Service Component Architecture JMS Binding Specification Version 1.1

Committee Draft 022 revision 4

19th June16th February, 2009

Specification URIs:
This Version:

http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec-cd02-rev4.html
http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec-cd02-rev4.doc
http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec-cd02-rev4.pdf
(Authoritative)

Previous Version:

http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec-cd02.html
http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec-cd02.doc
http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec-cd02.pdf
(Authoritative)

Latest Version:

http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec.html
http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec.doc
http://docs.oasis-open.org/opencsa/sca-bindings/sca-binding-jms-1.1-spec.pdf (Authoritative)

Latest Approved Version:

Technical Committee:

OASIS Service Component Architecture / Bindings (SCA-Bindings) TC

Chair(s):

Simon Holdsworth, IBM

Editor(s):

Simon Holdsworth, IBM
Khanderao Kand, Oracle
Anish Karmarkar, Oracle
Sanjay Patil, SAP
Piotr Przybylski, IBM

Related work:

This specification replaces or supersedes:

- Service Component Architecture JMS Binding Specification Version 1.00, March 21 2007

This specification is related to:

- Service Component Architecture Assembly Model Specification Version 1.1
- Service Component Architecture Policy Framework Specification Version 1.1

Declared XML Namespace(s):

http://docs.oasis-open.org/ns/opencsa/sca/200903

Deleted: 0
Deleted: 1
Deleted: 21st January
Deleted: 1
Field Code Changed
Field Code Changed
Deleted: 1
Field Code Changed
Deleted: 1
Field Code Changed
Deleted: jmsbinding
Deleted: cd01
Field Code Changed
Deleted: jmsbinding
Deleted: cd01
Field Code Changed
Deleted: jmsbinding
Deleted: cd01
Deleted: supercedes
Deleted: 200712
Deleted: 1
Deleted: 21st January
Deleted: 2008.

Abstract:

This document defines the concept and behavior of a messaging binding, and a concrete JMS-based binding that provides that behavior.

The binding specified in this document applies to an SCA composite's services and references. The binding is especially well suited for use by services and references of composites that are directly deployed, as opposed to composites that are used as implementations of higher-level components. Services and references of deployed composites become system-level services and references, which are intended to be used by non-SCA clients.

The messaging binding describes a common pattern of behavior that may be followed by messaging-related bindings, including the JMS binding. In particular it describes the manner in which operations are selected based on message content, and the manner in which messages are mapped into the runtime representation. These are specified in a language-neutral manner.

The JMS binding provides JMS-specific details of the connection to the required JMS resources. It supports the use of Queue and Topic type destinations.

Status:

This document was last revised or approved by the OASIS Service Component Architecture / Bindings (SCA-Bindings) TC on the above date. The level of approval is also listed above. Check the "Latest Version" or "Latest Approved Version" location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the "Send A Comment" button on the Technical Committee's web page at http://www.oasis-open.org/committees/sca-bindings/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (http://www.oasis-open.org/committees/sca-bindings/ipr.php.

The non-normative errata page for this specification is located at http://www.oasis-open.org/committees/sca-bindings/.

Deleted: 1

Deleted: 21st January

Deleted: 2008.

Notices

Deleted: 2008

Deleted: 1

Deleted: 21st January

Deleted: 2008.

Table of Contents

Field Code Changed ... [1]
Field Code Changed ... [2]
Field Code Changed ... [3]
Field Code Changed ... [4]
Field Code Changed ... [5]
Deleted: 2 . Messaging ... [6]
Formatted ... [7]
Field Code Changed ... [8]
Field Code Changed ... [9]
Deleted: 3 . JMS Bindi ... [10]
Field Code Changed ... [11]
Field Code Changed ... [12]
Deleted: 4 . Operation ... [13]
Field Code Changed ... [14]
Formatted ... [15]
Field Code Changed ... [16]
Deleted: 4.1 Default O ... [17]
Field Code Changed ... [18]
Field Code Changed ... [19]
Deleted: 2…Wire Form ... [20]
Field Code Changed ... [21]
Formatted ... [22]
Field Code Changed ... [23]
Deleted: 5 . Policy .
Field Code Changed ... [24]
Formatted ... [25]
Field Code Changed ... [26]
Deleted: 6 . Message ... [27]
Field Code Changed ... [28]
Formatted ... [29]
Field Code Changed ... [30]
Deleted: 6.1 One-way ... [31]
Field Code Changed ... [32]
Field Code Changed ... [33]
Deleted: 6.2 ... [34]
Field Code Changed ... [35]
Field Code Changed ... [36]
Deleted: 6.3 JMS User ... [37]
Field Code Changed ... [38]
Field Code Changed ... [39]
Deleted: 4
Field Code Changed ... [40]
Formatted ... [41]
Field Code Changed ... [42]
Deleted: 6.4.1 Invocati ... [43]
Field Code Changed ... [44]
Field Code Changed ... [45]
Deleted: 6.4.2 Invocati ... [46]
Field Code Changed ... [47]
Field Code Changed ... [48]
Deleted: 6.4.3 Use of ... [49]
Field Code Changed ... [50]
Formatted ... [51]
Field Code Changed ... [52]
Field Code Changed ... [53]
Field Code Changed ... [54]
... [55]
Field Code Changed ... [56]

1 Introduction

This document defines the concept and behavior of a messaging binding, and a concrete JMS-based [JMS] binding that provides that behavior. The binding specified in this document applies to an SCA composite's services and references. The binding is especially well suited for use by services and references of composites that are directly deployed, as opposed to composites that are used as implementations of higher-level components. Services and references of deployed composites become system-level services and references, which are intended to be used by non-SCA clients.

The messaging binding describes a common pattern of behavior that may be followed by messaging-related bindings, including the JMS binding. In particular it describes the manner in which operations are selected based on message content, and the manner in which messages are mapped into the runtime representation. These are specified in a language-neutral manner.

The JMS binding provides JMS-specific details of the connection to the required JMS resources. It supports the use of Queue and Topic type destinations.

1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This specification uses predefined namespace prefixes throughout; they are given in the following list. Note that the choice of any namespace prefix is arbitrary and not semantically significant.

Table 1-1 Prefixes and Namespaces used in this specification

Prefix	***Namespace***	***Notes***
xs	"http://www.w3.org/2001/XMLSchema"	Defined by XML Schema 1.0 specification
sca	"http://docs.oasis-open.org/ns/opencsa/sca/200903"	Defined by the SCA specifications

Deleted: 200712"

1.2 Normative References

[RFC2119] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, http://www.ietf.org/rfc/rfc2119.txt, IETF RFC 2119, March 1997.

[JMS] JMS Specification http://java.sun.com/products/jms/

[WSDL] E. Christensen et al, *Web Service Description Language (WSDL) 1.1*, http://www.w3.org/TR/2001/NOTE-wsdl-20010315, W3C Note, March 15 2001.

R. Chinnici et al, *Web Service Description Language (WSDL) Version 2.0 Part 1: Core Language*, http://www.w3.org/TR/2007/REC-wsdl20-20070626/, W3C Recommendation, June 26 2007.

[JCA15] Java Connector Architecture Specification Version 1.5 http://java.sun.com/j2ee/connector/

Deleted: 1
Deleted: 21st January
Deleted: 2008.

[IETFJMS]	IETF URI Scheme for Java™ Message Service 1.0 http://www.ietf.org/internet-drafts/draft-merrick-jms-uri-05.txt[1]
[SCA-Assembly]	http://docs.oasis-open.org/opencsa/sca-assembly/sca-assembly-1.1-spec.html
[SCA-Policy]	http://docs.oasis-open.org/opencsa/sca-policy/sca-policy-1.1-spec.pdf

1.3 Non-Normative References

TBD	TBD

1.4 Naming Conventions

This specification follows some naming conventions for artifacts defined by the specification. In addition to the conventions defined by section 1.3 of the SCA Assembly Specification [SCA-Assembly], this specification adds three additional conventions:

- Where the names of elements and attributes consist partially or wholly of acronyms, the letters of the acronyms use the same case. When the acronym appears at the start of the name of an element or an attribute, or after a period, it is in lower case. If it appears elsewhere in the name of an element or an attribute, it is in upper case. For example, an attribute might be named "uri" or "jndiURL".
- Where the names of types consist partially or wholly of acronyms, the letters of the acronyms are in all upper case. For example, an XML Schema type might be named "JCABinding" or "MessageID".
- Values, including local parts of QName values, follow the rules for names of elements and attributes as stated above, with the exception that the letters of acronyms are in all upper case. For example, a value might be "JMSDefault" or "namespaceURI".

[1] Note that this URI scheme is currently in draft. The reference for this specification will be updated when the IETF standard is finalized

Deleted: 1
Deleted: 21st January
Deleted: 2008.

2 Messaging Bindings

Messaging bindings form a category of SCA bindings that represent the interaction of SCA composites with messaging providers. It is felt that documenting, and following this pattern is beneficial for implementers of messaging bindings, although it is not strictly necessary.

This pattern is embodied in the JMS binding, described later.

Messaging bindings utilize operation selector and wire format elements to provide the mapping from the native messaging format to an invocation on the target component. A default operation selection and data binding behavior is identified, along with any associated properties.

In addition, each operation may have specific properties defined, that may influence the way native messages are processed depending on the operation being invoked.

Comment [SAJH1]: I don't think this section really says anything that is not already said elsewhere and should be deleted

Deleted: 1

Deleted: 21st January

Deleted: 2008.

3 JMS Binding Schema

The JMS binding element is defined by the following schema.

```
<binding.jms correlationScheme="QName"?
             initialContextFactory="xs:anyURI"?
             jndiURL="xs:anyURI"?
             requestConnection="QName"?
             responseConnection="QName"?
             operationProperties="QName"?
             name="NCName"?
             requires="list of QName"?
             policySets="list of QName"?
             uri="xs:anyURI"?
             ... >
    <destination jndiName="xs:anyURI" type="queue or topic"?
                 create="always or never or ifNotExist"?>
        <property name="NMTOKEN" type="NMTOKEN"?>*
    </destination>?
    <connectionFactory jndiName="xs:anyURI"
                       create="always or never or ifNotExist"?>
        <property name="NMTOKEN" type="NMTOKEN"?>*
    </connectionFactory>?
    <activationSpec jndiName="xs:anyURI"
                    create="always or never or ifNotExist"?>
        <property name="NMTOKEN" type="NMTOKEN"?>*
    </activationSpec>?

    <response>
        <destination jndiName="xs:anyURI" type="queue or topic"?
                     create="always or never or ifNotExist"?>
            <property name="NMTOKEN" type="NMTOKEN"?>*
        </destination>?
        <connectionFactory jndiName="xs:anyURI"
                           create="always or never or ifNotExist"?>
            <property name="NMTOKEN" type="NMTOKEN"?>*
        </connectionFactory>?
        <activationSpec jndiName="xs:anyURI"
                        create="always or never or ifNotExist"?>
            <property name="NMTOKEN" type="NMTOKEN"?>*
        </activationSpec>?
        <wireFormat/>?
    </response>?

    <resourceAdapter name="NMTOKEN">?
        <property name="NMTOKEN" type="NMTOKEN"?>*
    </resourceAdapter>?

    <headers type="string"?
             deliveryMode="persistent or nonpersistent"?
             timeToLive="long"?
             priority="0 . . 9"?>
        <property name="NMTOKEN" type="NMTOKEN"?>*
    </headers>?

    <messageSelection selector="string"?>
        <property name="NMTOKEN" type="NMTOKEN"?>*
    </headers>?

    <operationProperties name="string" nativeOperation="string"?>
        <property name="NMTOKEN" type="NMTOKEN"?>*
```

Deleted: ifnotexist

Deleted: ifnotexist

Deleted: ifnotexist

Deleted: ifnotexist

Deleted: ifnotexist

Deleted: ifnotexist

Deleted: JMSType

Deleted: JMSDeliveryMode="PERSISTENT or NON_PERSISTENT"?¶ JMSTimeToLive="long"?.¶ JMSPriority="0 ..

Deleted: subscriptionHeaders JMSSelector

Deleted: 1

Deleted: 21st January

Deleted: 2008.

```
        <headers type="string"?
                 deliveryMode="persistent or nonpersistent"?
                 timeToLive="long"?
                 priority="0 .. 9"?>
            <property name="NMTOKEN" type="NMTOKEN"?>*
        </headers>?
    </operationProperties>*

    <wireFormat/>?
    <operationSelector/>?
</binding.jms>
```

The binding can be used in one of two ways, either identifying existing JMS resources using JNDI names, or providing the required information to enable the JMS resources to be created.

The ***binding.jms*** element has the following attributes:

- ***/binding.jms*** – This is the JMS binding element. The element is extensible so that JMS binding implementers can add additional JMS provider-specific attributes and elements although such extensions are not guaranteed to be portable across runtimes.
- ***/binding.jms/@uri*** – as defined in the SCA Assembly Specification [SCA-Assembly]. This attribute identifies the destination, connection factory or activation spec, and other properties to be used to send/receive the JMS message. There is an implicit ***@create="never"*** for the resources referred to in the ***@uri*** attribute.

  The value of the ***@uri*** attribute MUST have the format defined by the IETF URI Scheme for Java™ Message Service 1.0 [IETFJMS] [BJM30001].

  The following illustrates the structure of the URI and the set of property names that have specific semantics - all other property names are treated as user property names:

  – ***jms:jndi:<jms-dest>?***
  jndiURL=<jndi-url> &
  jndiInitialContextFactory=<jndi-initial-context-factory> &
  jndiConnectionFactoryName=<Connection-Factory-Name> &
  deliveryMode=<Delivery-Mode> &
  timeToLive=<Time-To-Live> &
  priority=<Priority> &
  <param-name>=<param-value> & ...

  When the ***@uri*** attribute is specified, the SCA runtime MUST raise an error if the referenced resources do not already exist [BJM30002].

  When the ***@uri*** attribute is specified, the ***destination*** element MUST NOT be present [BJM30034].

  An SCA runtime MUST use the values specified in the ***@uri*** attribute in preference to corresponding attributes and elements in the binding [BJM30035].
- ***/binding.jms/@name*** - as defined in the SCA Assembly SpecificationS [SCA-Assembly].
- ***/binding.jms/@requires*** - as defined in the SCA Assembly SpecificationS [SCA-Assembly].
- ***/binding.jms/@policySets*** - as defined in the SCA Assembly Specification [SCA-Assembly].
- ***/binding.jms/@correlationScheme*** – identifies the correlation scheme used when sending reply or callback messages, default value is "sca:messageID".
  If the value of the ***@correlationScheme*** attribute is "***sca:messageID***" the SCA runtime MUST set the correlation ID of replies to the message ID of the corresponding request [BJM30003].
  If the value of the ***@correlationScheme*** attribute is "***sca:correlationID***" the SCA runtime MUST set the correlation ID of replies to the correlation ID of the corresponding request [BJM30004].
  If the value of the ***@correlationScheme*** attribute is "***sca:none***" the SCA runtime MUST NOT set the correlation ID[BJM30005].
  SCA runtimes MAY allow other values of the ***@correlationScheme*** attribute to indicate other correlation schemes [BJM30006].
- ***/binding.jms/@initialContextFactory*** – the name of the JNDI initial context factory.

Deleted: JMSType

Deleted: JMSDeliveryMode="PERSISTENT or NON_PERSISTENT"?¶ JMSTimeToLive="long"?¶ JMSPriority="0 ..

Deleted: generic

Deleted: type.

Deleted: type

Deleted: (from binding)

Deleted: URI that

Deleted: . . The value of the ***@uri*** attribute MUST have the following format, defined by the IETF URI Scheme

Deleted: Java™ Message Service 1.0

Field Code Changed

Deleted: .

Deleted: ***connectionFactoryName***

Deleted: ***destinationType={queue|topic}*** .

Deleted: ***selector=<Selector>*** & . ***<User-Property>=<User-Property-Value***

Deleted: When the ***@uri*** attribute is specified, the SCA runtime MUST raise an error if the referenced resources do not already exist.¶

Deleted: specification

Deleted: in Section 9, "Binding"

Deleted: specification

Deleted: in Section 9, "Binding"

Deleted: . Possible values for the ***@correlationScheme*** attribute are "***sca:MessageID***" (the default) where the SCA runtime MUST set the correlation ID of replies to the message ID of the corresponding request; "***sca:CorrelationID***" where the SCA runtime MUST set the correlation ID of replies ... [63]

Deleted: 1

Deleted: 21st January

Deleted: 2008.

- ***/binding.jms/@jndiURL*** – the URL for the JNDI provider.
- ***/binding.jms/@requestConnection*** – identifies a ***binding.jms*** element that is present in a definition document, whose ***destination***, ***connectionFactory***, ***activationSpec*** and ***resourceAdapter*** children are used to define the values for this binding.
  If the ***@requestConnection*** attribute is specified, the ***binding.jms*** element MUST NOT contain a ***destination***, ***connectionFactory***, ***activationSpec*** or ***resourceAdapter*** element [BJM30007].
- ***/binding.jms/@responseConnection*** – identifies a ***binding.jms*** element that is present in a definition document, whose ***response*** child element is used to define the values for this binding.
  If the ***@responseConnection*** attribute is specified, the ***binding.jms*** element MUST NOT contain a ***response*** element [BJM30008].
- ***/binding.jms/@operationProperties*** – identifies a ***binding.jms*** element that is present in a definition document, whose ***operationProperties*** children are used to define the values for this binding.
  If the ***@operationProperties*** attribute is specified, the ***binding.jms*** element MUST NOT contain an ***operationProperties*** element [BJM30009].
- ***/binding.jms/destination*** – identifies the destination that is to be used to process requests by this binding.
- ***/binding.jms/destination/@type*** - the type of the request destination. Valid values are "***queue***" and "***topic***". The default value is "***queue***".
  Whatever the value of the ***destination/@type*** attribute, the runtime MUST ensure a single response is delivered for request/response operations [BJM30010].
- ***binding.jms/destination/@jndiName*** – the JNDI name of the JMS Destination that the binding uses to send or receive messages. The behaviour of this attribute is determined by the value of the ***@create*** attribute as follows:
  - If the ***@create*** attribute value for a destination, connectionFactory or activationSpec element is "***always***" then the ***@jndiName*** attribute is optional; if the resource cannot be created at the specified location then the SCA runtime MUST raise an error [BJM30011].
    If the ***@jndiName*** attribute is omitted this specification places no restriction on the JNDI location of the created resource.
  - If the ***@create*** attribute value for a destination, connectionFactory or activationSpec element is "***ifNotExist***" then the ***@jndiName*** attribute MUST specify the location of the possibly existing resource [BJM30012].
    If the destination, connectionFactory or activationSpec does not exist at the location identified by the ***@jndiName*** attribute, but cannot be created there then the SCA runtime MUST raise an error [BJM30013].
    If the destination, connectionFactory or activationSpec's ***@jndiName*** attribute refers to an existing resource that is not a JMS Destination of the appropriate type, a JMS connection factory or a JMS activation spec respectively then the SCA runtime MUST raise an error [BJM30014].
  - If the ***@create*** attribute value for a destination, connectionFactory or activationSpec element is "***never***" then the ***@jndiName*** attribute MUST specify the location of the existing resource [BJM30015].
    If the destination, connection factory or activation spec is not present at the location identified by the ***@jndiName*** attribute, or the location refers to a resource of an incorrect type then the SCA runtime MUST raise an error [BJM30016].
- ***/binding.jms/destination/@create*** – indicates whether the destination should be created when the containing composite is deployed. Valid values are "***always***", "***never***" and "***ifNotExist***". The default value is "***ifNotExist***".
- ***/binding.jms/destination/property*** – defines properties to be used to create the destination, if required.
- ***/binding.jms/connectionFactory*** – identifies the connection factory that the binding uses to process request messages. The attributes of this element follow the rules defined for the ***destination*** element.
  A ***binding.jms*** element MUST NOT include both a ***connectionFactory*** element and an

Deleted: In this case this ***binding.jms*** element MUST NOT also contain the corresponding elements

Deleted: In this case this ***binding.jms*** element MUST NOT contain a ***response*** element

Deleted: In this case this ***binding.jms*** element MUST NOT contain an ***operationProperties*** element

Deleted: In either case the runtime MUST ensure a single response is delivered for request/response operations

Deleted: If the ***@create*** attribute value is "***always***" then the ***@jndiName*** attribute is optional; if the destination cannot be created at the specified location then the SCA runtime MUST raise an error.

Formatted: Attribute, English (U.S.)

Deleted: <#>If the ***@create*** attribute value is "***ifnotexist***" then the ***@jndiName*** attribute MUST specify the location of the possibly existing destination; if the destination does not exist at this location, but cannot be created there then the SCA runtime MUST raise an error. If the ***@jndiName*** refers to an existing resource other than a JMS Destination of the specified type then the SCA runtime MUST raise an error. ¶ <#>If the ***@create*** attribute value is "***never***" then the ***@jndiName*** attribute MUST specify the location of the existing destination; If the destination is not present at the location, or the location refers to a resource other than a JMS Destination of the specified type then the SCA runtime MUST raise an error.¶

Formatted: Attribute, English (U.S.)

Formatted: Attribute, English (U.S.)

Deleted: ***ifnotexist***".

Deleted: ***ifnotexist***"..

Deleted: those

Deleted: 1

Deleted: 21st January

Deleted: 2008.

activationSpec element [BJM30017].
When the ***connectionFactory*** element is present, then the destination MUST be defined either by the ***destination*** element or the ***@uri*** attribute [BJM30018].

- ***/binding.jms/activationSpec*** – identifies the activation spec that the binding uses to connect to a JMS destination to process request messages. The attributes of this element follow the rules defined for the ***destination*** element.
  If the ***activationSpec*** element is present and the destination is also specified via a ***destination*** element or the ***@uri*** attribute then it MUST refer to the same JMS destination as the ***activationSpec*** [BJM30019].
  The ***activationSpec*** element MUST NOT be present when the binding is being used for an SCA reference [BJM30020].
- ***/binding.jms/response*** – defines the resources used for handling response messages (receiving responses for a reference, and sending responses from a service).
- ***/binding.jms/response/destination*** – identifies the destination that is to be used to process responses by this binding. Attributes follow the rules defined for the parent's ***destination*** element. For a service, this destination is used to send responses to messages that have a null value for the ***JMSReplyTo*** destination. For a reference, this destination is used to receive reply messages
- ***/binding.jms/response/connectionFactory*** – identifies the connection factory that the binding uses to process response messages. The attributes of this element follow those defined for the ***destination*** element.
  A ***response*** element MUST NOT include both a ***connectionFactory*** element and an ***activationSpec*** element [BJM30021].
- ***/binding.jms/response/activationSpec*** – identifies the activation spec that the binding uses to connect to a JMS destination to process response messages. The attributes of this element follow those defined for the ***destination*** element.
  If a ***response/destination*** and ***response/activationSpec*** element are both specified they MUST refer to the same JMS destination [BJM30022].
  The ***response/activationSpec*** element MUST NOT be present when the binding is being used for an SCA service [BJM30023].
- ***/binding.jms/response/wireFormat*** – identifies the wire format used by responses sent or received by this binding. This value overrides the ***wireFormat*** specifed at the binding level.
- ***/binding.jms/headers*** – this element specifies values for standard JMS headers.
  The SCA runtime MUST set JMS headers in messages that it creates to the values specified by the ***headers*** element unless overridden for the operation being invoked. [BJM30024].
  These values apply to requests from a reference and responses from a service.
- ***/binding.jms/headers/@type, @deliveryMode, @timeToLive, @priority*** – specifies the value to use for the JMS header property JMSType, JMSDeliveryMode, JMSTimeToLive or JMSPriority respectively.
  If the ***@uri*** attribute includes values for the type, delivery mode, time to live or priority properties then the @uri values are used and the ***headers*** and ***operationProperties/headers @type, @deliveryMode, @timeToLive*** or ***@priority*** attributes are ignored [BJM30025].
  Valid values for ***@deliveryMode*** are "***persistent***" and "***nonpersistent***"; valid values for ***@priority*** are "***0***" to "***9***".
- ***/binding.jms/headers/property*** – specifies the value for the given JMS user property.
  For each ***header/properties*** element the SCA runtime MUST set the named JMS user property to the given value in messages it creates unless overridden for the operation being invoked [BJM30026].
- ***/binding.jms/messageSelection*** - this element allows JMS message selection options to be set. These values apply to a service receiving messages from the request destination or for a reference receiving messages from the callback or reply-to destination.
- ***/binding.jms/messageSelection/@selector*** - specifies the value to use for the JMS selector. If the ***@uri*** attribute includes a value for the message selector then the ***@uri*** value is used and the ***messageSelection/@selector*** attribute is ignored [BJM30027].

Deleted: A ***binding.jms*** element MUST NOT include both this element and an ***activationSpec*** element. When this element is present, the ***destination*** element MUST also be present

Deleted: those

Deleted: If a ***destination*** element is also specified it MUST refer to the same JMS destination as the ***activationSpec***. This element MUST NOT be present when the binding is being used for an SCA reference

Deleted: are as

Deleted: A ***response*** element MUST NOT include both this element and an ***activationSpec*** element

Deleted: If a response ***destination*** element is also specified it MUST refer to the same JMS destination as the ***activationSpec***. This element MUST NOT be present when the binding is being used for an SCA service

Deleted: that the SCA runtime MUST set to the given v ... [64]

Deleted: ***JMSType, @JMSDeliveryMode,*** ... [65]

Deleted: . The value of the ***@uri*** attribute MUST N ... [66]

Deleted: ***JMSDeliveryMode*** are "***PERSISTENT***" and ... [67]

Deleted: ***JMSPriority***

Deleted: that the SCA runtime MUST set

Deleted: specified

Deleted: when creating messages..

Deleted: ***subscriptionHeaders***

Deleted: subscription

Deleted: subscribing to the

Deleted: subscribing to the

Deleted: s

Deleted: ***subscriptionHeaders/@JMSSelector***

Deleted: The value of the ***@uri*** attribute MUST N ... [68]

Deleted: 1

Deleted: 21st January

Deleted: 2008.

- ***/binding.jms/resourceAdapter*** – specifies name, type and properties of the Resource Adapter Java bean.
  The ***resourceAdapter*** element MUST be present when JMS resources are to be created for a JMS provider that implements the JCA 1.5 specification [JCA15], and is ignored otherwise [BJM30031].
  SCA runtimes MAY place restrictions on the properties of the resource adapter Java bean that can be set using the ***resourceAdapter*** element [BJM30028].
  For JMS providers that do not implement the JCA 1.5 specification JCA15, information necessary for resource creation can be added in provider-specific elements or attributes allowed by the extensibility of the ***binding.jms*** element.
- ***/binding.jms/operationProperties*** – specifies various properties that are specific to the processing of a particular operation.
- ***/binding.jms/operationProperties/@name*** – The name of the operation in the interface.
- ***/binding.jms/operationProperties/@selectedOperation*** – The value generated by the ***operationSelector*** that corresponds to the operation in the service or reference interface identified by the ***operationProperties/@name*** attribute. If this attribute is omitted then the value defaults to the value of the ***operationProperties/@name*** attribute.
  The value of the ***operationProperties/@selectedOperation*** attribute MUST be unique across the containing ***binding.jms*** element [BJM30029].
- ***/binding.jms/operationProperties/property*** – specifies properties specific to this operation. These properties are intended to be used to parameterize the ***wireFormat*** identified for the binding for a particular operation.
  The SCA runtime SHOULD make the ***operationProperties*** element corresponding to the ***selectedOperation*** available to the ***wireFormat*** implementation [BJM30030].
- ***/binding.jms/operationProperties/headers*** – this element specifies values for standard JMS headers. These values apply to requests from a reference and responses from a service.
  The SCA runtime MUST set JMS headers in messages it creates when the operation identified by the ***operationProperties/@name*** attribute is invoked to the values specified by the corresponding ***operationProperties/headers*** element [BJM30032].
- ***/binding.jms/operationProperties/headers/@type, @deliveryMode, @timeToLive, @priority*** – specifies the value to use for the JMS header property JMSType, JMSDeliveryMode, JMSTimeToLive or JMSPriority, respectively.
- ***/binding.jms/operationProperties/headers/property*** – specifies the value for the given JMS user property.
  For each ***operationProperties/headers/property*** element the SCA runtime MUST set the named JMS user property to the given value in messages it creates when the operation identified by the ***operationProperties/@name*** attribute is invoked [BJM30033].
- ***/binding.jms/wireFormat*** – identifies the wire format used by requests and responses sent or received by this binding.
- ***/binding.jms/operationSelector*** – identifies the operation selector used when receiving requests for a service. If specified for a reference this provides the default operation selector for callbacks if not specified via a callback service element.
- ***/binding.jms/@{any}*** - this is an extensibility mechanism to allow extensibility via attributes.
- ***/binding.jms/any*** – this is an extensibility mechanism to allow extensibility via elements.

The ***binding.jms*** element MUST conform to the XML schema defined in sca-binding-jms.xsd [BJM30036].

Deployers/assemblers can configure ***nonpersistent*** for ***@deliveryMode*** in order to provide higher performance with a decreased quality of service. A ***binding.jms*** element configured in this way cannot satisfy either of the "***atLeastOnce***" and "***exactlyOnce***" policy intents. The SCA Runtime MUST raise an error for this invalid combination at deployment time.

Deleted: This element MUST be present when the JMS resources are to be created for a JMS provider that implements the JCA 1.5 specification

Deleted: JCA15

Deleted: , and is ignored otherwise. SCA runtimes MAY place restrictions on the properties of the RA Java bean that can be set.

Deleted: The value of this attribute MUST be unique across the containing ***binding.jms*** element..

Deleted: The SCA runtime SHOULD make the ***operationProperties*** element corresponding to the ***selectedOperation*** available to the ***wireFormat*** implementation

Deleted: that the SCA runtime MUST set to the given values for the given operation.

Deleted: ***JMSType, @JMSDeliveryMode, @JMSTimeToLive, @JMSPriority***

Deleted: . The SCA runtime MUST use values specified for particular operations in preference to those defined for all operations in the ***binding.jms/headers*** element or via the binding's ***@uri*** attribute.

Deleted: that

Deleted: SCA runtime MUST set for the specified

Deleted: when creating messages

Deleted: ***NON_PERSISTENT***

Deleted: ***JMSDeliveryMode***

Comment [SAJH2]: Pending move of this text for resolution to issue BINDINGS-48

Deleted: 1

Deleted: 21st January

Deleted: 2008.

4 Operation Selectors and Wire Formats

In general messaging providers deal with message formats and destinations. There is not usually a built-in concept of "operation" that corresponds to that defined in a WSDL portType [WSDL]. Messages have a wire format which corresponds in some way to the schema of an input or output message of an operation in the interface of a service or reference, however additional information is required in order for an SCA runtime to know how to identify the operation and understand the wire format of messages.

The process of identifying the operation to be invoked is *operation selection*; the information that describes the contents of messages is a *wire format*. The ***binding*** element as described in the SCA Assembly Specification [SCA-Assembly] provides the means to identify specific operation selection via the ***operationSelector*** element and the wire format of messages received and to be sent using the ***wireFormat*** element. When the JMS binding receives a message, the ***operationSelector*** is used to generate a selected operation name from the message content. The selected operation name is then mapped to an operation in the service's interface via a matching ***operationProperties*** element in the JMS binding. If there is no matching element, the operation name is assumed to be the same as the selected operation name.

No standard means is provided for linking the ***wireFormat*** or ***operationSelector*** elements with the runtime components that implement their behavior.

The following sections describe the default ***operationSelector*** and ***wireFormat*** for a JMS binding.

The SCA runtime MUST support the default JMS wire format and operation selector behavior, and MAY provide additional means to override it [BJM40001].

4.1 Default Operation Selection

The following defines the **default operation selection algorithm** when receiving a request at a service, or a callback at a reference. When using the default operation selection algorithm, the selected operation name is determined as follows:

- If there is only one operation on the service's interface, then that operation is the selected operation name;
- Otherwise, if the JMS user property "***scaOperationName***" is present, then the value of that user property is used as the selected operation name;
- Otherwise, if the message is a JMS text or bytes message containing XML, then the selected operation name is the local name of the root element of the XML payload;
- Otherwise, the selected operation name is "***onMessage***".

When a ***binding.jms*** element specifies the ***operationSelector.jmsDefault*** element, the SCA runtime MUST use the default operation selection algorithm to determine the selected operation [BJM40008].

If no ***operationSelector*** element is specified then SCA runtimes MUST use ***operationSelector.jmsDefault*** as the default [BJM40002].

4.2 Default Wire Format

The default wire format maps between a ***JMSMessage*** and the object(s) expected by the component implementation. We encourage component implementers to avoid exposure of JMS APIs to component implementations, however in the case of an existing implementation that expects a ***JMSMessage***, this provides for simple reuse of that as an SCA component.

When using the default wire format, the message body is mapped to the parameters or return value of the target operation as follows:

- If there is a single parameter that is a ***JMSMessage***, then the ***JMSMessage*** is passed as is.
- Otherwise, if the ***JMSMessage*** is not a JMS text message or bytes message containing XML it is invalid.

Deleted: specification

Deleted: behaviour

Deleted: This section describes

Deleted: The SCA runtime MUST support this default behavior, and MAY provide additional means to override it.

Deleted: When

Deleted: assumed as

Deleted: .

Deleted: its

Deleted: .

Deleted: taken from

Deleted: .

Deleted: assumed to be

Deleted: The selected operation name is then mapped to an operation in the service's interface via a matching ***operationProperties*** element in the JMS binding. If there is no matching element, the operation name is assumed to be the same as the selected operation name.¶
The use of this operation selector can be explicitly specified in a ***binding.jms*** using the ***operationSelector.jmsdefault*** element; if no ***operationSelector*** element is specified then SCA runtimes MUST use this as the default.¶

Deleted: The

Deleted: must be

Deleted: ; an SCA runtime MUST be able to receive both forms. When sending messages either form may be used; an SCA runtime MAY provide additional configuration to allow one or other to be selected

Deleted: 1

Deleted: 21st January

Deleted: 2008.

- Otherwise if there is a single parameter, or for the return value, the JMS text or bytes XML payload is the XML serialization of that parameter according to the WSDL schema for the message.
- Otherwise the multiple parameters are encoded in XML using the document wrapped style, according to the WSDL schema for the message.

When a ***binding.jms*** element specifies the ***wireFormat.jmsDefault*** element, the SCA runtime MUST use the default wire format [BJM40009].

When using the default wire format to send request messages, if there is a single parameter and the interface includes more than one operation, the SCA runtime MUST set the JMS user property "***scaOperationName***" to the name of the operation being invoked [BJM40003].

When using the default wire format an SCA runtime MUST be able to receive both JMS text and bytes messages [BJM40005].

When using the default wire format an SCA runtime MUST send either a JMS text or a JMS bytes message [BJM40006].

When using the default wire format an SCA runtime MAY provide additional configuration to allow selection between JMS text or bytes messages to be sent [BJM40007].

If no ***wireFormat*** element is specified in a JMS binding then SCA runtimes MUST use ***wireFormat.jmsDefault*** as the default [BJM40004].

4.2.1 Example of default wire format

For the following interface definition:

```
<wsdl:definitions name="Coordinates"
targetNamespace="http://tempuri.org/coordinates"
xmlns:tns="http://tempuri.org/coordinates"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:types>
    <xsd:schema targetNamespace="http://tempuri.org/coordinates">
      <xsd:element name="setCoordinates">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="x" type="xsd:int"/>
            <xsd:element name="y" type="xsd:int"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>
  </wsdl:types>

  <wsdl:message name="setCoordinatesRequestMsg">
    <wsdl:part element="tns:setCoordinates" name="setCoordinatesParameters"/>
  </wsdl:message>

  <wsdl:portType name="Coordinates">
    <wsdl:operation name="setCoordinates">
      <wsdl:input message="tns:setCoordinatesRequestMsg"
name="setCoordinatesRequest"/>
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```

When the ***setCoordinates*** operation is invoked via a reference with a JMS binding that uses the default wire format, the message sent from the JMS binding is a JMS text or bytes message with the following content:

```
<setCoordinates xmlns="http://tempuri.org/coordinates">
  <x>10</x>
  <y>5</y>
```

Deleted: If

Deleted: If there are

Deleted: , then they

Deleted: <#>When sending request messages, if there is a single parameter and the interface includes more than one operation, the SCA runtime MUST set the JMS user property "***scaOperationName***" to the name of the operation being invoked.¶
The use of this wire format can be explicitly specified in a ***binding.jms*** using the ***wireFormat.jmsdefault*** element; if no ***wireFormat*** element is specified then SCA runtimes MUST use this as the default. ¶

Deleted: example, for

Deleted: ¶

Deleted: 1

Deleted: 21st January

Deleted: 2008.

```
</setCoordinates>
```


Deleted: 1
Deleted: 21st January
Deleted: 2008.

5 Policy

The JMS binding provides attributes that control the sending of messages, requests from references and replies from services. These values can be set directly on the binding element for a particular service or reference, or they can be set using policy intents. An example of setting these via intents is shown later.

JMS binding implementations MAY support the following standard intents, as defined by the JMS binding's ***bindingType***:

```
<bindingType type="binding.jms"
             alwaysProvides="JMS"
             mayProvide="atLeastOnce atMostOnce ordered"/>
```

The atLeastOnce, atMostOnce and ordered intent are defined in the SCA Policy Specification document in section 8, "Reliability Policy".

Comment [SAJH3]: Pending update to this text for resolution to issue BINDINGS-48

Deleted: jms"

Deleted: conversational

Deleted: The conversational intent is defined in the SCA Assembly Specification document in section 8.3, "Conversational Interfaces".

Deleted: 1

Deleted: 21st January

Deleted: 2008.

6 Message Exchange Patterns

This section describes the message exchange patterns that are possible when using the JMS binding, including one-way, request/response and callbacks. JMS has a looser concept of message exchange patterns than WSDL, so this section explains how JMS messages that are sent and received by the SCA runtime relate to the WSDL input/output messages. Each operation in a WSDL interface is either one-way or request/response. Callback interfaces may include both one-way and request/response operations.

6.1 One-way message exchange (no Callbacks)

A one-way message exchange is one where a request message is sent that does not require or expect a corresponding response message. These are represented in WSDL as an operation with an ***input*** element and no ***output*** elements and no ***fault*** elements.

For an SCA reference with a JMS binding, when a request message is sent as part of a one-way MEP, the SCA runtime SHOULD NOT set the ***JMSReplyTo*** destination header in the JMS message that it creates, regardless of whether the JMS binding has a ***response*** element with a ***destination*** defined [BJM60001].

For an SCA service with a JMS binding, when a request message is received as part of a one-way MEP, the SCA runtime MUST ignore the ***JMSReplyTo*** destination header in the JMS message, and not raise an error [BJM60002].

The use of one-way exchanges when using a bidirectional interface is described in section 6.4.

6.2 Request/response message exchange (no Callbacks)

A request/response message exchange is one where a request message is sent and a response message is expected, possibly identified by its correlation identifier. These are represented in WSDL as an operation with an ***input*** element and an ***output*** and/or a ***fault*** element.

For an SCA reference with a JMS binding, when a request message is sent as part of a request/response MEP, the SCA runtime MUST set a non-null value for the ***JMSReplyTo*** header in the JMS message it creates for the request [BJM60003].

For an SCA reference with a JMS binding, when a request message is sent as part of a request/response MEP, and the JMS binding has a ***response*** element with a ***destination*** defined, then the SCA runtime MUST use that destination for the ***JMSReplyTo*** header in the JMS message it creates for the request [BJM60004].

For an SCA reference with a JMS binding, when a request message is sent as part of a request/response MEP, and the JMS binding does not have a ***response*** element with a ***destination*** defined, the SCA runtime MUST provide an appropriate destination on which to receive response messages and use that destination for the ***JMSReplyTo*** header in the JMS message it creates for the request [BJM60005].

For an SCA reference with a JMS binding, the SCA runtime MAY choose to receive response messages on the basis of their correlation ID as defined by the binding's ***@correlationScheme*** attribute, or use a unique destination for each response [BJM60006].

For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP where the request message included a non-null ***JMSReplyTo*** destination, the SCA runtime MUST send the response message to that destination [BJM60007].

For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP where the request message included a null ***JMSReplyTo*** destination and the JMS binding includes a ***response/destination*** element the SCA runtime MUST send the response message to that destination [BJM60008].

Deleted: ,

Deleted: and conversations

Deleted: When a request message is sent by a reference with a JMS binding for a one-way MEP, the SCA runtime SHOULD NOT set the ***JMSReplyTo*** destination header in the JMS message that it creates, regardless of whether the JMS binding has a ***response*** element with a ***destination*** defined. ¶ When a request message is received by a service with a JMS binding for a one-way MEP, the SCA runtime MUST ignore the ***JMSReplyTo*** destination header in the JMS message, and MUST NOT raise an error.¶

Deleted: 7.4

Deleted: 1

Deleted: 21st January

Deleted: 2008.

For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP where the request message included a null ***JMSReplyTo*** destination and the JMS binding does not include a ***response/destination*** then an error SHOULD be raised by the SCA runtime [BJM60009].

For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP the SCA runtime MUST set the correlation identifier in the JMS message that it creates for the response as defined by the JMS binding's ***@correlationScheme*** attribute [BJM60010].

The use of request/response exchanges when using a bidirectional interface is described in section 6.4.

6.3 JMS User Properties

This protocol assigns specific behavior to JMS user properties:

- "***scaCallbackDestination***" holds the name of the JMS Destination to which callback messages are sent.

6.4 Callbacks

Callbacks are SCA's way of representing bidirectional interfaces, where messages are sent in both directions between a client and a service. A callback is the invocation of an operation on a service's callback interface. A callback operation can be one-way or request/response. Messages that correspond to one-way or request/response operations on a bidirectional interface use either the ***scaCallbackDestination*** user property or the ***JMSReplyTo*** destination, or both, to identify the destination to which messages are to be sent when operations are invoked on the callback interface. The use of ***JMSReplyTo*** for this purpose is to enable interaction with non-SCA JMS applications, as described below.

SCA runtimes MUST follow the behavior described in section 6.4 and its subsections when ***binding.jms*** is used in both the forward and callback directions [BJM60018].

SCA runtimes can use different bindings for forward calls and callbacks, however the behavior and requirements on messages is vendor-specific.

6.4.1 Invocation of operations on a bidirectional interface

Error! Reference source not found. [BJM60011].

For an SCA reference with a JMS binding and bidirectional interface, when a request message is sent the SCA runtime MAY set the ***JMSReplyTo*** destination to the same value as the scaCallbackDestination user property [BJM60012].

For an SCA reference with a JMS binding and bidirectional interface, when a request message is sent as part of a request/response MEP, the SCA runtime MUST set the ***JMSReplyTo*** header in the message it creates as described in section 6.2 [BJM60013].

For both one-way and request/response operations, the reference's callback service can be used to identify the destination to which callback messages are to be sent.

For an SCA reference with a JMS binding and bidirectional interface, the SCA runtime MUST identify the callback destination from the reference's callback service binding if present, or supply a suitable callback destination if not present [BJM60014].

6.4.2 Invocation of operations on a callback interface

An SCA service with a callback interface can invoke operations on that callback interface by sending messages to the destination identified by the ***scaCallbackDestination*** user property in a message that it has received, the ***JMSReplyTo*** destination of a one-way message that it has received, or the destination identified by the service's callback reference JMS binding.

For an SCA service with a JMS binding, the *callback destination* is identified as follows, in order of priority:

- The ***scaCallbackDestination*** identified by an earlier request, if not null;

Deleted: When a request message is sent by a reference with a JMS binding for a request/response MEP, the SCA runtime MUST set a non-null value for the ***JMSReplyTo*** header in the JMS message it creates for the request. If the JMS binding has a ***response*** element with a ***destination*** defined, then the SCA runtime MUST use that destination for the ***JMSReplyTo*** header value, otherwise the SCA runtime MUST provide an appropriate destination on which to receive response messages. The SCA runtime MAY choose to receive the response message on the basis of its correlation ID as defined by the binding's ***@correlationScheme*** attribute, or use a unique destination for each response.¶
When a response message is sent by a service with a JMS binding for a request/res ... [69]

Deleted: 7.4

Deleted: ¶
<#>"***scaConversation*** ... [70]

Deleted: When a request message is sent by a re ... [71]

Formatted: Font color: Auto, English (U.S.)

Comment [SAJH4]: Is it OK to have a normative sta ... [72]

Deleted: 7.2

Deleted: if

Deleted: reference has a

Deleted: element with a JMS binding with a request ... [73]

Deleted: , otherwise the SCA runtime MUST provide a ... [74]

Deleted: When a callback request message is sent by a

Deleted: for either a one-way or request/response MEP

Deleted: SCA runtime MUST send the

Formatted: Font: Italic

Deleted: request message to the JMS

Formatted: Font: Italic

Formatted

Deleted:

Deleted: 1

Deleted: 21st January

Deleted: 2008.

- the ***JMSReplyTo*** destination identified by an earlier one-way request, if not null;
- the request destination of the service's callback reference JMS binding, if specified

For an SCA service with a JMS binding, when a callback request message is sent for either a one-way or request/response MEP, the SCA runtime MUST send the callback request message to the callback destination. [BJM60015].

For an SCA service with a JMS binding, when a callback request message is sent and no callback destination can be identified then the SCA runtime SHOULD raise an error, and MUST throw an exception to the caller of the callback operation [BJM60016].

For an SCA service with a JMS binding, when a callback request message is sent the SCA runtime MUST set the ***JMSReplyTo*** destination and correlation identifier in the callback request message as defined in sections 6.1 or 6.2 as appropriate for the type of the callback operation invoked [BJM60017].

6.4.3 Use of JMSReplyTo for callbacks for non-SCA JMS applications

When interacting with non-SCA JMS applications, the assembler can choose to model a request/response message exchange using a bidirectional interface. In this case it is likely that the non-SCA JMS application does not support the use of the ***scaCallbackDestination*** user property. To support this, for one-way messages the ***JMSReplyTo*** header can be used to identify the destination to be used to deliver callback messages, as described in sections 0 and 0.

Deleted: .

Deleted: If no destination is identified then the SCA runtime SHOULD raise an error, and MUST throw an exception to the caller of the callback operation. ¶
The SCA runtime MUST set the ***JMSReplyTo*** destination and correlation identifier in the callback request message as defined in sections 7.1 or 7.2 as appropriate for the type of the callback operation invoked. ¶

Deleted: 7.4.1

Deleted: 7.4.2

Deleted: **<#>Conversations** ¶
A conversation is a sequence of operations between two parties that have a common context. The conversation can include a mixture of operations in either direction between the two parties, if the interface is also bidirectional. Interfaces are marked as conversational in order to ensure that the runtime manages the lifecycle of this context. Component implementation specifications define the manner in which the context that is associated with the conversation identifier is made available to component implementations. ¶
<#>Starting a conversation ¶
A conversation is started when an operation is invoked on a conversational interface and there is no active conversation with the target of the invocation. When this happens the SCA runtime MUST supply an identifier for the conversation, if the client component has not already supplied an identifier, and the SCA runtime MUST set the ***scaConversationStart*** user property to this value in the JMS message that it sends for the request, and associate a new runtime context with this conversation identifier. ¶
When a message is received that contains a value for the ***scaConversationStart*** user property, the SCA runtime MUST associate a new runtime context with the given conversation identifier. ¶
The SCA runtime MAY include in the message that starts the conversation the ***scaConversationMaxI*** ... [75]

Deleted: 1

Deleted: 21st January

Deleted: 2008.

7 Examples

The following snippets show the ***sca.composite*** file for the ***MyValueComposite*** file containing the ***service*** element for the MyValueService and a ***reference*** element for the StockQuoteService. Both the service and the reference use a JMS binding.

7.1 Minimal Binding Example

The following example shows the JMS binding being used with no further attributes or elements. In this case, it is left to the deployer to identify the resources to which the binding is connected.

```
<?xml version="1.0" encoding="ASCII"?>
<composite xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200903"
           name="MyValueComposite">

    <service name="MyValueService">
        <interface.java interface="services.myvalue.MyValueService"/>
        <binding.jms/>
    </service>

    <reference name="StockQuoteService">
        <interface.java interface="services.stockquote.StockQuoteService"/>
        <binding.jms/>
    </reference>
</composite>
```

Deleted: 200712

7.2 URI Binding Example

The following example shows the JMS binding using the ***@uri*** attribute to specify the connection type and its information:

```
<?xml version="1.0" encoding="ASCII"?>
<composite xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200903"
           name="MyValueComposite">

    <service name="MyValueService">
        <interface.java interface="services.myvalue.MyValueService"/>
        <binding.jms uri="jms:MyValueServiceQueue?
                           activationSpecName=MyValueServiceAS&
                           ... "/>
    </service>

    <reference name="StockQuoteService">
        <interface.java interface="services.stockquote.StockQuoteService"/>
        <binding.jms uri="jms:StockQuoteServiceQueue?
                           connectionFactoryName=StockQuoteServiceQCF&
                           deliveryMode=1&
                           ... "/>
    </reference>
</composite>
```

Deleted: 200712

7.3 Binding with Existing Resources Example

The following example shows the JMS binding using existing resources:

```
<?xml version="1.0" encoding="ASCII"?>
<composite xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200903"
           name="MyValueComposite">

    <service name="MyValueService">
```

Deleted: 200712

Deleted: 1

Deleted: 21st January

Deleted: 2008.

```
        <interface.java interface="services.myvalue.MyValueService"/>
        <binding.jms>
            <destination jndiName="MyValueServiceQ" create="never"/>
            <activationSpec jndiName="MyValueServiceAS" create="never"/>
        </binding.jms>
    </service>
</composite>
```

Comment [SAJH5]: Is there a more realistic example of a JNDI name?

7.4 Resource Creation Example

The following example shows the JMS binding providing information to create JMS resources rather than using existing ones:

```
<?xml version="1.0" encoding="ASCII"?>
<composite xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200903"
           name="MyValueComposite">

    <service name="MyValueService">
        <interface.java interface="services.myvalue.MyValueService"/>
        <binding.jms>
            <destination jndiName="MyValueServiceQueue" create="always">
                <property name="prop1" type="string">XYZ</property>
                <property name="destName" type="string">MyValueDest</property>
            </destination>
            <activationSpec jndiName="MyValueServiceAS" create="always"/>
            <resourceAdapter jndiName="com.example.JMSRA"/>
        </binding.jms>
    </service>

    <reference name="StockQuoteService">
        <interface.java interface="services.stockquote.StockQuoteService"/>
        <binding.jms>
            <destination jndiName="StockQuoteServiceQueue"/>
            <connectionFactory jndiName="StockQuoteServiceQCF"/>
            <resourceAdapter name="com.example.JMSRA"/>
        </binding.jms>
    </reference>
</composite>
```

Deleted: 200712

7.5 Request/Response Example

The following example shows the JMS binding using existing resources to support request/response operations. The service uses the ***JMSReplyTo*** destination to send response messages, and does not specify a response queue:

```
<?xml version="1.0" encoding="ASCII"?>
<composite xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200903"
           name="MyValueComposite">

    <service name="MyValueService">
        <interface.java interface="services.myvalue.MyValueService"/>
        <binding.jms correlationScheme="sca:messageId">
            <destination jndiName="MyValueServiceQ" create="never"/>
            <activationSpec jndiName="MyValueServiceAS" create="never"/>
        </binding.jms>
    </service>

    <reference name="StockQuoteService">
        <interface.java interface="services.stockquote.StockQuoteService"/>
        <binding.jms correlationScheme="sca:messageId">
            <destination jndiName="StockQuoteServiceQueue"/>
            <connectionFactory jndiName="StockQuoteServiceQCF"/>
            <response>
```

Deleted: 200712

Deleted: MessageId

Deleted: MessageId

Deleted: 1

Deleted: 21st January

Deleted: 2008.

```
                <destination jndiName="MyValueResponseQueue"/>
                <activationSpec jndiName="MyValueResponseAS"/>
            </response>
        </binding.jms>
    </reference>
</composite>
```

7.6 Use of Predefined Definitions Example

This example shows the case where there is common connection information shared by more than one reference.

The common connection information is defined in a separate definitions file:

```
<?xml version="1.0" encoding="ASCII"?>
<definitions targetNamespace="http://acme.com"
             xmlns="http://docs.oasis-open.org/ns/opencsa/sca/2007903">
    <binding.jms name="StockQuoteService">
        <destination jndiName="StockQuoteServiceQueue" create="never"/>
        <connectionFactory jndiName="StockQuoteServiceQCF" create="never"/>
    </binding.jms>
</definitions>
```

Deleted: 200712

Any ***binding.jms*** element may then refer to that definition:

```
<?xml version="1.0" encoding="ASCII"?>
<composite xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200903"
           xmlns:acme="http://acme.com"
           name="MyValueComposite">
    <reference name="MyValueService">
        <interface.java interface="services.myvalue.MyValueService"/>
        <binding.jms requestConnection="acme:StockQuoteService"/>
    </reference>
</composite>
```

Deleted: 200712

7.7 Subscription with Selector Example

The following example shows how the JMS binding is used in order to consume messages from existing JMS infrastructure. The JMS binding subscribes using selector:

```
<?xml version="1.0" encoding="ASCII"?>
<composite xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200903"
           name="MyValueComposite">
    <service name="MyValueService">
        <interface.java interface="services.myvalue.MyValueService"/>
        <binding.jms>
            <destination jndiName="MyValueServiceTopic" create="never"/>
            <connectionFactory jndiName="StockQuoteServiceTCF"
create="never"/>
            <messageSelection selector="Price&gt;1000"/>
        </binding.jms>
    </service>
</composite>
```

Deleted: 200712

Deleted: subscriptionHeaders JMSSelector

7.8 Policy Set Example

A policy set defines the manner in which intents map to JMS binding properties. The following illustrates an example of a policy set that defines values for the ***@priority*** attribute using the "***priority***" intent, and also allows setting of a value for a user JMS property using the "***log***" intent.

Deleted: *JMS*

```
<policySet name="JMSPolicy"
           provides="priority log"
           appliesTo="binding.jms">
```

Deleted: 1

Deleted: 21st January

Deleted: 2008.

```
    <intentMap provides="priority" default="medium">
        <qualifier name="high">
            <headers priority="9"/>
        </qualifier>
        <qualifier name="medium">
            <headers priority="4"/>
        </qualifier>
        <qualifier name="low">
            <headers priority="0"/>
        </qualifier>
    </intentMap>

    <intentMap provides="log">
        <qualifier>
            <headers>
                <property name="user_example_log">logged</property>
            </headers>
        </qualifier>
    </intentMap>
</policySet>
```

Deleted: JMSPriority

Deleted: JMSPriority

Deleted: JMSPriority

Given this policy set, the intents can be required on a service or reference:

```
<reference name="StockQuoteService" requires="priority.high log">
    <interface.java interface="services.stockquote.StockQuoteService"/>
    <binding.jms>
        <destination name="StockQuoteServiceQueue"/>
        <connectionFactory name="StockQuoteServiceQCF"/>
    </binding.jms>
</reference>
```

Deleted: 1

Deleted: 21st January

Deleted: 2008.

8 Conformance

The XML schema pointed to by the RDDL document at the namespace URI, defined by this specification, are considered to be authoritative and take precedence over the XML schema defined in the appendix of this document. There are two categories of artifacts for which this specification defines conformance:

a) SCA JMS Binding XML Document

b) SCA Runtime

8.1 SCA JMS Binding XML Document

An SCA JMS Binding XML document is an SCA Composite Document, an SCA Definitions Document or an SCA ComponentType Document, as defined by the SCA Assembly Specification Section 13.1 , that uses the ***binding.jms*** element.

An SCA JMS Binding XML document MUST be a conformant SCA Composite Document, SCA Definitions Document or a SCA ComponentType Document, as defined by the SCA Assembly Specification , and MUST comply with all the applicable requirements specified in this specification.

8.2 SCA Runtime

An implementation that claims to conform to the requirements of an SCA Runtime defined in this specification has to meet the following conditions:

1. The implementation MUST comply with all statements in Appendix B: Conformance Items related to an SCA Runtime, notably all "MUST" statements have to be implemented
2. The implementation MUST conform to the SCA Assembly Model Specification Version 1.1 , and to the SCA Policy Framework Version 1.1 [SCA-Policy]
3. The implementation MUST reject an SCA JMS Binding XML Document that is not conformant per Section 8.1

Deleted: Any SCA runtime that claims to support this binding MUST abide by the requirements of this specification.¶

Deleted: available

Deleted: is

Deleted: takes

Deleted: Schema

Deleted: 1

Deleted: 21st January

Deleted: 2008.

A. JMS XML Binding Schema: sca-binding-jms.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Copyright (C) OASIS(R) 2005,2009. All Rights Reserved.
     OASIS trademark, IPR and other policies apply. -->
<schema xmlns="http://www.w3.org/2001/XMLSchema"
        targetNamespace="http://docs.oasis-open.org/ns/opencsa/sca/200903"
        xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200903"
        elementFormDefault="qualified">

   <include schemaLocation="sca-core-1.1-cd03.xsd"/>

   <complexType name="JMSBinding">
      <complexContent>
         <extension base="sca:Binding">
            <sequence>
               <element name="destination" type="sca:JMSDestination"
                        minOccurs="0"/>
               <choice minOccurs="0" maxOccurs="1">
                  <element name="connectionFactory"
                           type="sca:JMSConnectionFactory"/>
                  <element name="activationSpec" type="sca:JMSActivationSpec"/>
               </choice>
               <element name="response" type="sca:JMSResponse" minOccurs="0"/>
               <element name="headers" type="sca:JMSHeaders" minOccurs="0"/>
               <element name="messageSelection" type="sca:JMSMessageSelection"
                        minOccurs="0"/>
               <element name="resourceAdapter" type="sca:JMSResourceAdapter"
                        minOccurs="0"/>
               <element name="operationProperties"
                        type="sca:JMSOperationProperties"
                        minOccurs="0" maxOccurs="unbounded"/>
               <any namespace="##other" processContents="lax"
                    minOccurs="0" maxOccurs="unbounded"/>
            </sequence>
            <attribute name="correlationScheme" type="QName"
                       default="sca:messageId"/>
            <attribute name="initialContextFactory" type="anyURI"/>
            <attribute name="jndiURL" type="anyURI"/>
            <attribute name="requestConnection" type="QName"/>
            <attribute name="responseConnection" type="QName"/>
            <attribute name="operationProperties" type="QName"/>
            <anyAttribute/>
         </extension>
      </complexContent>
   </complexType>

   <simpleType name="JMSCreateResource">
      <restriction base="string">
         <enumeration value="always"/>
         <enumeration value="never"/>
         <enumeration value="ifNotExist"/>
      </restriction>
   </simpleType>

   <complexType name="JMSDestination">
      <sequence>
         <element name="property" type="sca:BindingProperty"
                  minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="jndiName" type="anyURI" use="required"/>
```

Deleted: Within this specification, the following conformance targets are used:¶ <#>XML document elements and attributes, including ***binding.jms*** and its children, and ***bindingType***¶ <#>The SCA runtime – this refers to the implementation that provides the functionality to support the SCA specifications, including that specific to the JMS binding as well as other SCA capabilities¶ <#>JMS objects, including Destinations, ConnectionFactories and ActivationSpecs¶ <#>WSDL documents¶ **<#>JMS Binding Schema¶**

Deleted: (c)

Deleted: 2006, 2008

Deleted: 200712"

Deleted: 200712

Deleted: <sequence> ¶ <element name="destination" type="sca:JMSDestination"/> ¶

Deleted:

Deleted: </sequence> ¶ <sequence> ¶ <element name="destination" ¶ type="sca:JMSDestination" minOccurs="0"/> ¶

Deleted: ¶ </sequence>

Deleted: subscriptionHeaders "

Deleted: type="sca:JMSSubscriptionHeaders"¶

Deleted: ="

Deleted: "

Deleted: MessageId

Deleted: ifnotexist

Deleted: 1

Deleted: 21st January

Deleted: 2008.

```
      <attribute name="type" use="optional" default="queue">
         <simpleType>
            <restriction base="string">
               <enumeration value="queue"/>
               <enumeration value="topic"/>
            </restriction>
         </simpleType>
      </attribute>
      <attribute name="create" type="sca:JMSCreateResource"
                 use="optional" default="ifNotExist"/>
   </complexType>

   <complexType name="JMSConnectionFactory">
      <sequence>
         <element name="property" type="sca:BindingProperty"
                  minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="jndiName" type="anyURI" use="required"/>
      <attribute name="create" type="sca:JMSCreateResource"
                 use="optional" default="ifNotExist"/>
   </complexType>

   <complexType name="JMSActivationSpec">
      <sequence>
         <element name="property" type="sca:BindingProperty"
                  minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="jndiName" type="anyURI" use="required"/>
      <attribute name="create" type="sca:JMSCreateResource"
                 use="optional" default="ifNotExist"/>
   </complexType>

   <complexType name="JMSResponse">
      <sequence>
         <element name="wireFormat" type="sca:WireFormatType" minOccurs="0"/>
         <element name="destination" type="sca:JMSDestination" minOccurs="0"/>
         <choice minOccurs="0">
            <element name="connectionFactory" type="sca:JMSConnectionFactory"/>
            <element name="activationSpec" type="sca:JMSActivationSpec"/>
         </choice>
      </sequence>
   </complexType>

   <complexType name="JMSHeaders">
      <sequence>
         <element name="property" type="sca:BindingProperty"
                  minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="type" type="string"/>
      <attribute name="deliveryMode">
         <simpleType>
            <restriction base="string">
               <enumeration value="persistent"/>
               <enumeration value="nonpersistent"/>
            </restriction>
         </simpleType>
      </attribute>
      <attribute name="timeToLive" type="long"/>
      <attribute name="priority">
         <simpleType>
            <restriction base="string">
               <enumeration value="0"/>
               <enumeration value="1"/>
               <enumeration value="2"/>
```

Deleted: ifnotexist

Deleted: ifnotexist

Deleted: ifnotexist

Deleted: JMSType

Deleted: JMSDeliveryMode

Deleted: PERSISTENT

Deleted: NON_PERSISTENT

Deleted: JMSTimeToLive

Deleted: JMSPriority

Deleted: 1

Deleted: 21st January

Deleted: 2008.

```
                <enumeration value="3"/>
                <enumeration value="4"/>
                <enumeration value="5"/>
                <enumeration value="6"/>
                <enumeration value="7"/>
                <enumeration value="8"/>
                <enumeration value="9"/>
             </restriction>
          </simpleType>
       </attribute>
    </complexType>

    <complexType name="JMSMessageSelection">
       <sequence>
          <element name="property" type="sca:BindingProperty"
                   minOccurs="0" maxOccurs="unbounded"/>
       </sequence>
       <attribute name="selector" type="string"/>
    </complexType>

    <complexType name="JMSResourceAdapter">
       <sequence>
          <element name="property" type="sca:BindingProperty"
                   minOccurs="0" maxOccurs="unbounded"/>
       </sequence>
       <attribute name="name" type="string" use="required"/>
    </complexType>

    <complexType name="JMSOperationProperties">
       <sequence>
          <element name="property" type="sca:BindingProperty"
                   minOccurs="0" maxOccurs="unbounded"/>
          <element name="headers" type="sca:JMSHeaders"/>
       </sequence>
       <attribute name="name" type="string" use="required"/>
       <attribute name="nativeOperation" type="string"/>
    </complexType>

    <complexType name="BindingProperty">
       <simpleContent>
          <extension base="string">
             <attribute name="name" type="NMTOKEN"/>
             <attribute name="type" type="string" use="optional"
                        default="xs:string"/>
          </extension>
       </simpleContent>
    </complexType>

    <element name="binding.jms" type="sca:JMSBinding"
             substitutionGroup="sca:binding"/>

    <element name="wireFormat.jmsDefault" type="sca:WireFormatType"
             substitutionGroup="sca:wireFormat"/>

    <element name="operationSelector.jmsDefault" type="sca:OperationSelectorType"
             substitutionGroup="sca:operationSelector"/>
</schema>
```

Deleted: JMSSubscriptionHeaders

Deleted: JMSSelector

Deleted: jmsdefault

Deleted: jmsdefault

Deleted: 1

Deleted: 21st January

Deleted: 2008.

B. Conformance Items

This section contains a list of conformance items for the SCA JMS Binding specification.

Conformance ID	Description
[BJM30001]	The value of the ***@uri*** attribute MUST have the format defined by the IETF URI Scheme for Java™ Message Service 1.0 [IETFJMS]
[BJM30002]	When the ***@uri*** attribute is specified, the SCA runtime MUST raise an error if the referenced resources do not already exist
[BJM30003]	If the value of the ***@correlationScheme*** attribute is "***sca:messageID***" the SCA runtime MUST set the correlation ID of replies to the message ID of the corresponding request
[BJM30004]	If the value of the ***@correlationScheme*** attribute is "***sca:correlationID***" the SCA runtime MUST set the correlation ID of replies to the correlation ID of the corresponding request
[BJM30005]	If the value of the ***@correlationScheme*** attribute is "***sca:none***" the SCA runtime MUST NOT set the correlation ID
[BJM30006]	SCA runtimes MAY allow other values of the ***@correlationScheme*** attribute to indicate other correlation schemes
[BJM30007]	If the ***@requestConnection*** attribute is specified, the ***binding.jms*** element MUST NOT contain a ***destination***, ***connectionFactory***, ***activationSpec*** or ***resourceAdapter*** element
[BJM30008]	If the ***@responseConnection*** attribute is specified, the ***binding.jms*** element MUST NOT contain a ***response*** element
[BJM30009]	If the ***@operationProperties*** attribute is specified, the ***binding.jms*** element MUST NOT contain an ***operationProperties*** element
[BJM30010]	Whatever the value of the ***destination/@type*** attribute, the runtime MUST ensure a single response is delivered for request/response operations
[BJM30011]	If the ***@create*** attribute value for a destination, connectionFactory or activationSpec element is "***always***" then the ***@jndiName*** attribute is optional; if the resource cannot be created at the specified location then the SCA runtime MUST raise an error
[BJM30012]	If the ***@create*** attribute value for a destination, connectionFactory or activationSpec element is "***ifNotExist***" then the ***@jndiName*** attribute MUST specify the location of the possibly existing resource
[BJM30013]	If the destination, connectionFactory or activationSpec does not exist at the location identified by the ***@jndiName*** attribute, but cannot be created there then the SCA runtime MUST raise an error
[BJM30014]	If the destination, connectionFactory or activationSpec's ***@jndiName*** attribute refers to an existing resource that is not a JMS Destination of the approprate type, a JMS connection factory or a JMS activation spec respectively then the SCA runtime MUST raise an error
[BJM30015]	If the ***@create*** attribute value for a destination, connectionFactory or

Deleted: 1
Deleted: 21st January
Deleted: 2008.

	activationSpec element is "***never***" then the ***@jndiName*** attribute MUST specify the location of the existing resource
[BJM30016]	If the destination, connection factory or activation spec is not present at the location identified by the ***@jndiName*** attribute, or the location refers to a resource of an incorrect type then the SCA runtime MUST raise an error
[BJM30017]	A ***binding.jms*** element MUST NOT include both a ***connectionFactory*** element and an ***activationSpec*** element
[BJM30018]	When the ***connectionFactory*** element is present, then the destination MUST be defined either by the ***destination*** element or the ***@uri*** attribute
[BJM30019]	If the ***activationSpec*** element is present and the destination is also specified via a ***destination*** element or the ***@uri*** attribute then it MUST refer to the same JMS destination as the ***activationSpec***
[BJM30020]	The ***activationSpec*** element MUST NOT be present when the binding is being used for an SCA reference
[BJM30021]	A ***response*** element MUST NOT include both a ***connectionFactory*** element and an ***activationSpec*** element
[BJM30022]	If a ***response/destination*** and ***response/activationSpec*** element are both specified they MUST refer to the same JMS destination
[BJM30023]	The ***response/activationSpec*** element MUST NOT be present when the binding is being used for an SCA service
[BJM30024]	The SCA runtime MUST set JMS headers in messages that it creates to the values specified by the ***headers*** element unless overridden for the operation being invoked.
[BJM30025]	If the ***@uri*** attribute includes values for the type, delivery mode, time to live or priority properties then the @uri values are used and the ***headers*** and ***operationProperties/headers @type***, ***@deliveryMode***, ***@timeToLive*** or ***@priority*** attributes are ignored
[BJM30026]	For each ***header/properties*** element the SCA runtime MUST set the named JMS user property to the given value in messages it creates unless overridden for the operation being invoked
[BJM30027]	If the ***@uri*** attribute includes a value for the message selector then the ***@uri*** value is used and the ***messageSelection/@selector*** attribute is ignored
[BJM30028]	SCA runtimes MAY place restrictions on the properties of the resource adapter Java bean that can be set using the ***resourceAdapter*** element
[BJM30029]	The value of the ***operationProperties/@selectedOperation*** attribute MUST be unique across the containing ***binding.jms*** element
[BJM30030]	The SCA runtime SHOULD make the ***operationProperties*** element corresponding to the ***selectedOperation*** available to the ***wireFormat*** implementation
[BJM30031]	The ***resourceAdapter*** element MUST be present when JMS resources are to be created for a JMS provider that implements the JCA 1.5 specification [JCA15], and is ignored otherwise
[BJM30032]	The SCA runtime MUST set JMS headers in messages it creates when the operation identified by the ***operationProperties/@name*** attribute is invoked to

Deleted: 1

Deleted: 21st January

Deleted: 2008.

	the values specified by the corresponding ***operationProperties/headers*** element
[BJM30033]	For each ***operationProperties/headers/property*** element the SCA runtime MUST set the named JMS user property to the given value in messages it creates when the operation identified by the ***operationProperties/@name*** attribute is invoked
[BJM30034]	When the ***@uri*** attribute is specified, the ***destination*** element MUST NOT be present
[BJM30035]	An SCA runtime MUST use the values specified in the ***@uri*** attribute in preference to corresponding attributes and elements in the binding
[BJM30036]	The ***binding.jms*** element MUST conform to the XML schema defined in sca-binding-jms.xsd
[BJM40001]	The SCA runtime MUST support the default JMS wire format and operation selector behavior, and MAY provide additional means to override it
[BJM40002]	If no ***operationSelector*** element is specified then SCA runtimes MUST use ***operationSelector.jmsDefault*** as the default
[BJM40003]	When using the default wire format to send request messages, if there is a single parameter and the interface includes more than one operation, the SCA runtime MUST set the JMS user property "***scaOperationName***" to the name of the operation being invoked
[BJM40004]	If no ***wireFormat*** element is specified in a JMS binding then SCA runtimes MUST use ***wireFormat.jmsDefault*** as the default
[BJM40005]	When using the default wire format an SCA runtime MUST be able to receive both JMS text and bytes messages
[BJM40006]	When using the default wire format an SCA runtime MUST send either a JMS text or a JMS bytes message
[BJM40007]	When using the default wire format an SCA runtime MAY provide additional configuration to allow selection between JMS text or bytes messages to be sent
[BJM40008]	When a ***binding.jms*** element specifies the ***operationSelector.jmsDefault*** element, the SCA runtime MUST use the default operation selection algorithm to determine the selected operation
[BJM40009]	When a ***binding.jms*** element specifies the ***wireFormat.jmsDefault*** element, the SCA runtime MUST use the default wire format
[BJM60001]	For an SCA reference with a JMS binding, when a request message is sent as part of a one-way MEP, the SCA runtime SHOULD NOT set the ***JMSReplyTo*** destination header in the JMS message that it creates, regardless of whether the JMS binding has a ***response*** element with a ***destination*** defined
[BJM60002]	For an SCA service with a JMS binding, when a request message is received as part of a one-way MEP, the SCA runtime MUST ignore the ***JMSReplyTo*** destination header in the JMS message, and not raise an error
[BJM60003]	For an SCA reference with a JMS binding, when a request message is sent as part of a request/response MEP, the SCA runtime MUST set a non-null value for the ***JMSReplyTo*** header in the JMS message it creates for the request

Deleted: 1

Deleted: 21st January

Deleted: 2008.

[BJM60004]	For an SCA reference with a JMS binding, when a request message is sent as part of a request/response MEP, and the JMS binding has a ***response*** element with a ***destination*** defined, then the SCA runtime MUST use that destination for the ***JMSReplyTo*** header in the JMS message it creates for the request
[BJM60005]	For an SCA reference with a JMS binding, when a request message is sent as part of a request/response MEP, and the JMS binding does not have a ***response*** element with a ***destination*** defined, the SCA runtime MUST provide an appropriate destination on which to receive response messages and use that destination for the ***JMSReplyTo*** header in the JMS message it creates for the request
[BJM60006]	For an SCA reference with a JMS binding, the SCA runtime MAY choose to receive response messages on the basis of their correlation ID as defined by the binding's ***@correlationScheme*** attribute, or use a unique destination for each response
[BJM60007]	For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP where the request message included a non-null ***JMSReplyTo*** destination, the SCA runtime MUST send the response message to that destination
[BJM60008]	For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP where the request message included a null ***JMSReplyTo*** destination and the JMS binding includes a ***response/destination*** element the SCA runtime MUST send the response message to that destination
[BJM60009]	For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP where the request message included a null ***JMSReplyTo*** destination and the JMS binding does not include a ***response/destination*** then an error SHOULD be raised by the SCA runtime
[BJM60010]	For an SCA service with a JMS binding, when a response message is sent as part of a request/response MEP the SCA runtime MUST set the correlation identifier in the JMS message that it creates for the response as defined by the JMS binding's ***@correlationScheme*** attribute
[BJM60011]	For an SCA reference with a JMS binding and a bidirectional interface, when a request message is sent the SCA runtime MUST set the destination to which callback messages are to be sent as the value of the ***scaCallbackDestination*** user property in the message it creates
[BJM60012]	For an SCA reference with a JMS binding and bidirectional interface, when a request message is sent the SCA runtime MAY set the ***JMSReplyTo*** destination to the same value as the scaCallbackDestination user property
[BJM60013]	For an SCA reference with a JMS binding and bidirectional interface, when a request message is sent as part of a request/response MEP, the SCA runtime MUST set the ***JMSReplyTo*** header in the message it creates as described in section 6.2
[BJM60014]	For an SCA reference with a JMS binding and bidirectional interface, the SCA runtime MUST identify the callback destination from the reference's callback service binding if present, or supply a suitable callback destination if not present
[BJM60015]	For an SCA service with a JMS binding, when a callback request message is

Deleted: 1

Deleted: 21st January

Deleted: 2008.

	sent for either a one-way or request/response MEP, the SCA runtime MUST send the callback request message to the callback destination.
[BJM60016]	For an SCA service with a JMS binding, when a callback request message is sent and no callback destination can be identified then the SCA runtime SHOULD raise an error, and MUST throw an exception to the caller of the callback operation
[BJM60017]	For an SCA service with a JMS binding, when a callback request message is sent the SCA runtime MUST set the ***JMSReplyTo*** destination and correlation identifier in the callback request message as defined in sections 6.1 or 6.2 as appropriate for the type of the callback operation invoked
[BJM60018]	SCA runtimes MUST follow the behavior described in section 6.4 and its subsections when ***binding.jms*** is used in both the forward and callback directions

938

Deleted: 1
Deleted: 21st January
Deleted: 2008.

C. Acknowledgements

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Participant Name	Affiliation
Bryan Aupperle	IBM
Ron Barack	SAP AG
Michael Beisiegel	IBM
Henning Blohm	SAP AG
David Booz	IBM
Martin Chapman	Oracle Corporation
Jean-Sebastien Delfino	IBM
Laurent Domenech	TIBCO Software Inc.
Jacques Durand	Fujitsu Limited
Mike Edwards	IBM
Billy Feng	Primeton Technologies, Inc.
Nimish Hathalia	TIBCO Software Inc.
Simon Holdsworth	IBM
Eric Johnson	Software Inc.
Uday Joshi	Oracle Corporation
Khanderao Kand	Oracle Corporation
Anish Karmarkar	Oracle Corporation
Nickolaos Kavantzas	Oracle Corporation
Mark Little	Red Hat
Ashok Malhotra	Oracle Corporation
Jim Marino	Individual
Jeff Mischkinsky	Oracle Corporation
Dale Moberg	Axway Software
Simon Nash	Individual
Sanjay Patil	SAP AG
Plamen Pavlov	SAP AG
Peter Peshev	SAP AG
Piotr Przybylski	IBM
Luciano Resende	IBM
Tom Rutt	Fujitsu Limited
Vladimir Savchenko	SAP AG
Scott Vorthmann	TIBCO Software Inc.
Tim Watson	Oracle Corporation
Owen Williams	Avaya, Inc.
Prasad Yendluri	Software AG, Inc.

Deleted: 1

Deleted: 21st January

Deleted: 2008.

943

Deleted: DisplayText canr
DisplayText cannot span
¶
<#>Non-Normative Text¶

Deleted: 1
Deleted: 21st January
Deleted: 2008.

944

Deleted: 1
Deleted: 21st January
Deleted: 2008.

D. Revision History

[optional; should not be included in OASIS Standards]

Revision	Date	Editor	Changes Made
1	2007-09-25	Anish Karmarkar	Applied the OASIS template + related changes to the Submission
2	2008-03-12	Simon Holdsworth	Updated text for RFC2119 conformance Updates to resolve following issues: BINDINGS-1 BINDINGS-5 BINDINGS-6 BINDINGS-12 BINDINGS-14 BINDINGS-18 BINDINGS-26 Applied updates discussed at Bindings TC meeting of 27th March
3	2008-06-19	Simon Holdsworth	* Applied most of the editorial changes from Eric Johnson's review
cd01	2008-08-01	Simon Holdsworth	Updates to resolve following issues: BINDINGS-13 (JMS part) BINDINGS-20 (complete) BINDINGS-30 (JMS part) BINDINGS-32 (JMS part) BINDINGS-33 (complete) BINDINGS-34 (complete) BINDINGS-35 (complete) BINDINGS-38 (JMS part)
cd01-rev1	2008-10-16	Simon Holdsworth	Updated text for RFC2119 conformance throughout Updates to resolve following issues: BINDINGS-41 BINDINGS-46 BINDINGS-47
cd01-rev2	2008-12-01	Simon Holdsworth	Added comments identifying those updates that relate to RFC2119 language (issue 52)
cd01-rev3	2008-12-02	Simon Holdsworth	Final RFC2119 language updates BINDINGS-52
cd01-rev4	2009-01-09	Simon Holdsworth	Updates to resolve following issues:

Deleted: 1
Deleted: 21st January
Deleted: 2008.

			BINDINGS-7 BINDINGS-31 BINDINGS-40 BINDINGS-42 BINDINGS-44 BINDINGS-50
cd02	2009-02-16	Simon Holdsworth	Rename and editorial updates
cd02-rev1	2009-05-22	Simon Holdsworth	Updates to resolve issue BINDINGS-62 (conformance statement numbering) Updated assembly namespace to 200903 Fixed errors in schema
cd02-rev2	2009-05-22	Simon Holdsworth	Updates to resolve following issues: BINDINGS-39 BINDINGS-59 BINDINGS-65 BINDINGS-66 BINDINGS-67 BINDINGS-68 BINDINGS-70 BINDINGS-71
cd02-rev3	2009-06-18	Simon Holdsworth	Editorial concerns addressed Added acknowledgements appendix
cd02-rev4	2009-06-19	Simon Holdsworth	Updates to resolve following issues BINDINGS-74 Some editorial updates Fixed normative statement missed in application of BINDINGS-67

948

Deleted: 1
Deleted: 21st January
Deleted: 2008.

Page 4: [1] Change **Unknown**

Field Code Changed

Page 4: [1] Change **Unknown**

Field Code Changed

Page 4: [2] Change **Unknown**

Field Code Changed

Page 4: [2] Change **Unknown**

Field Code Changed

Page 4: [3] Change **Unknown**

Field Code Changed

Page 4: [3] Change **Unknown**

Field Code Changed

Page 4: [4] Change **Unknown**

Field Code Changed

Page 4: [4] Change **Unknown**

Field Code Changed

Page 4: [5] Change **Unknown**

Field Code Changed

Page 4: [6] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [7] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 2, Tabs: Not at 0.85 cm

Page 4: [8] Change **Unknown**

Field Code Changed

Page 4: [9] Change **Unknown**

Field Code Changed

Page 4: [10] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [11] Change **Unknown**

Field Code Changed

Page 4: [12] Change **Unknown**

Field Code Changed

Page 4: [13] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [14] Change **Unknown**

Field Code Changed

Page 4: [15] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 1,TOC 1 Char,TOC 1 Char1 Char,TOC 1 Char Char Char, Tabs: 0.85 cm, Left

Page 4: [16] Change **Unknown**

Field Code Changed

Page 4: [17] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [18] Change **Unknown**

Field Code Changed

Page 4: [19] Change **Unknown**

Field Code Changed

Page 4: [20] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [20] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [21] Change **Unknown**

Field Code Changed

Page 4: [22] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 2, Tabs: Not at 0.85 cm

Page 4: [23] Change **Unknown**

Field Code Changed

Page 4: [24] Change **Unknown**

Field Code Changed

Page 4: [25] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 3, Tabs: Not at 0.85 cm

Page 4: [26] Change **Unknown**

Field Code Changed

Page 4: [27] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [28] Change **Unknown**

Field Code Changed

Page 4: [29] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 1,TOC 1 Char,TOC 1 Char1 Char,TOC 1 Char Char Char, Tabs: 0.85 cm, Left

Page 4: [30] Change **Unknown**

Field Code Changed

Page 4: [31] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [32] Change **Unknown**

Field Code Changed

Page 4: [33] Change **Unknown**

Field Code Changed

Page 4: [34] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [35] Change **Unknown**

Field Code Changed

Page 4: [36] Change **Unknown**

Field Code Changed

Page 4: [37] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [38] Change **Unknown**

Field Code Changed

Page 4: [39] Change **Unknown**

Field Code Changed

Page 4: [40] Change **Unknown**

Field Code Changed

Page 4: [41] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 2

Page 4: [42] Change **Unknown**

Field Code Changed

Page 4: [43] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [44] Change **Unknown**

Field Code Changed

Page 4: [45] Change **Unknown**

Field Code Changed

Page 4: [46] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [47] Change **Unknown**

Field Code Changed

Page 4: [48] Change **Unknown**

Field Code Changed

Page 4: [49] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [50] Change **Unknown**

Field Code Changed

Page 4: [51] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 3

Page 4: [52] Change **Unknown**

Field Code Changed

Page 4: [53] Change **Unknown**

Field Code Changed

Page 4: [54] Change **Unknown**

Field Code Changed

Page 4: [55] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [56] Change **Unknown**

Field Code Changed

Page 4: [57] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Page 4: [58] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 1,TOC 1 Char,TOC 1 Char1 Char,TOC 1 Char Char Char, Tabs: 0.85 cm, Left

Page 4: [59] Change **Unknown**

Field Code Changed

Page 4: [60] Change **Unknown**

Field Code Changed

Page 4: [61] Formatted **Simon Holdsworth** **19/06/2009 15:18:00**

TOC 2

Page 1: [62] Deleted **Simon Holdsworth** **16/02/2009 10:52:00**

1

Page 1: [62] Deleted **Simon Holdsworth** **16/02/2009 10:53:00**

21st January

Page 9: [63] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

. Possible values for the ***@correlationScheme*** attribute are "***sca:MessageID***" (the default) where the SCA runtime MUST set the correlation ID of replies to the message ID of the corresponding request; "***sca:CorrelationID***" where the SCA runtime MUST set the correlation ID of replies to the correlation ID of the corresponding request, and "***sca:None***" which indicates that the SCA runtime MUST NOT set the correlation ID. SCA runtimes MAY allow other values to indicate other correlation schemes

Page 11: [64] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

that the SCA runtime MUST set to the given values for all operations.

Page 11: [65] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

JMSType, @JMSDeliveryMode, @JMSTimeToLive, @JMSPriority

Page 11: [66] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

. The value of the ***@uri*** attribute MUST NOT include values for these properties if they are specified using these attributes.

Page 11: [67] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

JMSDeliveryMode are "***PERSISTENT***" and "***NON_PERSISTENT***

Page 11: [68] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

The value of the ***@uri*** attribute MUST NOT include values for this property if it is specified using this attribute

Page 18: [69] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

When a request message is sent by a reference with a JMS binding for a request/response MEP, the SCA runtime MUST set a non-null value for the ***JMSReplyTo*** header in the JMS message it creates for the request. If the JMS binding has a ***response*** element with a ***destination*** defined, then the SCA runtime MUST use that destination for the ***JMSReplyTo*** header value, otherwise the SCA runtime MUST provide an appropriate destination on which to receive response messages. The SCA runtime MAY choose to receive the response message on the basis of its correlation ID as defined by the binding's ***@correlationScheme*** attribute, or use a unique destination for each response.

When a response message is sent by a service with a JMS binding for a request/response MEP, the SCA runtime MUST send the response message to the destination identified by the request message's ***JMSReplyTo*** header value if it is not null, otherwise the SCA runtime MUST send the response message to the destination identified by the JMS binding's ***response*** element if specified. If there is no destination defined by either means then an error SHOULD be raised by the SCA runtime. The SCA runtime MUST set the correlation identifier in the JMS message that it creates for the response as defined by the JMS binding's ***@correlationScheme*** attribute.

Page 18: [70] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

"***scaConversationStart***" indicates that a conversation is to be started, its value is the identifier for the conversation.

"***scaConversationMaxIdleTime***" defines the maximum time that should be allowed between operations in the conversation.

"***scaConversationId***" holds the identifier for the conversation.

Page 18: [71] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

When a request message is sent by a reference with a JMS binding for a one-way MEP with a bidirectional interface, the SCA runtime MUST set the destination to which callback messages are to be sent as the value of the ***scaCallbackDestination*** user property in the message it creates. The SCA runtime MAY also set the ***JMSReplyTo*** destination to this value.

When a request message is sent by a reference with a JMS binding for a request/response MEP with a bidirectional interface, the SCA runtime MUST set the ***scaCallbackDestination*** user property in the message it creates to identify the destination from which it will read callback messages. The SCA runtime MUST set the ***JMSReplyTo*** header in the message it creates as described in section

Page 18: [72] Comment [SAJH4] **Simon Holdsworth** **20/04/2009 14:54:00**

Is it OK to have a normative statement refer to a section in the doc?

Page 18: [73] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

element with a JMS binding with a request destination, then the SCA runtime MUST use that destination as the one

Page 18: [74] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

, otherwise the SCA runtime MUST provide an appropriate destination for this purpose.

Page 19: [75] Deleted **Simon Holdsworth** **19/06/2009 15:18:00**

Conversations

A conversation is a sequence of operations between two parties that have a common context. The conversation can include a mixture of operations in either direction between the two parties, if the interface is also bidirectional. Interfaces are marked as conversational in order to ensure that the runtime manages the lifecycle of this context. Component implementation specifications define the manner in which the context that is associated with the conversation identifier is made available to component implementations.

Starting a conversation

A conversation is started when an operation is invoked on a conversational interface and there is no active conversation with the target of the invocation. When this happens the SCA runtime MUST supply an identifier for the conversation, if the client component has not already supplied an identifier, and the SCA runtime MUST set the ***scaConversationStart*** user property to this value in the JMS message that it sends for the request, and associate a new runtime context with this conversation identifier.

When a message is received that contains a value for the ***scaConversationStart*** user property, the SCA runtime MUST associate a new runtime context with the given conversation identifier.

The SCA runtime MAY include in the message that starts the conversation the ***scaConversationMaxIdleTime*** user property; if this value is not present the SCA runtime MUST derive the maximum idle time for the conversation by subtracting the current time from the value of the ***JMSExpiration*** property, unless the ***JMSExpiration*** property value is zero, in which case the maximum idle time is unlimited.

The SCA runtime MUST consider operations invoked on or by other parties to be outside of a conversation with a given party, and MUST use different conversation identifiers if those operations are conversational.

Continuing a conversation

When creating messages for subsequent operations between the sender and receiver that are part of this conversation, the SCA runtime MUST include the ***scaConversationId*** user property in the JMS message, set to the conversation identifier. The SCA runtime MAY also include an updated value of the ***scaConversationMaxIdleTime*** property. Once a conversation has been started, the SCA runtime MUST use the initial value of the ***scaCallbackDestination*** user property for all messages in the conversation, and MUST ignore the value of the ***scaCallbackDestination*** user property in subsequent messages in the same conversation.

The SCA runtime MUST deal with messages received either containing a conversation identifier that does not correspond to a started conversation, or containing the ***scaConversationStart*** user property with a conversation identifier that matches an active conversation, by raising an error, and MUST NOT deliver such messages.

Ending a conversation

When an operation is invoked by either party that is marked as "***endsConversation***", or the maximum idle time is exceeded, then the SCA runtime MUST discard the conversation identifier and associated context after the operation has been processed. The idle time is defined as the amount of time since the SCA runtime last completed processing of an operation that is part of the conversation. There may be times when one party ends the conversation before the other does. In that case if one party does invoke an operation on the other, the SCA runtime MUST NOT deliver the message and SHOULD raise an error.

The SCA runtime MAY reuse conversation identifiers. In particular, the SCA runtime does not have to guarantee unique conversation identifiers and does not have to be able to identify an ended conversation indefinitely, although it MAY do so for some period after the conversation ends. Due to the long-running nature of conversations, the SCA runtime SHOULD ensure conversation context is available across server restarts, although it MAY choose to treat a server restart as implicitly ending the conversation.