

Style Definition: Attribute:
Font: Bold, Italic



Service Component Architecture JMS Binding Specification Version 1.1

Working Draft 04

1 August 2008

Deleted: 2

Deleted: 28

Deleted: March

Specification URIs:

This Version:

<http://docs.oasis-open.org/opencsa/sca-bindings/sca-jmsbinding-1.1-spec-WD-04.doc>
<http://docs.oasis-open.org/opencsa/sca-bindings/sca-jmsbinding-1.1-spec-WD-04.pdf>
(Authoritative)

Deleted: 2

Deleted: 2

Previous Version:

<http://docs.oasis-open.org/opencsa/sca-bindings/sca-jmsbinding-1.1-spec-WD-03.doc>
<http://docs.oasis-open.org/opencsa/sca-bindings/sca-jmsbinding-1.1-spec-WD-03.pdf>

Deleted: 1

Deleted: 1

Latest Version:

<http://docs.oasis-open.org/opencsa/sca-bindings/sca-jmsbinding-1.1-spec.doc>
<http://docs.oasis-open.org/opencsa/sca-bindings/sca-jmsbinding-1.1-spec.pdf>

Deleted: -WD-01

Deleted: -WD-01

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 supercedes:

- 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/200712>

Comment [SAJH1]: Resolution to issue BINDINGS-13

Deleted: TBD

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/>.

Notices

Copyright © OASIS® 2005, 2008, All Rights Reserved.

Deleted: 7

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The names "OASIS", [insert specific trademarked names and abbreviations here] are trademarks of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

Table of Contents

1	Introduction	5
1.1	Terminology	5
1.2	Normative References	5
1.3	Non-Normative References	6
2	Operation Selection and Data Binding	7
3	Messaging Bindings	8
4	JMS Binding Schema	9
5	Default Operation Selection and Data Binding behavior	13
5.1	Default Operation Selection	13
5.2	Default Data Binding	13
6	Policy	14
7	Message Exchange Patterns	15
7.1	One-way message exchange (no Callbacks)	15
7.2	Request/response message exchange (no Callbacks)	15
7.3	JMS User Properties	15
7.4	Callbacks	16
7.4.1	Invocation of operations on a bidirectional interface	16
7.4.2	Invocation of operations on a callback interface	16
7.4.3	Use of JMSReplyTo for callbacks for non-SCA JMS applications	16
7.5	Conversations	17
7.5.1	Starting a conversation	17
7.5.2	Continuing a conversation	17
7.5.3	Ending a conversation	17
8	Examples	19
8.1	Minimal Binding Example	19
8.2	URI Binding Example	19
8.3	Binding with Existing Resources Example	19
8.4	Resource Creation Example	20
8.5	Request/Response Example	20
8.6	Use of Predefined Definitions Example	21
8.7	Subscription with Selector Example	21
8.8	Policy Set Example	22
9	Conformance	23
A.	JMS Binding Schema	24
B.	Acknowledgements	27
C.	Non-Normative Text	28
D.	Revision History	29

1 Introduction

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

Deleted: [1]

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.

Comment [sajh2]: Resolution of issue raised on OSOA mailing list

Deleted: Further work is needed for specifying the simplifications that are possible for messaging bindings used for SCA wires (see section 3: Open Issues).¶

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].

Deleted: [RFC2119]

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

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

Comment [sajh3]: Resolution of issue BINDINGS-14

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/>

Comment [SAJH4]: Resolution to issue BINDINGS-30

Deleted: 1

[WSDL] WSDL Specification

WSDL 1.1: <http://www.w3.org/TR/wsdl>

Comment [sajh5]: Not referenced

WSDL 2.0: <http://www.w3.org/TR/wsdl20/>

Deleted: [2] Java Enterprise Edition 1.4 specification <http://java.sun.com/j2ee/1.4/>

Deleted: 3

35 **[JCA15]** Java Connector Architecture Specification Version 1.5
36 <http://java.sun.com/j2ee/connector/>

37 **[IETFJMS]** IETF URI Scheme for Java™ Message Service 1.0
38 <http://www.ietf.org/internet-drafts/draft-merrick-jms-uri-03.txt>¹

39 **1.3 Non-Normative References**

40 TBD TBD

Deleted: 4

Comment [SAJH6]: Resolution of issue BINDINGS-20

Deleted: ¶
¶
TBD . TBD . ¶

Formatted: Bullets and Numbering

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

Formatted: English (U.K.)

41 2 Operation Selection and Data Binding

42 In general messaging providers deal with message formats and destinations. There is not usually a built-
43 in concept of "operation" that corresponds to that defined in a WSDL portType ~~WSDL~~. Messages have
44 a format which corresponds in some way to the schema of an input or output message of an operation in
45 the interface of a service or reference, however some means is required in order to identify the specific
46 operation and map the message information in to the required form.

47 No standard means for service providers and consumers to declare and exchange message format
48 information is provided.

49 The process of identifying the operation to be invoked is **operation selection**; that of mapping message
50 information to the required runtime form is **data binding**. The JMS binding defines default operation
51 selection and data binding behavior; SCA providers may provide extensions for custom behavior.

Deleted: t

Deleted: [3]

52 **3 Messaging Bindings**

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

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

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

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

62

4 JMS Binding Schema

63

The JMS binding element is defined by the following schema.

64

```

<binding.jms correlationScheme="QName"?
  initialContextFactory="xs:anyURI"?
  jndiURL="xs:anyURI"?
  requestConnection="QName"?
  responseConnection="QName"?
  operationProperties="QName"?
  name="NCName"?
  requires="list of QName"?
  uri="xsd:anyURI"?
  ... >
  <destination name="xs:anyURI" type="queue or topic"?
    create="always or never or ifnotexist"? >
    <property name="NMTOKEN" type="NMTOKEN"? > *
  </destination? >
  <connectionFactory name="xs:anyURI"
    create="always or never or ifnotexist"? >
    <property name="NMTOKEN" type="NMTOKEN"? > *
  </connectionFactory? >
  <activationSpec name="xs:anyURI"
    create="always or never or ifnotexist"? >
    <property name="NMTOKEN" type="NMTOKEN"? > *
  </activationSpec? >
  <response >
    <destination name="xs:anyURI" type="queue or topic"?
      create="always or never or ifnotexist"? >
      <property name="NMTOKEN" type="NMTOKEN"? > *
    </destination? >
    <connectionFactory name="xs:anyURI"
      create="always or never or ifnotexist"? >
      <property name="NMTOKEN" type="NMTOKEN"? > *
    </connectionFactory? >
    <activationSpec name="xs:anyURI"
      create="always or never or ifnotexist"? >
      <property name="NMTOKEN" type="NMTOKEN"? > *
    </activationSpec? >
  </response? >
  <resourceAdapter name="NMTOKEN"? >
    <property name="NMTOKEN" type="NMTOKEN"? > *
  </resourceAdapter? >
  <headers JMSType="string"?
    JMSCorrelationId="string"?
    JMSDeliveryMode="PERSISTENT or NON_PERSISTENT"?
    JMSTimeToLive="long"?
    JMSPriority="0 .. 9"? >
    <property name="NMTOKEN" type="NMTOKEN"? > *
  </headers? >
  <subscriptionHeaders JMSSelector="string"? >
    <property name="NMTOKEN" type="NMTOKEN"? > *
  </subscriptionHeaders? >
  <operationProperties name="string" nativeOperation="string"? >
    <property name="NMTOKEN" type="NMTOKEN"? > *
    <headers JMSType="string"?

```

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

Comment [SAJH7]: Resolution of issue BINDINGS-33

Deleted: string

Comment [SAJH8]: Resolution to issue BINDINGS-32

Deleted: int

Comment [sajh9]: Resolution of issue BINDINGS-26

Deleted: string

Comment [sajh10]: Resolution of issue BINDINGS-26

Deleted: string

Comment [sajh11]: Resolution of OSOA erratum 1

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: int

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

Deleted: string

121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171

```
JMSCorrelationId="string"?
JMSDeliveryMode="PERSISTENT or NON_PERSISTENT"?
JMSTimeToLive="long"?
JMSPriority="0..9"?
<property name="NMTOKEN" type="NMTOKEN"?*>
</headers?>
</operationProperties?*>
</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 generic JMS binding type. The type 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` – (from binding) URI that identifies the destination, connection factory or activation spec, and other properties to be used to send/receive the JMS message

The value of the `@uri` attribute MUST have the following format, defined by the IETF URI Scheme for Java™ Message Service 1.0 [JMS]. 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:<jms-dest?>
connectionFactoryName=<Connection-Factory-Name> &
destinationType={queue|topic}
deliveryMode=<Delivery-Mode> &
timeToLive=<Time-To-Live> &
priority=<Priority> &
selector=<Selector> &
<User-Property>=<User-Property-Value> & ...
```

When the `@uri` attribute is specified, it is assumed that the referenced resources already exist.

- `/binding.jms/@name` - as defined in the SCA Assembly specification in Section 9, "Binding"
- `/binding.jms/@requires` - as defined in the SCA Assembly specification in Section 9, "Binding"
- `/binding.jms/@correlationScheme` – identifies the correlation scheme used when sending reply or callback messages. Possible values for the `@correlationScheme` attribute are, `sca:MessageID` (the default) where the correlation ID of replies is set to the message ID of the corresponding request; `sca:CorrelationID`, where the correlation ID of replies is set to the correlation ID of the corresponding request, and `sca:None` which indicates that the correlation ID is not set. SCA runtimes MAY allow other values to indicate other correlation schemes.
- `/binding.jms/@initialContextFactory` – the name of the JNDI initial context factory.
- `/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. In this case this `binding.jms` element MUST NOT also contain the corresponding elements.
- `/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. In this case this `binding.jms` element MUST NOT contain a `response` element.
- `/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. In this case this `binding.jms` element MUST NOT contain an `operationProperties` element.

Deleted: string

Comment [sajh16]: Resolution of issue BINDINGS-1

Deleted: int

Deleted: string

Formatted: Attribute

Comment [sajh17]: Resolution of issue BINDINGS-38

Formatted: Attribute

Deleted: URI

Comment [SAJH18]: Resolution of issue BINDINGS-20

Formatted: Attribute, English (U.S.)

Comment [sajh19]: Resolution of issue BINDINGS-12

Deleted:

Formatted: Attribute, Font: Arial

Deleted: URI ...used [1]

Formatted: Font: Bold, Italic

Formatted: Bullets and Numbering

Formatted: Font: Bold, Italic

Formatted: Attribute

Deleted: Valid values are

Formatted: Attribute

Deleted: RequestMsgIDToCorrel

Formatted [2]

Deleted: RequestCorrelIDToCorrelID [3]

Formatted [4]

Comment [SAJH20]: Resolution of issue BINDINGS-33

Deleted: .

Formatted [5]

Deleted: the... must not be present within this binding element [6]

Formatted [7]

Deleted: no

Deleted: must be present [8]

Formatted: Attribute

Formatted [9]

Deleted: no

Formatted: Attribute

Deleted: s must be present [10]

- 172 • **/binding.jms/destination** – identifies the destination that is to be used to process requests by this
173 binding.
- 174 • **/binding.jms/destination/@type** - the type of the request destination. **Valid values are “queue” and**
175 **“topic”**. The default value is **“queue”**. **In either case the runtime MUST ensure a single response is**
176 **delivered for request/response operations.**
- 177 • **/binding.jms/destination/@name** – the name of the destination to which the binding is connected.
178 This may be a JNDI name or a plain destination name.
- 179 • **/binding.jms/destination/@create** – indicates whether the destination should be created when the
180 containing composite is deployed. Valid values are **“always”, “never” and “ifnotexist”**. The default
181 value is **“ifnotexist”**. If **“always”** is specified and the corresponding resource already exists, then **the**
182 **SCA runtime SHOULD** considered **this as** an error.
- 183 • **/binding.jms/destination/property** – defines properties to be used to create the destination, if
184 required.
- 185 • **/binding.jms/connectionFactory** – identifies the connection factory that the binding uses to process
186 request messages. This **can either** be a JNDI name or a plain connection factory name. The
187 attributes of this element follow those defined for the **destination** element. **A binding.jms element**
188 **MUST NOT include both this element and an activationSpec element.**
- 189 • **/binding.jms/activationSpec** – identifies the activation spec that the binding uses to connect to a
190 JMS destination to process request messages. This **can either** be a JNDI name or a plain activation
191 spec name. The attributes of this element follow those defined for the **destination** element.
- 192 • **/binding.jms/response** – defines the resources used for handling response messages (receiving
193 responses for a reference, and sending responses from a service).
- 194 • **/binding.jms/response/destination** – identifies the destination that is to be used to process
195 responses by this binding. Attributes are as for the parent’s **destination** element.
- 196 • **/binding.jms/response/connectionFactory** – identifies the connection factory that the binding uses
197 to process response messages. This **can either** be a JNDI name or a plain connection factory name.
198 The attributes of this element follow those defined for the **destination** element. **A response element**
199 **MUST NOT include both this element and an activationSpec element.**
- 200 • **/binding.jms/response/activationSpec** – identifies the activation spec that the binding uses to
201 connect to a JMS destination to process response messages. This **can either** be a JNDI name or a
202 plain activation spec name. The attributes of this element follow those defined for the **destination**
203 element.
- 204 • **/binding.jms/headers** – this element allows JMS headers to be set to the given values for all
205 operations. These values apply to requests from a reference and responses from a service.
- 206 • **/binding.jms/headers/@JMSType, @JMSCorrelationID, @JMSDeliveryMode,**
207 **@JMSTimeToLive, @JMSPriority** – specifies the value to use for the JMS header property. **The**
208 **value of the @uri attribute MUST NOT include values for these properties if they are specified using**
209 **these attributes. Valid values for @JMSDeliveryMode are “PERSISTENT” and**
210 **“NON_PERSISTENT”; valid values for @JMSPriority are “0” to “9”.**
- 211 • **/binding.jms/headers/property** – specifies the value to use for the specified JMS user property.
- 212 • **/binding.jms/subscriptionHeaders** - this element allows JMS subscription options to be set. These
213 values apply to a service subscribing to the destination or for a reference subscribing to the callback
214 or reply-to destinations.
- 215 • **/binding.jms/subscriptionHeaders/@JMSelector** - specifies the value to use for the JMS selector.
216 The value of the **@uri** attribute MUST NOT include values for this property if it is specified using this
217 attribute.
- 218 • **/binding.jms/resourceAdapter** – specifies name, type and properties of the Resource Adapter Java
219 bean. This is required when the JMS resources are to be created for a JMS provider **that implements**
220 **the JCA 1.5 specification [JCA15]**, and is ignored otherwise. **SCA runtimes MAY place restrictions on**
221 **the properties of the RA Java bean that can be set.** For JMS providers **that do not implement the**

Deleted: Must take on	[11]
Formatted	[12]
Deleted: or	
Formatted	[13]
Formatted	[14]
Comment [SAJH21]:	[15]
Deleted: When “topic”	[16]
Formatted	[17]
Formatted	[18]
Formatted	[19]
Formatted	[20]
Formatted	[21]
Formatted	[22]
Deleted: this should be	
Deleted: may	
Formatted	[23]
Formatted	[24]
Deleted: This elemen	[25]
Formatted	[26]
Deleted: may	
Formatted	[27]
Formatted	[28]
Deleted: may	
Formatted	[29]
Formatted	[30]
Deleted: This elemen	[31]
Formatted	[32]
Deleted: may	
Formatted	[33]
Formatted	[34]
Formatted	[35]
Formatted	[36]
Comment [sajh22]: F	[37]
Formatted	[38]
Formatted	[39]
Formatted	[40]
Formatted	[41]
Deleted: If these attrib	[42]
Formatted	[43]
Comment [sajh23]: F	[44]
Deleted: JCA 1.5-compliant	
Deleted: [4]	
Deleted: There may be	[45]
Deleted: non-JCA 1.5-	[46]

222 | JCA 1.5 specification, information necessary for resource creation can be added in provider-specific
223 | elements or attributes allowed by the extensibility of the **binding.jms** element.

Deleted: must

Deleted: done

Formatted: Attribute

224 | • **/binding.jms/operationProperties** – specifies various properties that are specific to the processing
225 | of a particular operation.

226 | • **/binding.jms/operationProperties/@name** – The name of the operation in the interface.

227 | • **/binding.jms/operationProperties/@nativeOperation** – The name of the native operation that
228 | corresponds to this operation in the interface.

229 | • **/binding.jms/operationProperties/property** – specifies properties specific to this operation.

230 | • **/binding.jms/operationProperties/headers** – this element allows JMS headers to be set to the
231 | given values for the given operation. These values apply to requests from a reference and responses
232 | from a service.

233 | • **/binding.jms/operationProperties/headers/@JMSType, @JMSCorrelationID,**
234 | **@JMSDeliveryMode, @JMSTimeToLive, @JMSPriority** – specifies the value to use for the JMS
235 | header property. Values specified for particular operations take precedence over those defined in the
236 | binding.jms/headers element or via the binding's @uri attribute.

Formatted: Attribute

Deleted: on the binding

Formatted: Attribute

Deleted: URI

237 | • **/binding.jms/operationProperties/headers/property** – specifies the value to use for the specified
238 | JMS user property.

239 | • **/binding.jms/@{any}** - this is an extensibility mechanism to allow extensibility via attributes.

240 | • **/binding.jms/any** – this is an extensibility mechanism to allow extensibility via elements.

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

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Comment [sajh24]: Resolution of issue BINDINGS-5

245 5 Default Operation Selection and Data Binding 246 behavior

247 This section describes the default behavior for operation selection and data binding for a JMS binding.
248 The SCA runtime MUST support this default behaviour, and MAY provide additional means to override it.

249 5.1 Default Operation Selection

250 When receiving a request at a service, or a callback at a reference, the native operation name is
251 determined as follows:

- 252 • If there is only one operation on the service's interface, then that operation is assumed as the native
253 operation name.
- 254 • Otherwise, if the JMS user property "**scaOperationName**" is present, then its value is used as the
255 native operation name.
- 256 • Otherwise, if the message is a JMS text or bytes message containing XML, then the operation name
257 is taken from the local name of the root element of the XML payload.
- 258 • Otherwise, the native operation name is assumed to be "**onMessage**".

259 The native operation name is then mapped to an operation in the service's interface via a matching
260 **operationProperties** element in the JMS binding. If there is no matching element, the operation name is
261 assumed to be the same as the native operation name.

262 When sending a request from a reference, or a callback from a service, if the interface includes more than
263 one operation then the "**scaOperationName**" JMS user property is set to the name of the operation being
264 invoked.

265 The SCA runtime MAY provide the means for supplying and identifying alternative function selection
266 behaviors.

267 5.2 Default Data Binding

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

272 The message body is mapped to the parameters or return value of the target operation as follows:

- 273 • If there is a single parameter that is a JMSMessage, then the JMSMessage is passed as is.
- 274 • Otherwise, the JMSMessage must be a JMS text message or bytes message containing XML.
- 275 • If there is a single parameter, or for the return value, the JMS text or bytes XML payload is the XML
276 serialization of that parameter according to the WSDL schema for the message.
- 277 • If there are multiple parameters, then they are encoded in XML using the document wrapped style,
278 according to the WSDL schema for the message.

279 The SCA runtime SHOULD provide the means for supplying and identifying alternative data binding
280 behaviors to support any other type of JMS message.

Deleted: .

Formatted: Attribute

Formatted: Bullets and
Numbering

Comment [sajh25]: Resoluti
on of BINDINGS-34

Formatted: Attribute

Deleted: may

Deleted: be

Deleted: operation

Formatted: Attribute

Formatted: Attribute

Deleted: To support any other
means of function selection,

Deleted: t

Deleted: may

Deleted: or return value

Deleted: To support any other
type of JMS message,

Deleted: t

Deleted: should

Deleted: .

281 6 Policy

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

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

```
287 <bindingType type="binding.jms"  
288             alwaysProvides="jms"  
289             mayProvide="atLeastOnce atMostOnce ordered conversation|all"/>
```

Deleted: may natively provide support for some

Formatted: Attribute, English (U.S.)

Comment [sajh26]: Resolution of BINDINGS-6

Comment [sajh27]: Resolution of issue BINDINGS-18

290

7 Message Exchange Patterns

291
292
293
294
295
296

This section describes the message exchange patterns that are possible when using the JMS binding, including one-way, request/response, callbacks and conversations. 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.

297

7.1 One-way message exchange (no Callbacks)

298
299
300
301
302
303
304
305
306
307

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.

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 treat this as an error.

The use of one-way exchanges when using a bidirectional interface is described in section 7.4.

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

308

7.2 Request/response message exchange (no Callbacks)

309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326

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.

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 recorded 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.

The use of request/response exchanges when using a bidirectional interface is described in section 7.4.

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

327

7.3 JMS User Properties

328
329
330
331
332

This protocol assigns specific behavior to JMS user properties:

- "*scaCallbackDestination*" holds the name of the JMS Destination to which callback messages are sent.
- "*scaConversationStart*" indicates that a conversation is to be started, its value is the identifier for the conversation.

Comment [SAJH28]: Resolution of issue BINDINGS-35

Formatted: Attribute

Formatted: Attribute

- 333 • "**scaConversationMaxIdleTime**" defines the maximum time that should be allowed between
334 operations in the conversation. Formatted: Attribute
- 335 • "**scaConversationId**" holds the identifier for the conversation. Formatted: Attribute

336 7.4 Callbacks

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

345 7.4.1 Invocation of operations on a bidirectional interface

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

350 When a request message is sent by a reference with a JMS binding for a request/response MEP with a
351 bidirectional interface, the SCA runtime MUST set the **scaCallbackDestination** user property in the
352 message it creates to identify the destination from which it will read callback messages. The SCA runtime
353 MUST set the **JMSReplyTo** header in the message it creates as described in section 7.2. Formatted: Attribute, Font color: Auto, English (U.S.)

354 For both one-way and request/response operations, if the reference has a callback service element with a
355 JMS binding with a request destination, then the SCA runtime MUST use that destination as the one to
356 which callback messages are to be sent, otherwise the SCA runtime MUST provide an appropriate
357 destination for this purpose. Formatted: Attribute, Font color: Auto, English (U.S.)

358 7.4.2 Invocation of operations on a callback interface

359 An SCA service with a callback interface can invoke operations on that callback interface by sending
360 messages to the destination identified by the **scaCallbackDestination** user property in a message that it
361 has received, the **JMSReplyTo** destination of a one-way message that it has received, or the destination
362 identified by the service's callback reference JMS binding. Formatted: Attribute, English (U.S.)

363 When a callback request message is sent by a service with a JMS binding for either a one-way or
364 request/response MEP, the SCA runtime MUST send the callback request message to the JMS
365 destination identified as follows, in order of priority: Formatted: Attribute, English (U.S.)

- 366 • The **scaCallbackDestination** identified by an earlier request, if not null; Formatted: Attribute
- 367 • the **JMSReplyTo** destination identified by an earlier one-way request, if not null; Formatted: Attribute
- 368 • the request destination of the service's callback reference JMS binding, if specified.

369 If no destination is identified then the SCA runtime SHOULD record an error, and MUST throw an
370 exception to the caller of the callback operation.

371 The SCA runtime MUST set the **JMSReplyTo** destination and correlation identifier in the callback request
372 message as defined in sections 7.1 or 7.2 as appropriate for the type of the callback operation invoked. Formatted: Attribute

373 7.4.3 Use of JMSReplyTo for callbacks for non-SCA JMS applications

374 When interacting with non-SCA JMS applications, the assembler can choose to model a
375 request/response message exchange using a bidirectional interface. In this case it is likely that the non-
376 SCA JMS application does not support the use of the **scaCallbackDestination** user property. To support
377 this, for one-way messages the **JMSReplyTo** header can be used to identify the destination to be used to
378 deliver callback messages, as described in sections 7.4.1 and 7.4.2. Formatted: Attribute

379 **7.5 Conversations**

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

386 **7.5.1 Starting a conversation**

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

Formatted: Attribute

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

Formatted: Attribute

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

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

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

402 **7.5.2 Continuing a conversation**

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

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

410 The SCA runtime MUST consider messages received either containing a conversation identifier that does
411 not correspond to a started conversation, or containing the *scaConversationStart* user property with a
412 conversation identifier that matches an active conversation, as an error, and MUST NOT deliver such
413 messages.

Formatted: Attribute

414 **7.5.3 Ending a conversation**

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

Formatted: Attribute

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

Deleted: This section describes the protocol that is used to support callbacks and conversational behavior when using the JMS binding. These apply to a JMS binding on a service or reference with a bidirectional interface.¶

<#>JMS User Properties¶

This protocol assigns specific behavior to JMS user properties:¶

<#>"scaCallbackQueue" holds the name of the queue to which callback messages are sent. ¶

<#>"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.¶

<#>Callbacks¶

A callback is the invocation of an operation on a service's callback interface.¶

When an SCA component with a reference with a bidirectional interface and JMS binding ("the sender") invokes an operation on that interface, the JMS message that is sent may identify the target for callbacks using the "scaCallbackQueue" user property, or for one-way operations the JMS replyTo header.¶

The invoked SCA component ("the receiver") can only invoke operations on the callback interface during the execution of the target operation for such a message, or when the service's callback binding identifies a fixed callback queue. The sender's callback queue can be specified on the reference's JMS callback binding, or it can be left to the runtime to provide one, by omitting the callbackService ... [47]

Formatted: Bullets and Numbering

428 8 Examples

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

Formatted: Attribute

Formatted: Attribute

Formatted: Attribute

432 8.1 Minimal Binding Example

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

```
435 <?xml version="1.0" encoding="ASCII"?>
436 <composite xmlns="http://www.oesa.org/xmlns/sca/1.0"
437           name="MyValueComposite">
438
439   <service name="MyValueService">
440     <interface.java interface="services.myvalue.MyValueService"/>
441     <binding.jms/>
442   </service>
443
444   <reference name="StockQuoteService">
445     <interface.java interface="services.stockquote.StockQuoteService"/>
446     <binding.jms/>
447   </reference>
448 </composite>
```

449

450 8.2 URI Binding Example

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

Formatted: Attribute

Deleted: URI

```
453 <?xml version="1.0" encoding="ASCII"?>
454 <composite xmlns="http://www.oesa.org/xmlns/sca/1.0"
455           name="MyValueComposite">
456
457   <service name="MyValueService">
458     <interface.java interface="services.myvalue.MyValueService"/>
459     <binding.jms uri="jms:MyValueServiceQueue?
460                   activationSpecName=MyValueServiceAS&
461                   ... "/>
462   </service>
463
464   <reference name="StockQuoteService">
465     <interface.java interface="services.stockquote.StockQuoteService"/>
466     <binding.jms uri="jms:StockQuoteServiceQueue?
467                   connectionFactoryName=StockQuoteServiceQCF&
468                   deliveryMode=1&
469                   ... "/>
470   </reference>
471 </composite>
```

472

473 8.3 Binding with Existing Resources Example

474 The following example shows the JMS binding using existing resources:

```
475 <?xml version="1.0" encoding="ASCII"?>
476 <composite xmlns="http://www.oesa.org/xmlns/sca/1.0"
```

```

477         name="MyValueComposite">
478
479     <service name="MyValueService">
480         <interface.java interface="services.myvalue.MyValueService"/>
481         <binding.jms>
482             <destination name="MyValueServiceQ" create="never"/>
483             <activationSpec name="MyValueServiceAS" create="never"/>
484         </binding.jms>
485     </service>
486 </composite>

```

487

488 8.4 Resource Creation Example

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

```

491 <?xml version="1.0" encoding="ASCII"?>
492 <composite xmlns="http://www.osoa.org/xmlns/sca/1.0"
493     name="MyValueComposite">
494
495     <service name="MyValueService">
496         <interface.java interface="services.myvalue.MyValueService"/>
497         <binding.jms>
498             <destination name="MyValueServiceQueue" create="always">
499                 <property name="prop1" type="string">XYZ</property>
500             </destination>
501             <activationSpec name="MyValueServiceAS" create="always">
502                 <resourceAdapter name="com.example.JMSRA"/>
503             </activationSpec>
504         </binding.jms>
505     </service>
506
507     <reference name="StockQuoteService">
508         <interface.java interface="services.stockquote.StockQuoteService"/>
509         <binding.jms>
510             <destination name="StockQuoteServiceQueue"/>
511             <connectionFactory name="StockQuoteServiceQCF"/>
512             <resourceAdapter name="com.example.JMSRA"/>
513         </binding.jms>
514     </reference>
515 </composite>

```

515

516 8.5 Request/Response Example

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

```

520 <?xml version="1.0" encoding="ASCII"?>
521 <composite xmlns="http://www.osoa.org/xmlns/sca/1.0"
522     name="MyValueComposite">
523
524     <service name="MyValueService">
525         <interface.java interface="services.myvalue.MyValueService"/>
526         <binding.jms correlationScheme="sca:MessageId">
527             <destination name="MyValueServiceQ" create="never"/>
528             <activationSpec name="MyValueServiceAS" create="never"/>
529         </binding.jms>
530     </service>
531
532     <reference name="StockQuoteService">

```

Deleted: r

Formatted: Attribute

Deleted: queue

Deleted: in

Deleted: RequestMsgIdToCorrelId

```

533     <interface.java interface="services.stockquote.StockQuoteService"/>
534     <binding.jms correlationScheme="sca:MessageId">
535         <destination name="StockQuoteServiceQueue"/>
536         <connectionFactory name="StockQuoteServiceQCF"/>
537         <response>
538             <destination name="MyValueResponseQueue"/>
539             <activationSpec name="MyValueResponseAS"/>
540         </response>
541     </binding.jms>
542 </reference>
543 </composite>

```

Deleted: RequestMsgIdToCoreId

544 8.6 Use of Predefined Definitions Example

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

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

Deleted: resource

```

548 <?xml version="1.0" encoding="ASCII"?>
549 <definitions targetNamespace="http://acme.com"
550             xmlns="http://www.osoa.org/xmlns/sca/1.0">
551     <binding.jms name="StockQuoteService">
552         <destination name="StockQuoteServiceQueue" create="never"/>
553         <connectionFactory name="StockQuoteServiceQCF" create="never"/>
554     </binding.jms>
555 </definitions>

```

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

Formatted: Attribute, English (U.S.)

```

557 <?xml version="1.0" encoding="ASCII"?>
558 <composite xmlns="http://www.osoa.org/xmlns/sca/1.0"
559           xmlns:acme="http://acme.com"
560           name="MyValueComposite">
561     <reference name="MyValueService">
562         <interface.java interface="services.myvalue.MyValueService"/>
563         <binding.jms requestConnection="acme:StockQuoteService"/>
564     </reference>
565 </composite>
566

```

567

568 8.7 Subscription with Selector Example

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

```

571 <?xml version="1.0" encoding="ASCII"?>
572 <composite xmlns="http://www.osoa.org/xmlns/sca/1.0"
573           name="MyValueComposite">
574     <service name="MyValueService">
575         <interface.java interface="services.myvalue.MyValueService"/>
576         <binding.jms>
577             <destination name="MyValueServiceTopic" create="never"/>
578             <connectionFactory name="StockQuoteServiceTCF" create="never" />
579             <subscriptionHeaders JMSSelector="Price>1000"/>
580         </binding.jms>
581     </service>
582 </composite>

```

Comment [sajh29]: Resolution of issue BINDINGS-12

583 8.8 Policy Set Example

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

```
587 <policySet name="JMSPolicy"  
588   provides="priority log"  
589   appliesTo="binding.jms">  
590  
591   <intentMap provides="priority" default="medium">  
592     <qualifier name="high">  
593       <headers JMSPriority="9"/>  
594     </qualifier>  
595     <qualifier name="medium">  
596       <headers JMSPriority="4"/>  
597     </qualifier>  
598     <qualifier name="low">  
599       <headers JMSPriority="0"/>  
600     </qualifier>  
601   </intentMap>  
602  
603   <intentMap provides="log">  
604     <qualifier>  
605       <headers>  
606         <property name="user_example_log">logged</property>  
607       </headers>  
608     </qualifier>  
609   </intentMap>  
610 </policySet>
```

Deleted: "p

Deleted: "

Formatted: Attribute, English (U.S.)

Formatted: Attribute, English (U.S.)

Formatted: Attribute, English (U.S.)

Formatted: Attribute, English (U.S.)

611

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

```
613 <reference name="StockQuoteService" requires="priority.high log">  
614   <interface.java interface="services.stockquote.StockQuoteService"/>  
615   <binding.jms>  
616     <destination name="StockQuoteServiceQueue"/>  
617     <connectionFactory name="StockQuoteServiceQCF"/>  
618   </binding.jms>  
619 </reference>
```

620

621

9 Conformance

622

The XML schema available at the namespace URI, defined by this specification, is considered to be authoritative and takes precedence over the XML Schema defined in the appendix of this document.

623

Deleted: TBD

A. JMS Binding Schema

```

625 <?xml version="1.0" encoding="UTF-8"?>
626 <!-- (c) Copyright SCA Collaboration 2006 -->
627 <schema xmlns="http://www.w3.org/2001/XMLSchema"
628       targetNamespace="http://docs.oasis-open.org/ns/opencsa/sca/200712"
629       xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200712"
630       elementFormDefault="qualified">
631
632   <include schemaLocation="sca-core.xsd"/>
633
634   <complexType name="JMSBinding">
635     <complexContent>
636       <extension base="sca:Binding">
637         <sequence>
638           <element name="destination" type="sca:Destination" minOccurs="0"/>
639           <element name="connectionFactory" type="sca:ConnectionFactory"
640             minOccurs="0"/>
641           <element name="activationSpec" type="sca:ActivationSpec"
642             minOccurs="0"/>
643           <element name="response" type="sca:Response" minOccurs="0"/>
644           <element name="headers" type="sca:Headers" minOccurs="0"/>
645           <element name="subscriptionHeaders" type="sca:SubscriptionHeaders"
646             minOccurs="0"/>
647           <element name="resourceAdapter" type="sca:ResourceAdapter"
648             minOccurs="0"/>
649           <element name="operationProperties" type="sca:OperationProperties"
650             minOccurs="0" maxOccurs="unbounded"/>
651           <any namespace="##other" processContents="lax"
652             minOccurs="0" maxOccurs="unbounded"/>
653         </sequence>
654         <attribute name="correlationScheme" type="QName"
655           default="sca:MessageId"/>
656         <attribute name="initialContextFactory" type="anyURI"/>
657         <attribute name="jndiURL" type="anyURI"/>
658         <attribute name="requestConnection" type="QName"/>
659         <attribute name="responseConnection" type="QName"/>
660         <attribute name="operationProperties" type="QName"/>
661         <anyAttribute/>
662       </extension>
663     </complexContent>
664   </complexType>
665
666   <simpleType name="CreateResource">
667     <restriction base="string">
668       <enumeration value="always"/>
669       <enumeration value="never"/>
670       <enumeration value="ifnotexist"/>
671     </restriction>
672   </simpleType>
673
674   <complexType name="Destination">
675     <sequence>
676       <element name="property" type="sca:BindingProperty"
677         minOccurs="0" maxOccurs="unbounded"/>
678     </sequence>
679     <attribute name="name" type="anyURI" use="required"/>
680     <attribute name="type" use="optional" default="queue">
681       <simpleType>
682         <restriction base="string">
683           <enumeration value="queue"/>

```

Comment [sajh30]: Resolution of issue BINDINGS-13

Deleted: <http://www.osoa.org/xmlns/sca/1.0>

Comment [sajh31]: Needs resolving for CD

Deleted: <http://www.osoa.org/xmlns/sca/1.0>

Comment [sajh32]: Resolution of issue BINDINGS-12

Comment [SAJH33]: Resolution to issue BINDINGS-33

Deleted: RequestMsgIDToCorrelID

Deleted:

```

<simpleType>¶
<restriction
base="string">¶
<enumeration
value="RequestMsgIDToCorrelID"/>¶
<enumeration
value="RequestCorrelIDToCorrelID"/>¶
<enumeration
value="None"/>¶
</restriction>¶
</simpleType>¶
</attribute>¶
¶

```

Comment [sajh34]: Resolution of OSOA erratum 1

Deleted: string

```

684         <enumeration value="topic"/>
685     </restriction>
686 </simpleType>
687 </attribute>
688 <attribute name="create" type="sca:CreateResource"
689     use="optional" default="ifnotexist"/>
690 </complexType>
691
692 <complexType name="ConnectionFactory">
693     <sequence>
694         <element name="property" type="sca:BindingProperty"
695             minOccurs="0" maxOccurs="unbounded"/>
696     </sequence>
697     <attribute name="name" type="anyURI" use="required"/>
698     <attribute name="create" type="sca:CreateResource"
699         use="optional" default="ifnotexist"/>
700 </complexType>
701
702 <complexType name="ActivationSpec">
703     <sequence>
704         <element name="property" type="sca:BindingProperty"
705             minOccurs="0" maxOccurs="unbounded"/>
706     </sequence>
707     <attribute name="name" type="anyURI" use="required"/>
708     <attribute name="create" type="sca:CreateResource"
709         use="optional" default="ifnotexist"/>
710 </complexType>
711
712 <complexType name="Response">
713     <sequence>
714         <element name="destination" type="sca:Destination" minOccurs="0"/>
715         <element name="connectionFactory" type="sca:ConnectionFactory"
716             minOccurs="0"/>
717         <element name="activationSpec" type="sca:ActivationSpec" minOccurs="0"/>
718     </sequence>
719 </complexType>
720
721 <complexType name="Headers">
722     <sequence>
723         <element name="property" type="sca:BindingProperty"
724             minOccurs="0" maxOccurs="unbounded"/>
725     </sequence>
726     <attribute name="JMSType" type="string"/>
727     <attribute name="JMSCorrelationID" type="string"/>
728     <attribute name="JMSDeliveryMode">
729         <simpleType>
730             <restriction base="string">
731                 <enumeration value="PERSISTENT"/>
732                 <enumeration value="NON_PERSISTENT"/>
733             </restriction>
734         </simpleType>
735     </attribute>
736     <attribute name="JMSTimeToLive" type="long"/>
737     <attribute name="JMSPriority">
738         <simpleType>
739             <restriction base="string">
740                 <enumeration value="0"/>
741                 <enumeration value="1"/>
742                 <enumeration value="2"/>
743                 <enumeration value="3"/>
744                 <enumeration value="4"/>
745                 <enumeration value="5"/>
746                 <enumeration value="6"/>
747                 <enumeration value="7"/>

```

Deleted: string

Deleted: string

Deleted: string

Deleted: type="string"/

Deleted: int

Deleted: type="string"/

```

748         <enumeration value="8"/>
749         <enumeration value="9"/>
750     </restriction>
751 </simpleType>
752 </attribute>
753 </complexType>
754
755 <complexType name="SubscriptionHeaders">
756 <sequence>
757 <element name="property" type="sca:BindingProperty"
758 minOccurs="0" maxOccurs="unbounded"/>
759 </sequence>
760 <attribute name="JMSSelector" type="string"/>
761 </complexType>
762
763 <complexType name="ResourceAdapter">
764 <sequence>
765 <element name="property" type="sca:BindingProperty"
766 minOccurs="0" maxOccurs="unbounded"/>
767 </sequence>
768 <attribute name="name" type="string" use="required"/>
769 </complexType>
770
771 <complexType name="OperationProperties">
772 <sequence>
773 <element name="property" type="sca:BindingProperty"
774 minOccurs="0" maxOccurs="unbounded"/>
775 <element name="headers" type="sca:Headers"/>
776 </sequence>
777 <attribute name="name" type="string" use="required"/>
778 <attribute name="nativeOperation" type="string"/>
779 </complexType>
780
781 <complexType name="BindingProperty">
782 <simpleContent>
783 <extension base="string">
784 <attribute name="name" type="NMTOKEN"/>
785 <attribute name="type" type="string" use="optional"
786 default="xs:string"/>
787 </extension>
788 </simpleContent>
789 </complexType>
790
791 <element name="binding.jms" type="sca:JMSBinding"
792 substitutionGroup="sca:binding"/>
793 </schema>

```

Comment [sajh35]: Resolution of issue BINDINGS-1

Comment [sajh36]: Resolution of issue BINDINGS-12

Deleted: string

Deleted: string

Comment [sajh37]: Resolution of OSOA erratum 1

794

795 **B. Acknowledgements**

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

798 **Participants:**

799 [Participant Name, Affiliation | Individual Member]

800 [Participant Name, Affiliation | Individual Member]

801

C. Non-Normative Text

803

D. Revision History

804 [optional; should not be included in OASIS Standards]

805

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	<p>Updated text for RFC2119 conformance</p> <p>Updates to resolve following issues:</p> <p>BINDINGS-1</p> <p>BINDINGS-5</p> <p>BINDINGS-6</p> <p>BINDINGS-12</p> <p>BINDINGS-14</p> <p>BINDINGS-18</p> <p>BINDINGS-26</p> <p>Applied updates discussed at Bindings TC meeting of 27th March</p>
3	2008-06-19	Simon Holdsworth	* Applied most of the editorial changes from Eric Johnson's review
4	2008-08-01	Simon Holdsworth	<p>Updates to resolve following issues:</p> <p>BINDINGS-13 (JMS part)</p> <p>BINDINGS-20 (complete)</p> <p>BINDINGS-30 (JMS part)</p> <p>BINDINGS-32 (JMS part)</p> <p>BINDINGS-33 (complete)</p> <p>BINDINGS-34 (complete)</p> <p>BINDINGS-35 (complete)</p> <p>BINDINGS-38 (JMS part)</p>

806

Page 10: [1] Deleted	Simon Holdsworth	19/06/2008 15:00:00
URI		
Page 10: [1] Deleted	Simon Holdsworth	19/06/2008 15:00:00
used		
Page 10: [2] Formatted	Simon Holdsworth	01/08/2008 14:19:00
Attribute		
Page 10: [2] Formatted	Simon Holdsworth	01/08/2008 14:19:00
Attribute		
Page 10: [3] Deleted	Simon Holdsworth	01/08/2008 14:18:00
<i>R</i>		
Page 10: [3] Deleted	Simon Holdsworth	01/08/2008 14:18:00
<i>equestCorrelIIDToCorrelIID</i>		
Page 10: [4] Formatted	Simon Holdsworth	19/06/2008 15:06:00
Attribute		
Page 10: [4] Formatted	Simon Holdsworth	01/08/2008 14:20:00
Attribute		
Page 10: [5] Formatted	Simon Holdsworth	19/06/2008 15:06:00
Attribute		
Page 10: [5] Formatted	Simon Holdsworth	19/06/2008 15:06:00
Attribute		
Page 10: [5] Formatted	Simon Holdsworth	19/06/2008 15:06:00
Attribute		
Page 10: [5] Formatted	Simon Holdsworth	19/06/2008 15:06:00
Attribute		
Page 10: [5] Formatted	Simon Holdsworth	19/06/2008 15:06:00
Attribute		
Page 10: [6] Deleted	Simon Holdsworth	12/03/2008 14:19:00
the		
Page 10: [6] Deleted	Simon Holdsworth	12/03/2008 14:20:00
must not be present within this binding element		
Page 10: [7] Formatted	Simon Holdsworth	19/06/2008 15:06:00

Attribute

Page 10: [7] Formatted	Simon Holdsworth	19/06/2008 15:07:00
-------------------------------	-------------------------	----------------------------

Attribute

Page 10: [7] Formatted	Simon Holdsworth	19/06/2008 15:07:00
-------------------------------	-------------------------	----------------------------

Attribute

Page 10: [8] Deleted	Simon Holdsworth	12/03/2008 14:21:00
-----------------------------	-------------------------	----------------------------

must be present within this binding element

Page 10: [9] Formatted	Simon Holdsworth	19/06/2008 15:07:00
-------------------------------	-------------------------	----------------------------

Attribute

Page 10: [9] Formatted	Simon Holdsworth	19/06/2008 15:07:00
-------------------------------	-------------------------	----------------------------

Attribute

Page 10: [9] Formatted	Simon Holdsworth	19/06/2008 15:07:00
-------------------------------	-------------------------	----------------------------

Attribute

Page 10: [10] Deleted	Simon Holdsworth	12/03/2008 14:21:00
------------------------------	-------------------------	----------------------------

s must be present within this binding element

Page 11: [11] Deleted	Simon Holdsworth	12/03/2008 14:21:00
------------------------------	-------------------------	----------------------------

Must take one of the values

Page 11: [12] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [13] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [14] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [15] Comment [SAJH21]	Simon Holdsworth	01/08/2008 14:16:00
---------------------------------------	-------------------------	----------------------------

Resolution of issue BINDINGS-35

Page 11: [16] Deleted	Simon Holdsworth	01/08/2008 14:01:00
------------------------------	-------------------------	----------------------------

When “*topic*” is specified, then all the operations in the interface that corresponds to the binding must be one-way

Page 11: [17] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [18] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [19] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [20] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [21] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [22] Formatted	Simon Holdsworth	19/06/2008 15:07:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [23] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [24] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [25] Deleted	Simon Holdsworth	12/03/2008 14:24:00
------------------------------	-------------------------	----------------------------

This element is mutually exclusive with the

Page 11: [26] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [27] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [28] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [29] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [30] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [31] Deleted	Simon Holdsworth	12/03/2008 14:25:00
------------------------------	-------------------------	----------------------------

This element is mutually exclusive with the

Page 11: [32] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [33] Formatted	Simon Holdsworth	19/06/2008 15:08:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [34] Formatted	Simon Holdsworth	19/06/2008 15:09:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [35] Formatted	Simon Holdsworth	19/06/2008 15:09:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [36] Formatted	Simon Holdsworth	19/06/2008 15:09:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [37] Comment [sajh22]	Simon Holdsworth	01/08/2008 14:16:00
---------------------------------------	-------------------------	----------------------------

Resolution of issue BINDINGS-1

Page 11: [38] Formatted	Simon Holdsworth	19/06/2008 15:09:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [39] Formatted	Simon Holdsworth	19/06/2008 15:09:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [40] Formatted	Simon Holdsworth	19/06/2008 15:09:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [41] Formatted	Simon Holdsworth	19/06/2008 15:09:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [42] Deleted	Simon Holdsworth	12/03/2008 14:27:00
------------------------------	-------------------------	----------------------------

If these attributes are specified they must not appear in the URI

Page 11: [43] Formatted	Simon Holdsworth	19/06/2008 15:10:00
--------------------------------	-------------------------	----------------------------

Attribute

Page 11: [44] Comment [sajh23]	Simon Holdsworth	01/08/2008 14:16:00
---------------------------------------	-------------------------	----------------------------

Resolution of issue BINDINGS-12

Page 11: [45] Deleted	Simon Holdsworth	12/03/2008 15:55:00
------------------------------	-------------------------	----------------------------

There may be a restriction, depending on the deployment platform, about specifying properties of the RA Java Bean.

Page 11: [46] Deleted	Simon Holdsworth	19/06/2008 15:10:00
------------------------------	-------------------------	----------------------------

non-JCA 1.5-compliant

Page 18: [47] Deleted	Simon Holdsworth	12/03/2008 14:07:00
------------------------------	-------------------------	----------------------------

This section describes the protocol that is used to support callbacks and conversational behavior when using the JMS binding. These apply to a JMS binding on a service or reference with a bidirectional interface.

JMS User Properties

This protocol assigns specific behavior to JMS user properties:

"scaCallbackQueue" holds the name of the queue to which callback messages are sent.

"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.

Callbacks

A callback is the invocation of an operation on a service's callback interface.

When an SCA component with a reference with a bidirectional interface and JMS binding ("the sender") invokes an operation on that interface, the JMS message that is sent may identify the target for callbacks using the "scaCallbackQueue" user property, or for one-way operations the JMS replyTo header.

The invoked SCA component ("the receiver") can only invoke operations on the callback interface during the execution of the target operation for such a message, or when the service's callback binding identifies a fixed callback queue. The sender's callback queue can be specified on the reference's JMS callback binding, or it can be left to the runtime to provide one, by omitting the callbackService element, the JMS callback binding, or omitting the uri and destination from the JMS callback binding.

Conversations

A conversation is a sequence of operations between two parties that have a common context. The conversation may include a mixture of operations in either direction between the two parties. Interfaces must be marked as conversational in order to ensure that the runtime manages the lifecycle of this context.

Either the sender or receiver must start a conversation when an operation is invoked on a conversational interface and there is no active conversation with the other party. This is done by including the "scaConversationStart" user property in the JMS message with the value set to the required conversation identifier. A new runtime context is associated with the conversation identifier in both the sender and receiver.

The message that starts the conversation may also include the "scaConversationMaxIdleTime" user property; if not present the maximum idle time for the conversation is derived 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 sender may provide a specific callback queue for the identified conversation by including a value for the "scaCallbackQueue" user property.

Subsequent operations between the sender and receiver that are part of this conversation must include the "scaConversationId" user property in the JMS message, set to the conversation identifier. The message may also include an updated value of the "scaConversationMaxIdleTime" property. The value of "scaCallbackQueue" is ignored within a conversation in messages after the one that starts the conversation.

When an operation is invoked either by the sender or receiver that is marked as "endsConversation", or the maximum idle time is exceeded, then the conversation identifier and associated context is discarded after the operation has been processed. The idle time is defined within the sender and receiver as the amount of time since the sender/receiver last completed processing of an operation that is part of the conversation. There may be times when the sender or receiver ends the conversation before the other does. In that case if one party

does invoke an operation on the other, it is treated as being after the conversation has ended and is an error.

Operations invoked on other parties must not be considered part of this conversation and must use different conversation identifiers.

Messages received containing a conversation identifier that does not correspond to a started conversation, or containing a start conversation property with a conversation identifier that matches an active conversation, should be treated as errors and should not be processed. Conversation identifiers may be reused. In particular, runtimes do not have to guarantee unique conversation identifiers and do not have to be able to identify an ended conversation indefinitely, although they may do for some period after the conversation ends. Due to the long-running nature of conversations, runtimes should ensure conversation context is available across server restarts, although they may choose to treat a restart as implicitly ending the conversation.

Component implementation specifications define the manner in which the context that is associated with the conversation identifier is made available to component implementations.