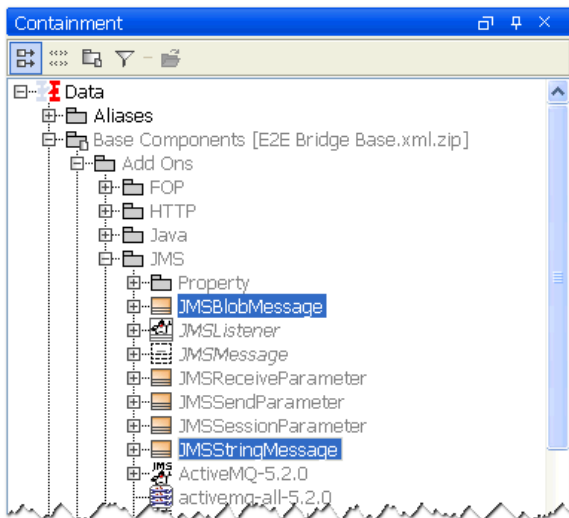
Figure: How to Set Properties in an Activity



JMS Message Body

The JMS adapter fully supports two types of a JMS message body: Byte and Text. Bridge type **JMSBlob Message** corresponds to JMS message type **ByteMessage**; **JMSStringMessage** corresponds to JMS message type **TextMessage** (see figure below).

Figure: JMS Message Types Supported by the E2E Bridge



The JMS adapter does not fully support the other JMS message types. It cannot make available the body content of an incoming message of an unsupported message type.

If required, the JMS adapter converts the content of a message body of type String into an object of type Blob and vice versa. However, the Java Virtual Machine (JVM) uses the system encoding for the conversion which often leads to undesired effects. You can avoid those by using **JMSBlobMessage** for byte messages and **JMSStringMessage** for string messages in the xUML model.

As an alternative to creating a **JMSStringMessage** or **JMSBlobMessage** object and setting its respective attributes, the JMS adapter accepts messages of type **String** or **Blob**.