



Test Assertions for the SCA_J Common Annotations and APIs Version 1.1 Specification

Working Draft 5

15 June 2009

Specification URIs:

This Version:

<http://docs.oasis-open.org/sca-assembly/sca-j-caa-1.1-test-assertions-WD5.html>
<http://docs.oasis-open.org/sca-assembly/sca-j-caa-1.1-test-assertions-WD5.odt>
<http://docs.oasis-open.org/sca-assembly/sca-j-caa-1.1-test-assertions-WD5.pdf> (Authoritative)

Previous Version:

Latest Version:

<http://docs.oasis-open.org/sca-assembly/sca-j-caa-1.1-test-assertions.html>
<http://docs.oasis-open.org/sca-assembly/sca-j-caa-1.1-test-assertions.odt>
<http://docs.oasis-open.org/sca-assembly/sca-j-caa-1.1-test-assertions.pdf> (Authoritative)

Technical Committee:

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

Chair(s):

Mark Combellack	Avaya
David Booz	IBM

Editor(s):

Mike Edwards,	IBM
---------------	-----

Related Work:

This document is related to:

- Service Component Architecture Java Common Annotations and APIs Specification Version 1.1

Declared XML Namespace(s):

None

Abstract:

This document defines the Test Assertions for the SCA J CAA specification.

The Test Assertions represent the testable items relating to the normative statements made in the SCA Java Common Annotations and APIs specification. The Test Assertions provide a bridge between the normative statements in the specification and the conformance TestCases which are designed to check that an SCA runtime conforms to the requirements of the specification.

Status:

This document was last revised or approved by the OASIS Service Component Architecture / J (SCA-J) 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-j/>.

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-j/ipr.php>).

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

Notices

Copyright © OASIS® 2009. All Rights Reserved.

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, abbreviations, etc. 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

2	1 Introduction.....	5
3	1.1 Example Test Assertion.....	5
4	1.2 Terminology.....	5
5	1.3 Normative References.....	6
6	1.4 Non-normative References.....	6
7	2 Test Assertions.....	7
8	2.1 Section 2.....	7
9	2.2 Section 3.....	10
10	2.3 Section 4.....	12
11	2.4 Section 5.....	18
12	2.5 Section 8.....	18
13	2.6 Section 9.....	21
14	2.7 Section 10.....	22
15	2.8 Section 11.....	37
16	3 Cross Mapping of Conformance Statements to Assertions.....	38
17	4 Conformance.....	42
18		

1 Introduction

This document defines the Test Assertions for the SCA Java Common Annotations and APIs Specification Version 1.1.

The test assertions in this document follow the format defined in the OASIS Test Assertion Guidelines specification [TA-GUIDE].

1.1 Example Test Assertion

Test assertions are presented in a tabular format with rows corresponding to the entry types defined in [TA-GUIDE].

Assertion ID	JCA-TA-xxxx
Source	[JCAx00yy]
Target	<kitchenSink/> element of composite file
Prerequisites	The <kitchenSink/> element has a @drain attribute
Predicate	The @drain attribute value of the <kitchenSink/> element is a URI that identifies a portal into the drainage system of the Domain.
Prescription Level	Mandatory
Tags	kitchenSink drain Domain

Assertion ID: Is a unique ID for the test assertion. Its format starts with a 3 letter string that identifies the specification to which it relates - "JCA" is for the SCA Java Common Annotations and APIs specification. This is followed by "-TA-" to indicate that this identifier is for a test assertion. This is then followed by a unique 4 digit number.

Source: Is the identifier(s) of the normative statement(s) in the specification to which this assertion relates.

Target: Identifies the target which is addressed by this assertion. This is typically some SCA document element, or other SCA artifact but possibly could identify an SCA runtime and its behaviour.

Prerequisites: Defines any prerequisites for this test assertion. The prerequisites may be defined in terms of one or more other test assertions that must be true.

Predicate: The meat of the assertion - something that should evaluate to true or false for the given target.

Prescription Level: Mandatory (for MUST requirements) or Preferred (for SHOULD requirements) or Permitted (for MAY requirements).

Tags: Zero or more labels that may be attached to this test assertion - these tags can be used to group sets of assertions.

1.2 Terminology

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as described in [IETF RFC 2119 \[RFC 2119\]](#)

48
49
50
51
52
53
54
55
56
57
58
59

1.3 Normative References

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

[TA-GUIDE] Test Assertion Guidelines, Draft 0.9.9.6, 16 November, 2008
http://www.oasis-open.org/apps/group_public/download.php/30070/TestAssertionsGuidelines-draft-0-9-9-6.pdf

[JAVACAA] OASIS SCA Java Common Annotations and APIs specification
<http://docs.oasis-open.org/opencsa/sca-j/sca-javacaa-1.1-spec.pdf>

1.4 Non-normative References

None

2 Test Assertions

2.1 Section 2

Assertion ID	JCA-TA-2001
Source	[JCA20001]
Target	Java service marked remotable
Prerequisites	
Predicate	Service does not use method overloading
Prescription Level	mandatory
Tags	"remotable" "overloading"

Assertion ID	JCA-TA-2002
Source	[JCA20002]
Target	Stateless scoped Java implementation instance
Prerequisites	
Predicate	Implementation instance is dispatched on a single thread at one time
Prescription Level	mandatory
Tags	"stateless scope" "single thread"

Assertion ID	JCA-TA-2003
Source	[JCA20003]
Target	Stateless scoped Java implementation instance
Prerequisites	
Predicate	Implementation instance is invoked once through one business method during the lifetime of the implementation instance
Prescription Level	mandatory
Tags	"stateless scope" "lifetime" "invocation"

Assertion ID	JCA-TA-2004
Source	[JCA20004]
Target	Domain level component implemented by a Java class marked "composite"

	scope"
Prerequisites	
Predicate	All clients of the component appear to interact with a single runtime instance of the implementation class
Prescription Level	mandatory
Tags	"composite scope" "invocation"

69

70

Assertion ID	JCA-TA-2005
Source	[JCA20005]
Target	Java implementation class marked with "composite scope" and with "eager initialization"
Prerequisites	Containing component is started
Predicate	The Java implementation class instance is created and initialized
Prescription Level	mandatory
Tags	"composite scope" "lifetime" "eager initialization"

71

72

Assertion ID	JCA-TA-2006
Source	[JCA20006]
Target	Method of Java implementation class with marked with @Init annotation
Prerequisites	Implementation instance is created
Predicate	Method is called
Prescription Level	mandatory
Tags	"composite scope" "lifetime" "eager initialization"

73

74

Assertion ID	JCA-TA-2007
Source	[JCA20007]
Target	Instance of a Java implementation class marked "composite scope"
Prerequisites	Implementation class is used as a component implementation Multiple invocations of services offered by the component are made, which overlap in time
Predicate	Multiple threads run in the single instance of the implementation class
Prescription Level	optional
Tags	"composite scope" "lifetime" "eager initialization"

75

Assertion ID	JCA-TA-2008
Source	[JCA20008]
Target	Java implementation class marked with "composite scope"
Prerequisites	Implementation class is used to implement a component nested inside a composite used to implement a domain level component Component offers at least one service Component has one or more clients for its service(s)
Predicate	Clients of the component appear to interact with a single instance of the implementation class
Prescription Level	mandatory
Tags	"composite scope" "nested component" "instance"

76

77

Assertion ID	JCA-TA-2009
Source	[JCA20009]
Target	Client invocation of a method of a service implementation
Prerequisites	Service method implementation marked "allows pass by reference" Client reference proxy marked "allows pass by reference" Client and service implementation run in the same JVM
Predicate	The invocation uses pass-by-reference call semantics for the input and response parameters
Prescription Level	optional
Tags	"allows pass by reference" "pass by reference" "local"

78

79

Assertion ID	JCA-TA-2010
Source	[JCA20010]
Target	Client invocation of a service implementation
Prerequisites	Service implementation method not marked "allows pass by reference" Client and service implementation run in the same JVM
Predicate	The invocation uses pass-by-value semantics for the input and response parameters
Prescription Level	mandatory
Tags	"allows pass by reference" "pass by value" "local"

80

81

Assertion ID	JCA-TA-2011
Source	[JCA20010]
Target	Client invocation of a service implementation
Prerequisites	Client reference is not marked "allows pass by reference" Client and service implementation run in the same JVM
Predicate	The invocation uses pass-by-value semantics for the input and response parameters
Prescription Level	mandatory
Tags	"allows pass by reference" "pass by value "local"

82

83

84

2.2 Section 3

85

86

Assertion ID	JCA-TA-3001
Source	[JCA30001]
Target	@interface attribute of a <interface.java/> element
Prerequisites	
Predicate	Value of @interface attribute is the fully qualified name of a Java interface class
Prescription Level	mandatory
Tags	"interface element" "interface attribute" "Java interface class"

87

88

Assertion ID	JCA-TA-3002
Source	[JCA30002]
Target	@callbackInterface attribute of <interface.java/> element
Prerequisites	interface.java element has a @callbackInterface attribute
Predicate	Value of @callbackInterface is the fully qualified name of a Java interface class
Prescription Level	mandatory
Tags	"interface element" "callbackInterface attribute" "Java interface class"

89

90

Assertion ID	JCA-TA-3003
--------------	-------------

Source	[JCA30003]
Target	@callbackInterface attribute of a <interface.java/> element
Prerequisites	interface.java element has a @callbackInterface attribute Java interface class referenced by the @interface attribute contains a @Callback annotation which references a Java interface class
Predicate	Value of @callbackInterface attribute is the same Java interface class as the one referenced by the @Callback annotation of the Java interface class referenced by the @interface attribute
Prescription Level	mandatory
Tags	"interface element" "callbackInterface attribute" "Java interface class" "@Callback"

91

92

Assertion ID	JCA-TA-3004
Source	[JCA30004]
Target	<interface.java/> element
Prerequisites	
Predicate	Conforms to the sca-interface-java.xsd schema
Prescription Level	mandatory
Tags	"interface element" "schema"

93

94

Assertion ID	JCA-TA-3005
Source	[JCA30005]
Target	@remotable attribute on <interface.java/> element
Prerequisites	Java interface class referenced by @interface attribute of interface.java element contains a @Remotable annotation
Predicate	@remotable attribute has the value "true"
Prescription Level	mandatory
Tags	"interface element" "remotable attribute" "Java interface class" "@Remotable"

95

96

Assertion ID	JCA-TA-3006
Source	[JCA30006]
Target	Java interface class referenced by @interface attribute of a <interface.java/> element
Prerequisites	

Predicate	Java interface class does not contain any of the annotations: @AllowsPassByReference, @ComponentName, @Constructor, @Context, @Destroy, @EagerInit, @Init, @Intent, @Property, @Qualifier, @Reference, @Scope, @Service.
Prescription Level	mandatory
Tags	"interface element" "SCA annotations" "Java interface class"

97

98

Assertion ID	JCA-TA-3007
Source	[JCA30007]
Target	Java interface class referenced by @callbackInterface attribute of a <interface.java/> element
Prerequisites	
Predicate	Java interface class does not contain any of the annotations: @AllowsPassByReference, @Callback, @ComponentName, @Constructor, @Context, @Destroy, @EagerInit, @Init, @Intent, @Property, @Qualifier, @Reference, @Scope, @Service.
Prescription Level	mandatory
Tags	"interface element" "SCA annotations" "Java interface class"

99

100



101

2.3 Section 4

102

Assertion ID	JCA-TA-4001
Source	[JCA40001]
Target	Constructor of Java component implementation class
Prerequisites	Component is at the start of the Constructing state of its lifecycle
Predicate	Constructor is invoked by the SCA runtime
Prescription Level	mandatory
Tags	"lifecycle" "constructing" "constructor"

103

104

Assertion ID	JCA-TA-4002
Source	[JCA40002]
Target	Constructor of Java component implementation class
Prerequisites	Constructor is invoked by the SCA runtime at the start of the Constructing state of its lifecycle

Predicate	Constructor references and properties are injected when the constructor is invoked
Prescription Level	mandatory
Tags	"lifecycle" "constructing" "constructor" "injection" "reference" "property"

105

106

Assertion ID	JCA-TA-4003
Source	[JCA40003]
Target	Java component implementation class in Constructing state
Prerequisites	Constructor method completes successfully
Predicate	Component implementation transitions to the Injecting state
Prescription Level	mandatory
Tags	"lifecycle" "constructing" "injecting" "transition"

107

Assertion ID	JCA-TA-4004
Source	[JCA40004]
Target	Java component implementation class in Constructing state
Prerequisites	Constructor method throws an exception
Predicate	Component implementation class is transitioned to the Terminating state
Prescription Level	mandatory
Tags	"lifecycle" "constructor" "exception" "terminating"

108

Assertion ID	JCA-TA-4005
Source	[JCA40005]
Target	Java component implementation class in Injecting state
Prerequisites	Implementation has one or more properties
Predicate	Properties are injected (via field or setter injection) before any other actions occur
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "property"

109

Assertion ID	JCA-TA-4006
Source	[JCA40006]
Target	Java component implementation class in Injecting state
Prerequisites	Implementation has one or more references Implementation has one or more properties

Predicate	References are injected (via field or setter injection) after properties have been injected
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "reference"

110

111

Assertion ID	JCA-TA-4007
Source	[JCA40007]
Target	Java component implementation with one or more injected property / reference
Prerequisites	Implementation code does not use synchronization
Predicate	Injected properties and references are visible to the component implementation
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "synchronization" "property" "reference"

112

Assertion ID	JCA-TA-4008
Source	[JCA40008]
Target	Java component implementation in injecting state
Prerequisites	
Predicate	Service methods are not invoked
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "service methods"

113

Assertion ID	JCA-TA-4009
Source	[JCA40009]
Target	Java component implementation in injecting state
Prerequisites	All properties and references have been injected
Predicate	Component implementation transitions to the Initializing state
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "transition" "initializing"

114

Assertion ID	JCA-TA-4010
Source	[JCA40010]
Target	Java component implementation in injecting state
Prerequisites	Exception is thrown while a property or a reference is being injected

Predicate	Component implementation transitions to the Destroying state
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "exception" "transition" "destroying"

115

Assertion ID	JCA-TA-4011
Source	[JCA40011]
Target	Java component implementation entering initializing state
Prerequisites	Implementation has a method annotated with @Init
Predicate	Method marked with @Init is invoked
Prescription Level	mandatory
Tags	"lifecycle" "initializing" "@Init"

116

117

Assertion ID	JCA-TA-4012
Source	[JCA40012]
Target	Java component implementation in initializing state
Prerequisites	Implementation has an injected reference Target of injected reference has not been initialized Implementation invokes a method on the reference
Predicate	Implementation receives a ServiceUnavailableException
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "exception" "reference" "uninitialized"

118



Assertion ID	JCA-TA-4013
Source	[JCA40013]
Target	Java component implementation in initializing state
Prerequisites	Implementation has 1 or more service methods
Predicate	Service methods are not invoked
Prescription Level	mandatory
Tags	"lifecycle" "injecting" "service" "invocation"

119


Assertion ID	JCA-TA-4014
Source	[JCA40014]
Target	Java component implementation in initializing state
Prerequisites	Invocation of method annotated with @Init completes successfully

Predicate	Component implementation transitions to the Running state
Prescription Level	mandatory
Tags	"lifecycle" "initializing" "transition" "running"

120

Assertion ID	JCA-TA-4015
Source	[JCA40015]
Target	Java component implementation in initializing state
Prerequisites	Exception is thrown by the component implementation
Predicate	Component implementation transitions to the Destroying state
Prescription Level	mandatory
Tags	"lifecycle" "initializing" "exception" "transition" "destroying"

121

Assertion ID	JCA-TA-4016
Source	[JCA40010] 
Target	Java component implementation in running state
Prerequisites	Component has one or more service Client component invokes one of the service operations
Predicate	Implementation service method corresponding to the operation is invoked
Prescription Level	mandatory
Tags	"lifecycle" "running" "service" "invocation" "method"

122

123

Assertion ID	JCA-TA-4017
Source	[JCA40017]
Target	Java component implementation in running state
Prerequisites	Implementation scope ends
Predicate	Component implementation transitions to the Destroying state
Prescription Level	mandatory
Tags	"lifecycle" "running" "scope ends" "transition" "destroying"

124



Assertion ID	JCA-TA-4018
Source	[JCA40018]
Target	Java component implementation
Prerequisites	Implementation transitions to Destroying state
Predicate	Implementation method annotated with @Destroy is invoked

Prescription Level	mandatory
Tags	"lifecycle" "destroying" "@Destroy" "transition"

125

Assertion ID	JCA-TA-4019
Source	[JCA40019]
Target	Java component implementation in destroying state
Prerequisites	Implementation has an injected reference Implementation invokes a method on the reference Target of reference has already been destroyed
Predicate	Implementation receives an InvalidServiceException
Prescription Level	mandatory
Tags	"lifecycle" "destroying" "exception" "reference"

126

Assertion ID	JCA-TA-4020
Source	[JCA40020]
Target	Java component implementation in destroying state
Prerequisites	Implementation has one or more services Client component invokes service operations
Predicate	Implementation service methods are not invoked
Prescription Level	mandatory
Tags	"lifecycle" "destroying" "service" "method" "invocation"

127

128

Assertion ID	JCA-TA-4021
Source	[JCA40021]
Target	Java component implementation in Destroying state
Prerequisites	Method with @Destroy annotation completes
Predicate	Component implementation transitions to terminated state
Prescription Level	mandatory
Tags	"lifecycle" "destroying" "@Destroy" "transition" "terminated"

129

Assertion ID	JCA-TA-4022
Source	[JCA40022]
Target	Java component implementation in Destroying state
Prerequisites	Method with @Destroy annotation throws an exception

Predicate	Component implementation transitions to the terminated state
Prescription Level	mandatory
Tags	"lifecycle" "destroying" "exception" "transition" "terminated" "@Destroy"

130

Assertion ID	JCA-TA-4023
Source	[JCA40023]
Target	Java component implementation is in Terminated state
Prerequisites	Component implementation has service Client component invokes operations on a service
Predicate	Implementation service methods are not invoked
Prescription Level	mandatory
Tags	"lifecycle" "terminated" "service" "invocation"

131

132

2.4 Section 5

133

134

Assertion ID	JCA-TA-5001
Source	[JCA50001]
Target	
Prerequisites	
Predicate	
Prescription Level	
Tags	

135

136

137

2.5 Section 8

138

139

Assertion ID	JCA-TA-8001
Source	[JCA70001]
Target	Declaration of a Java annotation that corresponds to an SCA intent
Prerequisites	
Predicate	Declaration includes the @Intent annotation

Prescription Level	mandatory
Tags	"annotation" "declaration" "@Intent" "intent"

140

141

Assertion ID	JCA-TA-8002
Source	[JCA70002]
Target	Method of Java implementation class
Prerequisites	Method is annotated with an Intent annotation
Predicate	Method is a setter method for an SCA reference, either through explicit marking with @Reference or through introspection as defined in component implementation specification
Prescription Level	mandatory
Tags	"annotation" "method" "intent"

142

143

Assertion ID	JCA-TA-8003
Source	[JCA70002]
Target	Field of a Java implementation class
Prerequisites	Field is annotated with an Intent annotation
Predicate	Field is an SCA reference field, either through explicit marking with @Reference or through introspection as defined in the component implementation specification
Prescription Level	mandatory
Tags	"annotation" "field" "intent"

144

Assertion ID	JCA-TA-8004
Source	[JCA70002]
Target	Constructor parameter of a Java implementation class
Prerequisites	Constructor parameter is annotated with an Intent annotation
Predicate	Constructor parameter is marked as an SCA reference using the @Reference annotation
Prescription Level	mandatory
Tags	"annotation" "constructor" "parameter" "intent"

145

Assertion ID	JCA-TA-8005
Source	[JCA70003]
Target	Java element annotated with multiple intent annotations

Prerequisites	
Predicate	The set of intents which apply to the Java element are the combination of all the separate intents declared by the annotations, following the combination rules defined in the SCA Policy Framework spec
Prescription Level	mandatory
Tags	"annotation" "multiple" "intent" "combination"

146

147

Assertion ID	JCA-TA-8006
Source	[JCA70004]
Target	Method of a Java interface
Prerequisites	One set of Intent annotations are specified on the Java interface A second set of Intent annotations are specified on the method
Predicate	The set of intents which apply to the method are the combination of the set of intents on the interface with the set of intents on the method, following the merging rules for a structural hierarchy defined in the SCA Policy Framework specification
Prescription Level	mandatory
Tags	"annotation" "multiple" "intent" "combination" "interface" "method"

148

149

Assertion ID	JCA-TA-8007
Source	[JCA70005]
Target	Method of Java implementation class
Prerequisites	Method is annotated with a PolicySets annotation
Predicate	Method is a setter method for an SCA reference, either through explicit marking with @Reference or through introspection as defined in component implementation specification
Prescription Level	mandatory
Tags	"annotation" "reference" "@PolicySets" "method"

150

151

Assertion ID	JCA-TA-8008
Source	[JCA70005]
Target	Field of a Java implementation class
Prerequisites	Field is annotated with a PolicySets annotation
Predicate	Field is an SCA reference field, either through explicit marking with @Reference or through introspection as defined in the component implementation specification

Prescription Level	mandatory
Tags	"annotation" "reference" "@PolicySets" "field"

152

153

Assertion ID	JCA-TA-8009
Source	[JCA70005]
Target	Constructor parameter of a Java implementation class
Prerequisites	Constructor parameter is annotated with a PolicySets annotation
Predicate	Constructor parameter is marked as an SCA reference using the @Reference annotation
Prescription Level	mandatory
Tags	"annotation" "constructor" "@PolicySets" "reference"

154

155

Assertion ID	JCA-TA-8010
Source	[JCA70006]
Target	Method of a Java interface
Prerequisites	Java interface is annotated with @PolicySets annotation Method is annotated with a second @PolicySets annotation
Predicate	The set of intents which apply to the method is the union of the policy sets specified by the two @PolicySets annotations.
Prescription Level	mandatory
Tags	"annotation" "@PolicySets" "interface" "method" "combination"

156

2.6 Section 9

157

158

Assertion ID	JCA-TA-9001
Source	[JCA80001]
Target	invocation of getService(...) method of ComponentContext API
Prerequisites	referenceName parameter of the method identifies a reference of multiplicity 0..n
Predicate	getService(...) method throws an IllegalArgumentException
Prescription Level	mandatory
Tags	"ComponentContext" "getService" "0..n" "IllegalArgumentException"

159

160

Assertion ID	JCA-TA-9002
Source	[JCA80002]
Target	invocation of <code>getRequestContext(...)</code> method of ComponentContext API
Prerequisites	invocation takes place during the execution of a Java business method of a service operation invocation take place on the same Java thread use by the SCA runtime to invoke the business method
Predicate	invocation returns a non-null value for the RequestContext
Prescription Level	mandatory
Tags	"ComponentContext" "getRequestContext" "business method" "service"

161

Assertion ID	JCA-TA-9003
Source	[JCA80003]
Target	invocation of <code>getServiceReference()</code> method of RequestContext API
Prerequisites	invocation takes place during the execution of a Java business method of a service operation
Predicate	<code>getServiceReference()</code> method returns a ServiceReference object that represents the service that was invoked
Prescription Level	mandatory
Tags	"RequestContext" "service operation" "getServiceReference"

162

Assertion ID	JCA-TA-9004
Source	[JCA80003]
Target	invocation of <code>getServiceReference()</code> method of RequestContext API
Prerequisites	invocation takes place during the execution of a Java business method of a callback operation
Predicate	<code>getServiceReference()</code> method returns a ServiceReference object that represents the callback that was invoked
Prescription Level	mandatory
Tags	"RequestContext" "callback operation" "getServiceReference"

163

164

2.7 Section 10

165

Assertion ID	JCA-TA-10001
Source	[JCA90001]
Target	Component implementation contains an SCA annotation that is improperly used

Prerequisites	
Predicate	The component which uses the implementation does not run
Prescription Level	mandatory
Tags	"implementation" "invalid annotation"

166

Assertion ID	JCA-TA-10002
Source	[JCA90002]
Target	Component implementation class
Prerequisites	Class has static method Class has an SCA annotation
Predicate	Class does not have SCA annotation applied to static method
Prescription Level	mandatory
Tags	"SCA annotation" "static method"



167

168

Assertion ID	JCA-TA-10003
Source	[JCA90002]
Target	Component implementation class
Prerequisites	Class has static field Class has an SCA annotation
Predicate	Class does not have SCA annotation applied to static field
Prescription Level	mandatory
Tags	"SCA annotation" "static field"



169

170

Assertion ID	JCA-TA-10004		
Source	[JCA90052]		
Target	Java class containing @AllowsPassByReference annotation		
Prerequisites			
Predicate	@AllowsPassByReference annotation is attached to: <ul style="list-style-type: none"> - the Java class itself - or a method of a remotable service offered by the class - or an SCA reference of the class, in which case it is attached to the reference in a place where a @Reference annotation is valid 		
Prescription	mandatory		

Level	
Tags	"@AllowsPassByReference" "Java class" "attachment"

171

Assertion ID	JCA-TA-10031 
Source	[JCA90046]
Target	@Callback annotation used to annotation a setter method or field of an implementation class for injection of a callback object 
Prerequisites	
Predicate	@Callback annotation does not specify any attributes
Prescription Level	mandatory
Tags	"annotation" "Callback" "implementation" "injection"


172

Assertion ID	JCA-TA-10005
Source	[JCA90003]
Target	Constructor of an implementation class
Prerequisites	Constructor has parameters Constructor is annotated with the @Constructor annotation
Predicate	Each parameter of the Constructor is annotated with either @Reference or with @Property
Prescription Level	mandatory
Tags	"annotation" "constructor" "parameters" "@Reference" "@Property"

173


Assertion ID	JCA-TA-10006
Source	[JCA90004]
Target	Method of Java implementation class annotated with @Destroy
Prerequisites	
Predicate	Method has void return type and no arguments
Prescription Level	mandatory
Tags	"annotation" "@Destroy" "return type" "arguments"

174

Assertion ID	JCA-TA-10007 
Source	[JCA90005]
Target	Java implementation class with a method annotated with @Destroy
Prerequisites	Scope of the implementation class ends
Predicate	Method annotated with @Destroy is invoked.
Prescription Level	mandatory

Tags	"annotation" "@Destroy" "invoke" "scope"
------	--


175

Assertion ID	JCA-TA-10008
Source	[JCA90007]
Target	Java implementation class annotated with @EagerInit
Prerequisites	Implementation class is Composite scoped Component using the implementation is started
Predicate	Java implementation class is started 
Prescription Level	mandatory
Tags	"annotation" "@EagerInit" "composite scope" "started"

176

Assertion ID	JCA-TA-10009
Source	[JCA90008]
Target	Method annotated with @Init in a Java implementation class
Prerequisites	
Predicate	Method has void return type and no arguments
Prescription Level	mandatory
Tags	"anotation" "method" "@Init" "return type" "arguments"

177

Assertion ID	JCA-TA-10010 
Source	[JCA0009]
Target	Method annotated with @Init
Prerequisites	Java class is started Property and Reference injection is complete
Predicate	Method is invoked
Prescription Level	mandatory
Tags	"annotation" "@Init" "started" "invoked"

178

Assertion ID	JCA-TA-10011
Source	[JCA90011]
Target	Java implementation class field annotated with @Property
Prerequisites	
Predicate	Class field is not declared as final
Prescription Level	mandatory
Tags	"annotation" "@Property" "field" "final"

179


Assertion ID	JCA-TA-10012
Source	[JCA90013]
Target	@Property annotation of a constructor parameter
Prerequisites	
Predicate	@Property annotation has a @name attribute present
Prescription Level	mandatory
Tags	"annotation" "@Property" "@name" "constructor" "parameter"

180

181

Assertion ID	JCA-TA-10013
Source	[JCA90014]
Target	@Property annotation of a constructor parameter
Prerequisites	
Predicate	@Property annotation has @required=true
Prescription Level	mandatory
Tags	"annotation" "@Property" "@required" "constructor" "parameter"

182

Assertion ID	JCA-TA-10014 
Source	[JCA90047]
Target	Inspected component type of Java implementation class
Prerequisites	@Property annotation is applied to a class field or to the input parameter of a setter method or of a constructor Type of the element to which @Property is applied is an array or is a type that implements or extends java.util.Collection
Predicate	Component type contains a <property/> element corresponding to the @Property annotation, with @many attribute set to "true"
Prescription Level	mandatory
Tags	"annotation" "@Property" "componentType" "@many" "array" "collection"

183

Assertion ID	JCA-TA-10015
Source	[JCA90047]
Target	Inspected component type of Java implementation class
Prerequisites	@Property annotation is applied to a class field or to the input parameter of a setter method or of a constructor Type of the element to which @Property is applied is not an array and is not a type that implements or extends java.util.Collection
Predicate	Component type contains a <property/> element corresponding to the

	@Property annotation, with @many attribute set to "false"
Prescription Level	mandatory
Tags	"annotation" "@Property" "componentType" "@many" "array" "collection"

184

Assertion ID	JCA-TA-10016
Source	[JCA90015]
Target	Definition of an Intent annotation
Prerequisites	The SCA intent has qualifiers
Predicate	The Intent annotation definition contains the @Qualifier annotation
Prescription Level	mandatory
Tags	"annotation" "defintion" "intent" "qualifiers" "@Qualifier"

185

Assertion ID	JCA-TA-10017
Source	[JCA90016]
Target	Java implementation class field annotated with @Reference
Prerequisites	
Predicate	Class field is not declared as final
Prescription Level	mandatory
Tags	"annotation" "@Reference" "field" "final"


186

Assertion ID	JCA-TA-10018
Source	[JCA90018]
Target	@Reference annotation of a constructor method parameter
Prerequisites	
Predicate	@Reference annotation has a @name attribute present
Prescription Level	mandatory
Tags	"annotation" "constructor" "parameter" "@Reference" "@name"

187

Assertion ID	JCA-TA-10019
Source	[JCA0019]
Target	@Reference annotation of a constructor method parameter
Prerequisites	
Predicate	@Reference annotation has its @required attribute with the value "true"
Prescription Level	mandatory
Tags	"annotation" "constructor" "parameter" "@Reference" "@required"

188

Assertion ID	JCA-TA-10020
Source	[JCA90020] 
Target	Inspected component type of Java implementation class
Prerequisites	<p>@Reference annotation is applied to a class field or to the input parameter of a setter method or of a constructor</p> <p>Type of the element to which @Reference is applied is not an array and is not a type that implements or extends java.util.Collection</p> <p>@Reference @required attribute has the value "false"</p>
Predicate	Component type contains a <reference/> element corresponding to the @Reference annotation, with @multiplicity attribute set to "0..1"
Prescription Level	mandatory
Tags	"annotation" "@Reference" "componentType" "@multiplicity" "array" "collection"

189

Assertion ID	JCA-TA-10021
Source	[JCA90020]
Target	Inspected component type of Java implementation class
Prerequisites	<p>@Reference annotation is applied to a class field or to the input parameter of a setter method or of a constructor</p> <p>Type of the element to which @Reference is applied is not an array and is not a type that implements or extends java.util.Collection</p> <p>@Reference @required attribute has the value "true"</p>
Predicate	Component type contains a <reference/> element corresponding to the @Reference annotation, with @multiplicity attribute set to "1..1"
Prescription Level	mandatory
Tags	"annotation" "@Reference" "componentType" "@multiplicity" "array" "collection"

190

Assertion ID	JCA-TA-10022
Source	[JCA90021]
Target	Inspected component type of Java implementation class
Prerequisites	<p>@Reference annotation is applied to a class field or to the input parameter of a setter method or of a constructor</p> <p>Type of the element to which @Reference is applied is an array or is a type that implements or extends java.util.Collection</p> <p>@Reference @required attribute has the value "false"</p>
Predicate	Component type contains a <reference/> element corresponding to the @Reference annotation, with @multiplicity attribute set to "0..n"
Prescription	mandatory

Level	
Tags	"annotation" "@Reference" "componentType" "@multiplicity" "array" "collection"

191

Assertion ID	JCA-TA-10023
Source	[JCA90021]
Target	Introspected component type of Java implementation class
Prerequisites	@Reference annotation is applied to a class field or to the input parameter of a setter method or of a constructor Type of the element to which @Reference is applied is an array or is a type that implements or extends java.util.Collection @Reference @required attribute has the value "true"
Predicate	Component type contains a <reference/> element corresponding to the @Reference annotation, with @multiplicity attribute set to "1..n"
Prescription Level	mandatory
Tags	"annotation" "@Reference" "componentType" "@multiplicity" "array" "collection"

192

Assertion ID	JCA-TA-10024
Source	[JCA90022]
Target	Reference obtained by the Java implementation either via injection or via the ComponentContext getService(...) method
Prerequisites	Reference has multiplicity of 0..1 Component reference is unwired
Predicate	Reference is null
Prescription Level	mandatory
Tags	"reference" "unwired" "null" "0..1" "multiplicity"


193

Assertion ID	JCA-TA-10025
Source	[JCA90023]
Target	Reference obtained by the Java implementation either via injection or via the ComponentContext getService(...) method
Prerequisites	Reference has multiplicity of 0..n Component reference is unwired
Predicate	Reference is an empty array/collection
Prescription Level	mandatory
Tags	"reference" "unwired" "empty" "0..n" "multiplicity"

194

Assertion ID	JCA-TA-10026
Source	[JCA90024]
Target	Component reference is rewired during lifetime of an implementation instance
Prerequisites	
Predicate	Reference object is reinjected into the implementation instance
Prescription Level	optional
Tags	"re injection" "reference" "rewire" "implementation" "instance"

195

Assertion ID	JCA-TA-10027 
Source	[JCA90025]
Target	Implementation class which has a reference reinjected due to wiring changes
Prerequisites	
Predicate	Implementation is not STATELESS scope
Prescription Level	mandatory
Tags	"re injection" "reference" "implementation" "stateless" "scope"

196

Assertion ID	JCA-TA-10028
Source	[JCA90025]
Target	Implementation class which has a reference reinjected due to wiring changes
Prerequisites	
Predicate	Implementation does not use constructor based injection for that reference
Prescription Level	mandatory
Tags	"re injection" "reference" "implementation" "constructor"

197

Assertion ID	JCA-TA-10029
Source	[JCA90026]
Target	Reference object of an implementation instance
Prerequisites	Reference wiring changes so that reference target is changed Reference is not reinjected
Predicate	Reference object continues to work as if the reference target is not changed
Prescription Level	mandatory
Tags	"reference" "rewire" "re injection"

198

199

Assertion ID	JCA-TA-10030
Source	[JCA90027]
Target	Operation invocation on a reference of an implementation instance
Prerequisites	Target of the reference has been undeployed since the reference was injected
Predicate	Operation throws InvalidServiceException
Prescription Level	recommended
Tags	"reference" "target" "undeployed" "invocation" "InvalidServiceException"


200

Assertion ID	JCA-TA-10032
Source	[JCA90028]
Target	Operation invocation on a reference of an implementation instance
Prerequisites	Target of the reference has become unavailable for some reason
Predicate	Operation throws ServiceUnavailableException
Prescription Level	recommended
Tags	"reference" "target" "unavailable" "invocation" "ServiceUnavailableException"

201

Assertion ID	JCA-TA-10033
Source	[JCA90029]
Target	Operation invocation on a reference of an implementation instance
Prerequisites	Target of the reference has been changed since the reference was injected
Predicate	Operation either works or throws InvalidServiceException
Prescription Level	mandatory
Tags	"reference" "target" "changed" "invocation" "InvalidServiceException"

202

Assertion ID	JCA-TA-10034
Source	[JCA90030] 
Target	ServiceReference obtained from a reference object via the ComponentContext.cast() method
Prerequisites	Reference is reinjected
Predicate	ServiceReference object continues to work as if the reference target were not changed
Prescription Level	mandatory
Tags	"ServiceReference" "reference" "reinjection" "cast(...)"

203

Assertion ID	JCA-TA-10035
Source	[JCA90031]
Target	Operation invocation on ServiceReference object
Prerequisites	Target of the ServiceReference is undeployed
Predicate	Operation throws an InvalidServiceException
Prescription Level	recommended
Tags	"ServiceReference" "target" "undeployed" "InvalidServiceException" "invocation"

204

Assertion ID	JCA-TA-10036
Source	[JCA90032]
Target	Operation invocation on ServiceReference object
Prerequisites	Target of the ServiceReference becomes unavailable
Predicate	Operation throws a ServiceUnavailableException
Prescription Level	recommended
Tags	"ServiceReference" "target" "unavailable" "ServiceUnavailableException" "invocation"

205

Assertion ID	JCA-TA-10037
Source	[JCA90033]
Target	Operation invocation on a ServiceReference object
Prerequisites	Target of the ServiceReference has been changed since the ServiceReference was obtained
Predicate	Operation either works or throws InvalidServiceException
Prescription Level	mandatory
Tags	"ServiceReference" "target" "changed" "InvalidServiceException" "invocation"

206

Assertion ID	JCA-TA-10038
Source	[JCA90034]
Target	Reference obtained from getService() method of ComponentContext
Prerequisites	Domain configuration has changed since the component was started Configuration changes affect the reference
Predicate	Reference configuration reflects the Domain configuration at the time the getService() method was called
Prescription Level	mandatory
Tags	"reference" "getService()" "domain" "configuration" "changed"

207

Assertion ID	JCA-TA-10039
Source	[JCA90035]
Target	ServiceReference obtained from getServiceReference() method of ComponentContext
Prerequisites	Domain configuration has changed since the component was started Configuration changes affect the reference
Predicate	ServiceReference configuration reflects the Domain configuration at the time the getServiceReference() method was called
Prescription Level	mandatory
Tags	"ServiceReference" "getServiceReference()" "domain" "configuration" "changed"

208

Assertion ID	JCA-TA-10040
Source	[JCA90036]
Target	Reference obtained from getService() method of ComponentContext
Prerequisites	Target service of the component reference has changed since the component was started
Predicate	Reference target is the new target for the component reference
Prescription Level	recommended
Tags	"reference" "getService()" "target" "changed"

209

Assertion ID	JCA-TA-10041
Source	[JCA90036]
Target	ServiceReference obtained from getServiceReference() method of ComponentContext
Prerequisites	Target service of the component reference has changed since the component was started
Predicate	ServiceReference target is the new target for the reference
Prescription Level	recommended
Tags	"ServiceReference" "getServiceReference()" "target" "changed"

210

Assertion ID	JCA-TA-10042
Source	[JCA90037]
Target	Array or Collection object for a reference of multiplicity 0..n or 1..n
Prerequisites	Changes occur to the wiring of the reference or to the target(s) of the wiring Reference reinjection is not allowed
Predicate	Contents of the Array / Collection do not change

Prescription Level	mandatory
Tags	"reference" "multiplicity" "0..n" "1..n" "wiring" "target" "change" "injection"

211

212

Assertion ID	JCA-TA-10043
Source	[JCA90038]
Target	Reference with multiplicity 0..n or 1..n with a setter method for injection
Prerequisites	Changes occur to the wiring of the reference or to the target(s) of the wiring Reference was originally injected with an array or a collection object
Predicate	Setter method is invoked with an array / collection object containing the new targets of the component reference
Prescription Level	mandatory
Tags	"reference" "multiplicity" "0..n" "1..n" "wiring" "target" "change" "injection"

213

Assertion ID	JCA-TA-10044
Source	[JCA90039]
Target	Array or Collection object for a reference, where reinjection takes place
Prerequisites	Reference of multiplicity 0..n or 1..n Changes occur to the wiring of the reference or to the target(s) of the wiring
Predicate	Array or Collection object is not the same object that was originally injected for this reference
Prescription Level	mandatory
Tags	"reference" "multiplicity" "0..n" "1..n" "wiring" "target" "change" "injection"

214

Assertion ID	JCA-TA-10045
Source	[JCA90040]
Target	Interface of a service marked remotable by the @Remotable annotation
Prerequisites	
Predicate	The interface is translatable into a WSDL portType
Prescription Level	mandatory
Tags	"interface" "service" "remotable" "WSDL" "portType" "translatable"

215

Assertion ID	JCA-TA-10046
Source	[JCA90041]
Target	@Scope annotation
Prerequisites	

Predicate	Annotation is applied to a component implementation class
Prescription Level	mandatory
Tags	"annotation" "@Scope" "implementation class"

216

Assertion ID	JCA-TA-10047
Source	[JCA90042]
Target	Implementation class annotated with @Service annotation
Prerequisites	@Service annotation declares a set of services which implies a set of service interfaces
Predicate	Implementation class implements all of the methods of all the declared service interfaces
Prescription Level	mandatory
Tags	"annotation" "@Service" "methods" "interface" "implementation"

217

Assertion ID	JCA-TA-10048
Source	[JCA90048]
Target	@Service annotation
Prerequisites	@name attribute is declared
Predicate	@Service annotation declares a @value attribute
Prescription Level	mandatory
Tags	"annotation" "@Service" "@name" "@value"

218

Assertion ID	JCA-TA-10049
Source	[JCA90049]
Target	@Service annotation
Prerequisites	@names attribute is declared
Predicate	@Service annotation declares a @interfaces attribute
Prescription Level	mandatory
Tags	"annotation" "@Service" "@names" "@interfaces"

219

Assertion ID	JCA-TA-10050
Source	[JCA90050]
Target	@names attribute of @Service annotation
Prerequisites	
Predicate	Number of strings in the @names attribute array is the same as the number of element in the @interfaces attribute array
Prescription	mandatory

Level	
Tags	"annotation" "@Service" "@names" "@interfaces" "number"

220

Assertion ID	JCA-TA-10051
Source	[JCA90043]
Target	@Service annotation with @interfaces attribute declared
Prerequisites	
Predicate	@Service annotation does not declare an @value attribute
Prescription Level	mandatory
Tags	"annotation" "@Service" "@interfaces" "@value"

221

Assertion ID	JCA-TA-10052
Source	[JCA90043]
Target	@Service annotation with @value attribute declared
Prerequisites	
Predicate	@Service annotation does not declare an @interfaces attribute
Prescription Level	mandatory
Tags	"annotation" "@Service" "@interfaces" "@value"

222

Assertion ID	JCA-TA-10053
Source	[JCA90044]
Target	@Service annotation with single interface class object Void.class
Prerequisites	Void.class is either the default value for the @value attribute or the default content of the @interfaces array, or it is explicitly used as the value for the @value or @interfaces attributes
Predicate	@Service annotation is ignored - the implementation class is treated as if the annotation was not present at all
Prescription Level	mandatory
Tags	"annotation" "@Service" "@interfaces" "@value" "Void.class" "ignored"

223

Assertion ID	JCA-TA-10054
Source	[JCA90051]
Target	@Service annotation
Prerequisites	Annotation specifies one or more interface class other than Void.class for either its @value or @interfaces attributes
Predicate	@Service annotation does not specify Void.class for either its @value or its @interfaces attributes
Prescription	mandatory

Level	
Tags	"annotation" "@Service" "@interfaces" "@value" "Void.class"

224

Assertion ID	JCA-TA-10055
Source	[JCA90045]
Target	Implementation class declaring 2 or more services
Prerequisites	
Predicate	Each service has a distinct Java simple name
Prescription Level	mandatory
Tags	"annotation" "@Service" "service" "name"

225

226 2.8 Section 11

227

Assertion ID	JCA-TA-11001
Source	[JCA100001]
Target	Java interface which is mapped to WSDL by SCA runtime
Prerequisites	Java interface does not have a @WebService annotation
Predicate	Interface is mapped to WSDL as if it did have @WebService annotation
Prescription Level	mandatory
Tags	"interface" "Java" "WSDL" "mapping" "@WebService"

228

Assertion ID	JCA-TA-11002
Source	[JCA100002]
Target	Java interface containing @org.oasisopen.sca.annotation.OneWay annotation, mapped to WSDL by SCA runtime
Prerequisites	
Predicate	Interface is mapped to WSDL as if it contained a @javax.jws.OneWay annotation
Prescription Level	mandatory
Tags	"interface" "Java" "WSDL" "mapping" "@OneWay"

229

Assertion ID	JCA-TA-11003
Source	[JCA100003]
Target	WSDL interface mapped to a Java interface by SCA runtime
Prerequisites	
Predicate	Generated @WebService annotation in Java interface is taken to mean

	that the service interface is Remotable
Prescription Level	mandatory
Tags	"interface" "Java" "WSDL" "mapping" "@WebService" "remotable"


230

Assertion ID	JCA-TA-11004
Source	[JCA100004]
Target	Java interface used for service interface
Prerequisites	Interface contains JAXB 2.1 data types and annotations
Predicate	SCA runtime is able to map the interface to WSDL
Prescription Level	mandatory
Tags	"interface" "WSDL" "mapping" "JAXB" "2.1"

231

Assertion ID	JCA-TA-11005
Source	[JCA100005]
Target	Java interface used for service interface
Prerequisites	Interface contains SDO 2.1 data types and annotations
Predicate	SCA runtime is able to map the interface to WSDL
Prescription Level	optional
Tags	"interface" "WSDL" "mapping" "SDO" "2.1"

232


Assertion ID	JCA-TA-11006
Source	[JCA100006] 
Target	Java interface used to declare SCA service interface in <interface.java/> element
Prerequisites	
Predicate	Java interface does not contain additional client side asynchronous polling and callback methods defined by JAX-WS
Prescription Level	mandatory
Tags	"interface" "service" "asynchronous" "polling" "callback" "methods"

233

Assertion ID	JCA-TA-11007
Source	[JCA100007]
Target	Java interface used to declare SCA reference interface in a Java implementation class
Prerequisites	Java interface contains the additional client side asynchronous polling and callback methods defined by JAX-WS
Predicate	Java interface is treated as valid and is used

Prescription Level	optional
Tags	"interface" "reference" "asynchronous" "polling" "callback" "methods"

234

Assertion ID	JCA-TA-11008
Source	[JCA100008] 
Target	SCA reference interface in the component type of a Java implementation class
Prerequisites	Reference is declared in the implementation class using a Java interface which contains the additional client side asynchronous polling and callback methods defined by JAX-WS
Predicate	Reference interface in the component type does not contain the additional client side asynchronous polling and callback methods defined by JAX-WS
Prescription Level	mandatory
Tags	"interface" "reference" "componentType" "asynchronous" "polling" "callback" "methods"

235

Assertion ID	JCA-TA-11009
Source	[JCA100009]
Target	Java interface used to declare SCA reference interface in a Java implementation class
Prerequisites	Java interface contains the additional client side asynchronous polling and callback methods defined by JAX-WS
Predicate	The implementation class is able to use the client side asynchronous polling and callback methods, with the semantics as described in the JAX-WS 2.1 specification
Prescription Level	mandatory
Tags	"interface" "reference" "asynchronous" "polling" "callback" "methods"

236

237
238
239

3 Cross Mapping of Conformance Statements to Assertions

Conformance statement	Test Assertion
JCA20001	JCA-TA-2001
JCA20002	JCA-TA-2002
JCA20003	JCA-TA-2003
JCA20004	JCA-TA-2004
JCA20005	JCA-TA-2005
JCA20006	JCA-TA-2006
JCA20007	JCA-TA-2007
JCA20008	JCA-TA-2008
JCA20009	JCA-TA-2009
JCA20010	JCA-TA-2010 JCA-TA-2011

240
241

Conformance statement	Test Assertion
JCA30001	JCA-TA-3001
JCA30002	JCA-TA-3002
JCA30003	JCA-TA-3003
JCA30004	JCA-TA-3004
JCA30005	JCA-TA-3005
JCA30006	JCA-TA-3006
JCA30007	JCA-TA-3007
JCA30008	Covered by JCA-TA-3006 and JCA-TA-3007

242
243

Conformance statement	Test Assertion
JCA40001	JCA-TA-4001
JCA40002	JCA-TA-4002
JCA40003	JCA-TA-4003
JCA40004	JCA-TA-4004
JCA40005	JCA-TA-4005
JCA40006	JCA-TA-4006
JCA40007	JCA-TA-4007
JCA40008	JCA-TA-4008
JCA40009	JCA-TA-4009
JCA40010	JCA-TA-4010
JCA40011	JCA-TA-4011
JCA40012	JCA-TA-4012

Conformance statement	Test Assertion
JCA40013	JCA-TA-4013
JCA40014	JCA-TA-4014
JCA40015	JCA-TA-4015
JCA40016	JCA-TA-4016
JCA40017	JCA-TA-4017
JCA40018	JCA-TA-4018
JCA40019	JCA-TA-4019
JCA40020	JCA-TA-4020
JCA40021	JCA-TA-4021
JCA40022	JCA-TA-4022
JCA40023	JCA-TA-4023

244

Conformance statement	Test Assertion
JCA70001	JCA-TA-8001
JCA70002	JCA-TA-8002
	JCA-TA-8003
	JCA-TA-8004
JCA70003	JCA-TA-8005
JCA70004	JCA-TA-8006
JCA70005	JCA-TA-8007
	JCA-TA-8008
	JCA-TA-8009
JCA70006	JCA-TA-8010

245

Conformance statement	Test Assertion
JCA80001	JCA-TA-9001
JCA80002	JCA-TA-9002
JCA80003	JCA-TA-9003
	JCA-TA-9004

246

Conformance statement	Test Assertion
JCA90001	JCA-TA-10001
JCA90002	JCA-TA-10002
	JCA-TA-10003
JCA90003	JCA-TA-10005
JCA90004	JCA-TA-10006
JCA90005	JCA-TA-10007
JCA90007	JCA-TA-10008
JCA90008	JCA-TA-10009

Conformance statement	Test Assertion
JCA90009	JCA-TA-10010
JCA90011	JCA-TA-10011
JCA90013	JCA-TA-10012
JCA90014	JCA-TA-10013
JCA90015	JCA-TA-10016
JCA90016	JCA-TA-10017
JCA90018	JCA-TA-10018
JCA90019	JCA-TA-10019
JCA90020	JCA-TA-10020 JCA-TA-10021
JCA90021	JCA-TA-10022 JCA-TA-10023
JCA90022	JCA-TA-10024
JCA90023	JCA-TA-10025
JCA90024	JCA-TA-10026
JCA90025	JCA-TA-10027 JCA-TA-10028
JCA90026	JCA-TA-10029
JCA90027	JCA-TA-10030
JCA90028	JCA-TA-10032
JCA90029	JCA-TA-10033
JCA90030	JCA-TA-10034
JCA90031	JCA-TA-10035
JCA90032	JCA-TA-10036
JCA90033	JCA-TA-10037
JCA90034	JCA-TA-10038
JCA90035	JCA-TA-10039
JCA90036	JCA-TA-10040 JCA-TA-10041
JCA90037	JCA-TA-10042
JCA90038	JCA-TA-10043
JCA90039	JCA-TA-10044
JCA90040	JCA-TA-10045
JCA90041	JCA-TA-10046
JCA90042	JCA-TA-10047

Conformance statement	Test Assertion
JCA90043	JCA-TA-10051 JCA-TA-10052
JCA90044	JCA-TA-10053
JCA90045	JCA-TA-10055
JCA90046	JCA-TA-10031
JCA90047	JCA-TA-10014 JCA-TA-10015
JCA90048	JCA-TA-10048
JCA90049	JCA-TA-10049
JCA90050	JCA-TA-10050
JCA90051	JCA-TA-10054
JCA90052	JCA-TA-10004

247

248

Conformance statement	Test Assertion
JCA10001	JCA-TA-11001
JCA10002	JCA-TA-11002
JCA10003	JCA-TA-11003
JCA10004	JCA-TA-11004
JCA10005	JCA-TA-11005
JCA10006	JCA-TA-11006
JCA10007	JCA-TA-11007
JCA10008	JCA-TA-11008
JCA10009	JCA-TA-11009

249

250

4 Conformance

251

There are no conformance statements relating to the Test Assertions.

252

253

Appendix A. Acknowledgments

254

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

255

256

Participants:

257

Participant Name	Affiliation
------------------	-------------

258

259

260

261

Appendix B. Non-Normative Text

262

263

Appendix C. Revision History

264

265

Revision	Date	Editor	Changes Made
1	08/06/09	Mike Edwards	Initial version
2	09/06/09	Mike Edwards	Added TAs for Section 3
3	11/06/09	Mike Edwards	Added TAs for Section 4
4	14/06/09	Mike Edwards	Added TAs for Section 8
5	15/06/09	Mike Edwards	Added TAs for Section 9 Added TAs for Section 10 Added TAs for Section 11

266