Service Component Architecture Client and Implementation Model for C Test Cases Version 1.1

Committee Draft 02

14 October 2010

Specification URIs:
This Version:
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases-cd02.html
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases-cd02.doc
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases-cd02.pdf (Authoritative)

Previous Version:
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases-cd01.html
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases-cd01.doc
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases-cd01.pdf (Authoritative)

Latest Version:
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases.html
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases.doc
http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases.pdf (Authoritative)

Technical Committee:
OASIS Service Component Architecture / C and C++ (SCA-C-C++) TC

Chair:
Bryan Aupperle, IBM

Editors:
Bryan Aupperle, IBM
David Haney
Pete Robbins, IBM

Related work:

The Test Suite artifacts relating to this document can be found here:

- http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-testcases-cd02.zip

This document is related to:

- SCA Client and Implementation Model for C Specification Version 1.1

Declared XML Namespaces:
http://docs.oasis-open.org/ns/opencsa/scatests/200903
http://docs.oasis-open.org/ns/opencsa/scatests/2009032
http://test.sca.oasisopen.org/

Abstract:

This document defines the TestCases for the SCA C Client and Implementation Model specification.

The Test Cases represent a series of tests that an SCA implementation must pass in order to

claim conformance to the requirements of the SCA C Client and Implementation Model specification.

Status:

This document was last revised or approved by the Service Component Architecture / C and C++ 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-c-cpp/.

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-c-cpp/ipr.php).

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

Notices

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 name "OASIS" is a trademark 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

This document defines the Test Cases for the SCA Client and Implementation Model for C specification **[SCA-C]**.

The tests described in this document are related to the Test Assertions described in SCA Client and Implementation Model for C Test Assertions **[SCA-C-TA]**.

The test cases are structured in the same manner as the testcases for the SCA Assembly specification as described in the SCA Assembly testcases document **[SCA-TC]**.

1.1 Namespaces

The SCA C test case suite makes use of a set of XML namespaces.

1.1.1 SCA Artifact Namespaces

These apply to artifacts such as Composites

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

http://docs.oasis-open.org/ns/opencsa/scatests/2009032

1.1.2 WSDL Namespace

http://test.sca.oasisopen.org/

1.2 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]**.

1.3 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.

[SCA-C] OASIS Committee Draft 06, *Service Component Architecture Client and Implementation Model for C Specification Version 1.1*, October 2010. http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-spec-cd06.pdf

[SCA-C-TA] OASIS Committee Draft 02, *Service Component Architecture Client and Implementation Model for C Test Assertions Version 1.1*, October 2010. http://docs.oasis-open.org/opencsa/sca-c-cpp/sca-ccni-1.1-test-assertions-cd02.pdf

[SCA-ASM-TA] OASIS Committee Draft 03, *Test Assertions for the SCA Assembly Model Version 1.1 Specification*, August 2010. http://docs.oasis-open.org/opencsa/sca-assembly/sca-assembly-1.1-test-assertions-cd03.pdf

1.4 Non-Normative References

[SCA-TC] OASIS Committee Draft 03, *Testcases for the SCA Assembly Specification Version 1.1*, June 2010. http://docs.oasis-open.org/opencsa/sca-assembly/sca-assembly-1.1-testcases-cd02.pdf

2 Test Cases for Base Functionality

2.1 Section 2

Test Case ID	C_2001_TestCase
Test Assertion	C-TA-2001
Description	Tests that all functions of a service interface of a component are implemented by the component implementation.
Artifacts	C_2001_TestCase.java Test_C_2001.composite TestInvocation.h C_0002_Client.c C_0002_Client.componentType Service1Superset.h Service1SupersetImpl2.c Service1SupersetImpl2.componentType
Expected output	Negative test: “exception”
Expected output	Positive test: “C_2003 request service1 operation1 invoked service2 operation 1 invoked aservice2 operation 1 invoked a”

Test Case ID	C_2004_TestCase
Test Assertion	C-TA-2003
Description	Tests that lifecycle functions have no parameters and a void return type.
Artifacts	C_2004_TestCase.java Test_C_2004.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl14.c Service1Impl14.componentType
Expected output	Negative test: “exception”

Test Case ID	C_2005_TestCase

Test Assertion	C-TA-2007
Description	Tests that functions excluded from a service interface cannot be invoked.
Artifacts	C_2005_TestCase.java Test_C_2005.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service11.h Service1Impl15.c Service1Impl15.componentType Service11Impl.c Service11Impl.componentType
Expected output	Positive test: “C_2005 request Test service got an error during execution: Incorrect parameter value”

42

Test Case ID	C_2006_TestCase
Test Assertion	C-TA-2008
Description	Tests that functions excluded from a callback interface cannot be invoked.
Artifacts	C_2006_TestCase.java Test_C_2006.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Callback12Impl.c Service1Callback12Impl.componentType Service12.h Service12Impl.c Service12Impl.componentType
Expected output	Positive test: “C_2006 request service1 operation1 invoked Test service got an error during execution: Incorrect parameter value”

43

Test Case ID	C_2007_TestCase
Test Assertion	C-TA-2011
Description	Tests that each function explicitly identified in a c interface is uniquely named.

Artifacts	C_2007_TestCase.java Test_C_2007.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service11.h Service1Impl15.c Service1Impl15.componentType Service11Impl.c Service11Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_2008_TestCase
Test Assertion	C-TA-2012
Description	Tests that each function explicitly identified in a c callback interface is uniquely named.
Artifacts	C_2008_TestCase.java Test_C_2008.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Callback12Impl.c Service1Callback12Impl.componentType Service12.h Service12.Callbackh Service12Impl.c Service12Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_2009_TestCase
Test Assertion	C-TA-2017
Description	Tests that each function explicitly identified in a c implementation is uniquely named.
Artifacts	C_2009_TestCase.java Test_C_2009.composite

	TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl15.c Service1Impl15.componentType
Expected output	Negative test: “exception”

46

Test Case ID	C_2012_TestCase
Test Assertion	C-TA-2019
Description	Tests that where one component is a client of a service provided by a second component, both with C implementations and which both run in the same address space, but the service implementation methods are not marked "allows pass by reference" that invocations of the service use "pass by value" semantics.
Artifacts	C_2012_TestCase.java Test_C_2012.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service4.h Service1Impl7.c Service1Impl7.componentType Service4Impl.c Service4Impl.componentType
Expected output	Positive test: “C_2012 request service1 operation1 invoked service2 operation1 invoked request+1”

47

Test Case ID	C_2013_TestCase
Test Assertion	C-TA-2019
Description	Tests that where one component is a client of a service provided by a second component, both with C implementations and which both run in the same address space, with the service implementation methods are marked "allows pass by reference" but the client implementation is not marked "allows pass by reference" that invocations of the service use "pass by value" semantics.
Artifacts	C_2013_TestCase.java Test_C_2013.composite TestInvocation.h C_0001_Client.c

	C_0001_Client.componentType Service1.h Service4.h Service1Impl7.c Service1Impl7.componentType Service4Impl.c Service4Impl.componentType
Expected output	Positive test: “C_2013 request service1 operation1 invoked service2 operation1 invoked request+1”

Test Case ID	C_2014_TestCase
Test Assertion	C-TA-2006
Description	Tests that a runtime does no synchronization of component invocations.
Artifacts	C_2014_TestCase.java Test_C_2014.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl56.c Service1Impl56.componentType Service1Impl57.c Service1Impl57.componentType
Expected output	Positive test: “C_2014 request service1 operation1 invoked service2 operation1 invoked b”

2.2 Section 4

Test Case ID	C_4001_TestCase
Test Assertion	C-TA-4001
Description	Tests that the OneWay MEP is not rejected.
Artifacts	C_4001_TestCase.java Test_C_4001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service24OW.h Service24OW.wsdl

	Service1Impl29OW.c Service1Impl29OW.componentType Service24Impl.c Service24ImplOW.componentType
Expected output	Positive test: “C_4001 request service1 operation1 invoked service2 operation1 invoked”

2.3 Section 6

Test Case ID	C_6001_TestCase
Test Assertion	N/A – API Testing
Description	Tests that property values can be retrieved as integral types.
Artifacts	C_6001_TestCase.java Test_C_6001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl18.c Service1Impl18.componentType
Expected output	Positive test: “C_6001 request service1 operation1 invoked 1 22 333 4444 1A2B3C4D”

Test Case ID	C_6002_TestCase
Test Assertion	N/A – API Testing
Description	Tests that property values can be retrieved as boolean and floating-point types.
Artifacts	C_6002_TestCase.java Test_C_6002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl19.c Service1Impl19.componentType
Expected output	Positive test: “C_6002 request service1 operation1 invoked True 1.2 33.44”

Test Case ID	C_6003_TestCase
Test Assertion	N/A – API Testing

Description	Tests that property values can be retrieved as character types.
Artifacts	C_6003_TestCase.java Test_C_6003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl20.c Service1Impl20.componentType
Expected output	Positive test: "C_6003 request service1 operation1 invoked WideS C W abcde ABCDE"

Test Case ID	C_6004_TestCase
Test Assertion	N/A – API Testing
Description	Tests that property values can be retrieved as struct types.
Artifacts	C_6004_TestCase.java Test_C_6004.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl21.c Service1Impl21.componentType
Expected output	Positive test: "C_6004 request service1 operation1 invoked complex1 complex2"

Test Case ID	C_6005_TestCase
Test Assertion	N/A – API Testing
Description	Tests that truncated response data is detected and retrievable via SCAGetReplyMessage.
Artifacts	C_6005_TestCase.java Test_C_6005.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl22.c Service1Impl22.componentType

Expected output	Positive test: “C_6005 request service1 operation1 invoked operation1 invoked”

Test Case ID	C_6006_TestCase
Test Assertion	N/A – API Testing
Description	Tests that a component can access its own services using SCASelf.
Artifacts	C_6006_TestCase.java Test_C_6006.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service16.h Service1_16Impl.c Service1_16Impl.componentType
Expected output	Positive test: “C_6006 request service1 operation1 invoked operation2 invoked”

Test Case ID	C_6007_TestCase
Test Assertion	N/A – API Testing
Description	Tests that fault information is retrievable via SCAGetFaultMessage.
Artifacts	C_6007_TestCase.java Test_C_6007.composite TestInvocation.h C_0004_Client.c C_0004_Client.componentType Service17.h Service17Impl.c Service17Impl.componentType
Expected output	Positive test: “exception”

Test Case ID	C_6008_TestCase
Test Assertion	N/A – API Testing
Description	Tests that multi-valued property values can be retrieved.
Artifacts	C_6008_TestCase.java Test_C_6008.composite

	TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl32.c Service1Impl32.componentType
Expected output	Positive test: “C_6008 request service1 operation1 invoked 111 222”

Test Case ID	C_6015_TestCase
Test Assertion	N/A – API Testing
Description	Tests that all instances of a multiplicity 1..n reference are called.
Artifacts	C_6015_TestCase.java Test_C_6015.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl.c Service1Impl.componentType Service1Impl3.c Service1Impl3.componentType
Expected output	Positive test: “C_6015 request service1 operation1 invoked service2 operation1 invoked service2 operation1 invoked service4 operation1 invoked”

Test Case ID	C_6016_TestCase
Test Assertion	N/A – API Testing
Description	Tests that SCAInvokeAsync can be used with an interface marked requires="asyncInvocation".
Artifacts	C_6016_TestCase.java Test_C_6016.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service46.h Service1Impl53.c Service1Impl53.componentType

	Service46Impl.c Service46Impl.componentType
Expected output	Positive test: “C_6016 request service1 operation1 invoked operation2 invoked”

Test Case ID	C_6017_TestCase
Test Assertion	N/A – API Testing
Description	Tests that SCAInvokePoll can be used with an interface marked requires="asyncInvocation".
Artifacts	C_6017_TestCase.java Test_C_6017.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service46.h Service1Impl54.c Service1Impl54.componentType Service46Impl.c Service46Impl.componentType
Expected output	Positive test: “C_6017 request service1 operation1 invoked operation2 invoked”

Test Case ID	C_6018_TestCase
Test Assertion	N/A – API Testing
Description	Tests that SCAInvokePoll can be canceled with SCACancelAsync.
Artifacts	C_6018_TestCase.java Test_C_6018.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service46.h Service1Impl55.c Service1Impl55.componentType Service46Impl.c Service46Impl.componentType
Expected output	Positive test: “C_6018 request service1 operation1 invoked”

Test Case ID	C_6020_TestCase
Test Assertion	N/A – API Testing
Description	Tests that functions excluded from a callback interface cannot be invoked.
Artifacts	C_6020_TestCase.java Test_C_6020.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Callback47Impl.c Service1Callback47Impl.componentType Service47.h Service47Callback.h Service47Impl.c Service47Impl.componentType
Expected output	Positive test: "C_6020 request service1 operation1 invoked service2 operation1 invoked service1 callback1 invoked"

2.4 Section 7

Test Case ID	C_7001_TestCase
Test Assertion	C-TA-7001
Description	Tests <export.c/> elements have unique names within a domain.
Artifacts	C_7001_TestCase.java Test_C_7001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service4.h Service1Impl7.c Service1Impl7.componentType Service4Impl.c Service4Impl.componentType
Expected output	Negative test: "exception"

Test Case ID	C_7002_TestCase

Test Assertion	C-TA-7002
Description	Tests <import.c/> elements have unique names within a contribution.
Artifacts	C_7002_TestCase.java Test_C_7002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service4.h Service1Impl7.c Service1Impl7.componentType Service4Impl.c Service4Impl.componentType
Expected output	Negative test: “exception”

2.5 Section 8

Test Case ID	C_8001_TestCase
Test Assertion	C-TA-8001
Description	Tests that macros and typedefs are processed when testing interface compatibility.
Artifacts	C_8001_TestCase.java Test_C_8001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service15.h Service15Exp.h Service1Impl16.c Service1Impl16.componentType Service15Impl.c Service15Impl.componentType
Expected output	Positive test: “C_8001 request service1 operation1 invoked service1 operation 1”

Test Case ID	C_8002_TestCase
Test Assertion	C-TA-8002
Description	Tests that unions are not allowed in remotable interfaces.

Artifacts	C_8002_TestCase.java Test_C_8002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl17.c Service1Impl17.componentType Service16Impl.c Service16Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_8003_TestCase
Test Assertion	C-TA-8003
Description	Tests that a header file used to define a service interface contains function declarations.
Artifacts	C_8003_TestCase.java Test_C_8003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl15.c Service1Impl15.componentType Service13Impl.c Service13Impl.componentType
Expected output	Negative test: “exception”

2.6 Section 9

Test Case ID	C_10001_TestCase
Test Assertion	C-TA-10012, C-TA-10013, C-TA-10015, C-TA-10017, C-TA-10018, C-TA-10023, C-TA-F022, C-TA-F030, C-TA-F031, C-TA-F032
Description	Tests that the core default mapping of C to WSDL is correct.
Artifacts	C_10001_TestCase.java Test_C_10001.composite TestInvocation.h C_0001_Client.c

	C_0001_Client.componentType Service1.wsdl Service1.h Service1Impl.c Service1Impl.componentType
Expected output	Positive test: “C_10001 request service1 operation1 invoked”

Test Case ID	C_10003_TestCase
Test Assertion	C-TA-10016
Description	Tests the mapping of C unnamed parameters to WSDL.
Artifacts	C_10003_TestCase.java Test_C_10003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service18.h Service18.wsdl Service1Impl23.c Service1Impl23.componentType Service18Impl.c Service18Impl.componentType
Expected output	Positive test: “C_10003 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10004_TestCase
Test Assertion	C-TA-10018
Description	Tests the mapping of C parameter passing styles to WSDL.
Artifacts	C_10004_TestCase.java Test_C_10004.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service19.h Service19.wsdl Service1Impl24.c Service1Impl24.componentType

	Service19Impl.c Service19Impl.componentType
Expected output	Positive test: “C_10004 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10006_TestCase
Test Assertion	C-TA-10023
Description	Tests the mapping of C types to WSDL.
Artifacts	C_10006_TestCase.java Test_C_10006.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service21.h Service21.wsdl Service1Impl26.c Service1Impl26.componentType Service21Impl.c Service21Impl.componentType
Expected output	Positive test: “C_10006 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10007_TestCase
Test Assertion	C-TA-10025
Description	Tests the mapping of C structs to WSDL.
Artifacts	C_10007_TestCase.java Test_C_10007.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service22.h Service22.wsdl Service1Impl27.c Service1Impl27.componentType Service22Impl.c Service22Impl.componentType

Expected output	Positive test: “C_10007 request service1 operation1 invoked service2 operation1 invoked”

73

Test Case ID	C_10009_TestCase
Test Assertion	C-TA-10030
Description	Tests the core default mapping of an unannotated header to WSDL is correct.
Artifacts	C_10009_TestCase.java Test_C_10009.composite TestInvocation.h C_0001_ClientNA.c C_0001_ClientNA.componentType Service1.wsdl Service1NA.h Service1Impl.c Service1ImplNA.componentType
Expected output	Positive test: “C_10009 request service1 operation1 invoked”

74

Test Case ID	C_10010_TestCase
Test Assertion	C-TA-10002, C-TA-10006, C-TA-10029, C-TA-F001, C-TA-F002, C-TA-F005, C-TA-F006, C-TA-F007, C-TA-F010, C-TA-F011, C-TA-F018
Description	Tests that the core default mapping of WSDL to C is correct.
Artifacts	C_10010_TestCase.java Test_C_10010.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.wsdl Service1.h (system under test generated) Service1Impl.c Service1Impl.componentType
Expected output	Positive test: “C_10010 request service1 operation1 invoked”

75

Test Case ID	C_10011_TestCase
Test Assertion	C-TA-10004, C-TA-F012
Description	Tests that the mapping of WSDL unwrapped parameters to C is correct.

Artifacts	C_10011_TestCase.java Test_C_10011.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1U.wsdl Service1U.h (system under test generated) Service1ImplU.c Service1ImplU.componentType
Expected output	Positive test: “C_10011 request service1 operation1 invoked”

Test Case ID	C_10012_TestCase
Test Assertion	C-TA-10010, C-TA-F015
Description	Tests the mapping of WSDL faults to C.
Artifacts	C_10012_TestCase.java Test_C_10012.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h (previously system under test generated) Service1.wsdl Service31.h (system under test generated) Service31.wsdl Service1Impl39.c Service1Impl39.componentType Service31Impl.c Service31Impl.componentType
Expected output	Positive test: “C_10012 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10013_TestCase
Test Assertion	C-TA-10008
Description	Tests the default mapping of WSDL operation parameters to C parameter styles.
Artifacts	C_10013_TestCase.java Test_C_10013.composite TestInvocation.h C_0001_Client.c

	C_0001_Client.componentType Service1.h (previously system under test generated) Service1.wsdl Service19.h (system under test generated) Service19.wsdl Service1Impl24.c Service1Impl24.componentType Service19Impl.c Service19Impl.componentType
Expected output	Positive test: “C_10013 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10014_TestCase
Test Assertion	C-TA-10022, C-TA-10024
Description	Tests the default mapping of WSDL types to C.
Artifacts	C_10014_TestCase.java Test_C_10014.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h (previously system under test generated) Service1.wsdl Service32.h (system under test generated) Service32.wsdl Service1Impl40.c Service1Impl40.componentType Service32Impl.c Service32Impl.componentType
Expected output	Positive test: “C_10014 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10015_TestCase
Test Assertion	C-TA-10027
Description	Tests the default mapping of WSDL complex types to C.
Artifacts	C_10015_TestCase.java Test_C_10015.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType

	Service1.h (previously system under test generated) Service1.wsdl Service33.h (system under test generated) Service33.wsdl Service1Impl41.c Service1Impl41.componentType Service33Impl.c Service33Impl.componentType
Expected output	Positive test: “C_10015 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10020_TestCase
Test Assertion	C-TA-10001, C-TA-10029
Description	Tests the mapping of multiple portTypes in a WSDL definition.
Artifacts	C_10020_TestCase.java Test_C_10020.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h (previously system under test generated) Service1.wsdl Service34.h (system under test generated) Service35.h (system under test generated) Service34_35.wsdl Service1Impl42.c Service1Impl42.componentType Service3435Impl.c Service3435Impl.componentType (system under test generated)
Expected output	Positive test: “C_10020 request service1 operation1 invoked service2 operation1 invoked”

2.7 Section 10

Test Case ID	C_11001_TestCase
Test Assertion	C-TA-11001
Description	Tests that a composite containing incorrect interface.c elements is rejected.
Artifacts	C_11001_TestCase.java Test_C_11001.composite TestInvocation.h C_0001_Client.c

	C_0001_Client.componentType Service1.h Service11.h Service1Impl15.c Service1Impl15.componentType Service11Impl.c Service11Impl.componentType
Expected output	Negative test: “exception”

82

Test Case ID	C_11002_TestCase
Test Assertion	C-TA-11002
Description	Tests that a componentType containing incorrect interface.c elements is rejected.
Artifacts	C_11002_TestCase.java Test_C_11002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service11.h Service1Impl15.c Service1Impl15.componentType Service11Impl.c Service11Impl.componentType
Expected output	Negative test: “exception”

83

Test Case ID	C_11003_TestCase
Test Assertion	C-TA-11001
Description	Tests that a composite containing incorrect implementation.c attributes is rejected.
Artifacts	C_11003_TestCase.java Test_C_11003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl11.c Service1Impl11.componentType Service1Impl12.c

	Service1Impl12.componentType
Expected output	Negative test: “exception”

Test Case ID	C_11004_TestCase
Test Assertion	C-TA-11001
Description	Tests that a composite containing incorrect implementation.c attribute values is rejected.
Artifacts	C_11004_TestCase.java Test_C_11004.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl11.c Service1Impl11.componentType Service1Impl12.c Service1Impl12.componentType
Expected output	Negative test: “exception”

Test Case ID	C_11005_TestCase
Test Assertion	C-TA-11004
Description	Tests that a contribution containing incorrect import.c attributes is rejected.
Artifacts	C_11005_TestCase.java
Expected output	Negative test: “exception”

Test Case ID	C_11006_TestCase
Test Assertion	C-TA-11005
Description	Tests that a WSDL containing incorrect wsdl extension elements is rejected.
Artifacts	C_11006_TestCase.java Test_C_11006.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h

	Service18.h Service18.wsdl Service1Impl23.c Service1Impl23.componentType Service18Impl.c Service18Impl.componentType
Expected output	Negative test: “exception”

2.8 Appendix F

Test Case ID	C_F001_TestCase
Test Assertion	C-TA-F023
Description	Tests the WSDL mapping of C names containing "_x" and starting with "xml".
Artifacts	C_F001_TestCase.java Test_C_F001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service23.h Service23.wsdl Service1Impl28.c Service1Impl28.componentType Service23Impl.c Service23Impl.componentType
Expected output	Positive test: “C_F001 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_F002_TestCase
Test Assertion	C-TA-F025
Description	Tests the WSDL mapping of C functions with only in parameters and a void return type.
Artifacts	C_F002_TestCase.java Test_C_F002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service24.h

	Service24.wsdl Service1Impl29.c Service1Impl29.componentType Service24Impl.c Service24Impl.componentType
Expected output	Positive test: “C_F002 request service1 operation1 invoked service2 operation1 invoked”

89

Test Case ID	C_F003_TestCase
Test Assertion	C-TA-F026
Description	Tests that oneWay functions with a non-void return type are detected.
Artifacts	C_F003_TestCase.java Test_C_F003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service23.h Service23.wsdl Service1Impl28.c Service1Impl28.componentType Service23Impl.c Service23ImplOW.componentType
Expected output	Negative test: “exception”

90

Test Case ID	C_F005_TestCase
Test Assertion	C-TA-F008
Description	Tests the mapping or OneWay WSDL operations to C.
Artifacts	C_F005_TestCase.java Test_C_F005.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h (previously system under test generated) Service24OW.h (system under test generated) Service24OW.wsdl Service1Impl29OW.c Service1Impl29OW.componentType

	Service24Impl.c Service24Impl.componentType
Expected output	Positive test: “C_F005 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_F006_TestCase
Test Assertion	C-TA-F014
Description	Tests that parameter name clashes are detected.
Artifacts	C_F006_TestCase.java Test_C_F006.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service20.h (system under test generated – should fail) Service20.wsdl Service1Impl25.c Service1Impl25.componentType Service20Impl.c Service20Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_F007_TestCase
Test Assertion	C-TA-F016
Description	Tests the mapping of equivalent WSDL faults to C.
Artifacts	C_F007_TestCase.java Test_C_F007.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h (previously system under test generated) Service1.wsdl Service32F.h (system under test generated) Service32F.wsdl Service1Impl40.c Service1Impl40F.componentType Service32Impl.c Service32ImplF.componentType

Expected output	Positive test: “C_F007 request service1 operation1 invoked service2 operation1 invoked”

93

Test Case ID	C_F008_TestCase
Test Assertion	C-TA-F009, C-TA-F019
Description	Tests the mapping of a SOAP binding to C.
Artifacts	C_F008_TestCase.java Test_C_F008.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h (previously system under test generated) Service1.wsdl Service36.h (system under test generated) Service36.wsdl Service1Impl43.c Service1Impl43.componentType Service36Impl.c Service36Impl.componentType
Expected output	Positive test: “C_F008 request service1 operation1 invoked service2 operation1 invoked”

3 Test Cases for Program-Based Component Implementation Support

The test cases in this section apply to the optional program-based component implementation support and are only required if an implementation supports program-based component implementations.

3.1 Section 2

Test Case ID	C_2010_TestCase
Test Assertion	C-TA-2009
Description	Tests that operations excluded from a service interface defined by message formats cannot be invoked.
Artifacts	C_2010_TestCase.java Test_C_2010.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1ImplP.c Service1ImplP.componentType
Expected output	Positive test: “C_2009 request Test service got an error during execution: Incorrect parameter value”

Test Case ID	C_2011_TestCase
Test Assertion	C-TA-2010
Description	Tests that operations excluded from a callback interface defined by message formats cannot be invoked.
Artifacts	C_2011_TestCase.java Test_C_2011.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1Callback5ImplP.c Service1Callback5ImplP.componentType Service5P.h Service5CallbackP.h Service5ImplP.c

	Service5ImplP.componentType
Expected output	Positive test: “C_2006 request service1 operation1 invoked Test service got an error during execution: Incorrect parameter value”

3.2 Section 6

Test Case ID	C_6009_TestCase
Test Assertion	C-TA-6003, C-TA-10019
Description	Tests that operations in a service interface defined by message formats can be invoked.
Artifacts	C_6009_TestCase.java Test_C_6009.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1ImplP.c Service1ImplP.componentType
Expected output	Positive test: “C_6009 request service1 operation1 invoked”

Test Case ID	C_6010_TestCase
Test Assertion	C-TA-6003
Description	Tests that operations in a callback interface defined by message formats can be invoked.
Artifacts	C_6010_TestCase.java Test_C_6010.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1Callback5ImplP.c Service1Callback5ImplP.componentType Service5P.h Service5CallbackP.h Service5ImplP.c Service5ImplP.componentType
Expected output	Positive test: “C_6010 request service1 operation1 invoked service2 operation1 invoked service1

	callback1 invoked”

Test Case ID	C_6011_TestCase
Test Assertion	C-TA-6003
Description	Tests that specified input message formats are part of the service interface.
Artifacts	C_6011_TestCase.java Test_C_6011.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1Impl15P.c Service1Impl15P.componentType
Expected output	Positive test: “C_6011 request Test service got an error during execution: Other runtime detected error”

Test Case ID	C_6012_TestCase
Test Assertion	C-TA-6003
Description	Tests that specified input message formats are part of the callback interface.
Artifacts	C_6012_TestCase.java Test_C_6012.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1Callback12ImplP.c Service1Callback12ImplP.componentType Service12P.h Service12CallbackP.h Service12ImplP.c Service12ImplP.componentType
Expected output	Positive test: “C_6012 request service1 operation1 invoked Test service got an error during execution: Other runtime detected error”

Test Case ID	C_6013_TestCase
Test Assertion	C-TA-6003

Description	Tests that specified output message formats are part of the service interface.
Artifacts	C_6013_TestCase.java Test_C_6013.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1ImplP15.c Service1ImplP15.componentType
Expected output	Positive test: “C_6013 request Test service got an error during execution: Other runtime detected error”

105

Test Case ID	C_6014_TestCase
Test Assertion	C-TA-6003
Description	Tests that specified output message formats are part of the callback interface.
Artifacts	C_6014_TestCase.java Test_C_6014.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1Callback12ImplP.c Service1Callback12ImplP.componentType Service12P.h Service12CallbackP.h Service12ImplP.c Service12ImplP.componentType
Expected output	Positive test: “C_6014 request service1 operation1 invoked Test service got an error during execution: Other runtime detected error”

106

Test Case ID	C_6019_TestCase
Test Assertion	N/A – API Testing
Description	Tests that SCAInvokeAsync can be used with an program-based interface marked requires="asyncInvocation".
Artifacts	C_6019_TestCase.java Test_C_6019.composite TestInvocationP.h

	C_0003_Client.c C_0003_Client.componentType Service1P.h Service46P.h Service1Impl53P.c Service1Impl53P.componentType Service46ImplP.c Service46ImplP.componentType
Expected output	Positive test: "C_6019 request service1 operation1 invoked service2 operation1 invoked"

3.3 Section 8

Test Case ID	C_8004_TestCase
Test Assertion	C-TA-8003
Description	Tests that a header file used to define a service interface contains message format struct declarations.
Artifacts	C_8003_TestCase.java Test_C_8003.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1Impl15P.c Service1Impl15P.componentType Service13P.c Service13P.componentType
Expected output	Negavive test: "exception"

3.4 Section 9

Test Case ID	C_10002_TestCase
Test Assertion	C-TA-10014, C-TA-10020
Description	Tests the core default mapping of C to WSDL is correct for program component implementations.
Artifacts	C_10002_TestCase.java Test_C_10002.composite TestInvocation.h C_0003_Client.c C_0003_Client.componentType

	Service1.wsdl Service1P.h Service1ImplP.c Service1ImplP.componentType
Expected output	Positive test: “C_10002 request service1 operation1 invoked”

Test Case ID	C_10005_TestCase
Test Assertion	C-TA-10021
Description	Tests that parameters appearing in both a request and response message have the same type.
Artifacts	C_10005_TestCase.java Test_C_10005.composite TestInvocation.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service20P.h Service1Impl25P.c Service1Impl25P.componentType Service20P.c Service20P.componentType
Expected output	Negative test: “exception”

Test Case ID	C_10008_TestCase
Test Assertion	C-TA-10026
Description	Tests the mapping of C structs to WSDL.
Artifacts	C_10008_TestCase.java Test_C_10008.composite TestInvocation.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service1.wsdl Service22P.h Service22.wsdl Service1Impl27P.c Service1Impl27P.componentType

	Service22ImplP.c Service22ImpPl.componentType
Expected output	Positive test: “C_10008 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10016_TestCase
Test Assertion	C-TA-10003, C-TA-10007
Description	Tests that the core default mapping of WSDL to C is correct.
Artifacts	C_10016_TestCase.java Test_C_10016.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.wsdl Service1P.h (system under test generated) Service1ImplP.c Service1ImplP.componentType
Expected output	Positive test: “C_10016 request service1 operation1 invoked”

Test Case ID	C_10017_TestCase
Test Assertion	C-TA-10004, C-TA-F012
Description	Tests that the mapping of WSDL unwrapped parameters to C for program implementations.
Artifacts	C_10017_TestCase.java Test_C_10017.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1UP.wsdl Service1UP.h (system under test generated) Service1ImplUP.c Service1ImplUP.componentType
Expected output	Positive test: “C_10011 request service1 operation1 invoked”

Test Case ID	C_10018_TestCase
Test Assertion	C-TA-10009

Description	Tests the default mapping of WSDL operation parameters to C struct members.
Artifacts	C_10018_TestCase.java Test_C_10018.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h (previously system under test generated) Service1P.wsdl Service19Ph (system under test generated) Service19Pwsdl Service1Impl24Pc Service1Impl24PcomponentType Service19ImplP.c Service19ImplP.componentType
Expected output	Positive test: “C_10018 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_10019_TestCase
Test Assertion	C-TA-10028
Description	Tests the default mapping of WSDL complex types to C message structs.
Artifacts	C_10019_TestCase.java Test_C_10019.composite TestInvocation.h C_0003_Client.c C_0003_Client.componentType Service1P.h (previously system under test generated) Service1P.wsdl Service33P.h (system under test generated) Service33P.wsdl Service1Impl41P.c Service1Impl41P.componentType Service33ImplP.c Service33ImplP.componentType
Expected output	Positive test: “C_10019 request service1 operation1 invoked service2 operation1 invoked”

4 Test Cases for Proxy Function Support

The test cases in this section apply to the optional proxy function support and are only required if an implementation supports proxy functions.

4.1 Section 3

Test Case ID	C_3001_TestCase
Test Assertion	C-TA-3001
Description	Tests that operations in a service interface can be invoked by proxy functions.
Artifacts	C_3001_TestCase.java Test_C_3001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl.c Service1Impl.componentType Service1Impl2X.c Service1Impl2X.componentType
Expected output	Positive test: “C_3001 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_3002_TestCase
Test Assertion	C-TA-3002
Description	Tests that operations in a callback interface can be invoked by proxy functions.
Artifacts	C_3002_TestCase.java Test_C_3002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Callback5ImplX.c Service1Callback5ImplX.componentType Service5.h Service5Callback.h Service5ImplX.c Service5ImplX.componentType
Expected output	Positive test:

	"C_3002 request service1 operation1 invoked service2 operation1 invoked service1 callback1 invoked"

120

Test Case ID	C_3003_TestCase
Test Assertion	N/A – API Testing
Description	Tests that SCA_<operation>Async can be used with an interface marked requires="asyncInvocation".
Artifacts	C_3003_TestCase.java Test_C_3003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service46.h Service1Impl53X.c Service1Impl53X.componentType Service46Impl.c Service46Impl.componentType
Expected output	Positive test: "C_3003 request service1 operation1 invoked operation2 invoked"

121

Test Case ID	C_3004_TestCase
Test Assertion	N/A – API Testing
Description	Tests that SCA_<operation>Poll can be used with an interface marked requires="asyncInvocation".
Artifacts	C_3004_TestCase.java Test_C_3004.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service46.h Service1Impl54X.c Service1Impl54X.componentType Service46Impl.c Service46Impl.componentType
Expected output	Positive test: "C_3004 request service1 operation1 invoked operation2 invoked"

122

Test Case ID	C_3005_TestCase
Test Assertion	N/A – API Testing
Description	Tests that operations in a callback interface can be asynchronously invoked by proxy functions.
Artifacts	C_3005_TestCase.java Test_C_3005.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Callback47Impl.c Service1Callback47Impl.componentType Service47.h Service47Callback.h Service47ImplX.c Service47ImplX.componentType
Expected output	Positive test: “C_3005 request service1 operation1 invoked service2 operation1 invoked service1 callback1 invoked”

5 Test Cases for Annotation Support

The test cases in this section apply to the optional C++ SCA annotation support and are only required if an implementation supports these annotations.

5.1 Appendix A

Test Case ID	C_A001_TestCase
Test Assertion	C-TA-A001
Description	Tests that @Interface, @Service and @Property generate correct SCDL.
Artifacts	C_A001_TestCase.java Test_C_A001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service15Exp.h Service15A.h Service1Impl16.c Service1Impl16.componentType Service15ImplA.c Service15ImplA.componentType (system under test generated)
Expected output	Positive test: “C_A001 request service1 operation1 invoked service1 operation 1 invoked”

Test Case ID	C_A002_TestCase
Test Assertion	C-TA-A001
Description	Tests that @Remotable generates correct SCDL.
Artifacts	C_A002_TestCase.java Test_C_A002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl2.c Service1Impl2.componentType Service1Impl.c Service1Impl.componentType (system under test generated)
Expected output	Positive test:

	“C_A002 request service1 operation1 invoked service2 operation 1 invoked”

Test Case ID	C_A003_TestCase
Test Assertion	C-TA-A001
Description	Tests that @Callback generates correct SCDL.
Artifacts	C_A003_TestCase.java Test_C_A003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Callback5Impl.c Service1Callback5Impl.componentType Service5.h Service5Callback.h Service5Impl.c Service5Impl.componentType (system under test generated)
Expected output	Positive test: “C_A003 request service1 operation1 invoked service2 operation1 invoked service1 callback1 invoked”

Test Case ID	C_A004_TestCase
Test Assertion	C-TA-A001
Description	Tests that @OneWay generates correct SCDL.
Artifacts	C_A004_TestCase.java Test_C_A004.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service24OW.h Service24OW.wsdl Service1Impl29OW.c Service1Impl29OW.componentType Service24Impl.c Service24ImplOW.componentType (system under test generated)
Expected output	Positive test: “C_A004 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_A005_TestCase
Test Assertion	C-TA-A001
Description	Tests that @Reference generates correct SCDL.
Artifacts	C_A005_TestCase.java Test_C_A005.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl30.c Service1Impl30.componentType (system under test generated) Service1Impl.c Service1Impl.componentType (system under test generated)
Expected output	Positive test: "C_A005 request service1 operation1 invoked service2 operation 1 invoked"

131

Test Case ID	C_A007_TestCase
Test Assertion	C-TA-A001
Description	Tests that lifecycle annotations generate correct SCDL.
Artifacts	C_A007_TestCase.java Test_C_A007.composite TestCompositeC2.composite (system under test generated) • Contains 1 component implemented by Service1Impl31 • Provides 1 service (Service1) which promotes the Service1 of the component • Has one property (ServiceName, type string) that sets ServiceName of the component TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl2.c Service1Impl2.componentType Service1Impl31.c Service1Impl31.componentType
Expected output	Positive test: "C_A007 request service1 operation1 invoked service2 operation 1 invoked a"

132

Test Case ID	C_A008_TestCase

Test Assertion	C-TA-A002
Description	Tests that annotations are ignored at runtime.
Artifacts	C_A008_TestCase.java Test_C_A008.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Impl2.c Service1Impl2.componentType Service5.h Service1Impl.c Service5ImplA.componentType
Expected output	Positive test: "C_A008 request service1 operation1 invoked service2 operation1 invoked"

Test Case ID	C_A009_TestCase
Test Assertion	C-TA-A003
Description	Tests that annotations in multiple comment blocks are processed correctly.
Artifacts	C_A009_TestCase.java Test_C_A009.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1Callback5Impl.c Service1Callback5Impl.componentType Service5.h Service5AS.h Service5Callback.h Service5ImplAS.c Service5ImplAS.componentType (system under test generated)
Expected output	Positive test: "C_A009 request service1 operation1 invoked service2 operation1 invoked service1 callback1 invoked"

Test Case ID	C_A010_TestCase
Test Assertion	C-TA-A001, C-TA-C002

Description	Tests that @Function generates correct SCDL.
Artifacts	C_ A010_TestCase.java Test_C_ A010.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service11A.h Service11W.wsdl Service1Impl15A.c Service1Impl15A.componentType (system under test generated) Service11ImplA.c Service11ImplA.componentType (system under test generated)
Expected output	Positive test: "C_ A010 request Test service got an error during execution: Incorrect parameter value"

135

Test Case ID	C_A011_TestCase
Test Assertion	C-TA-A005, C-TA-C005
Description	Tests that @WebFunction generates same SCDL as equivalent, but missing @Function.
Artifacts	C_ A011_TestCase.java Test_C_ A011.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service11W.h Service11W.wsdl Service1Impl15W.c Service1Impl15W.componentType (system under test generated) Service11ImplW.c Service11ImplW.componentType (system under test generated)
Expected output	Positive test: "C_ A011 request Test service got an error during execution: Incorrect parameter value"

136

Test Case ID	C_A014_TestCase
Test Assertion	C-TA-A008

Description	Tests that all properties with a given name in an implementation have the same type.
Artifacts	C_A014_TestCase.java Test_C_A014.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service44.h Service1Impl51.c Service1Impl51.componentType Service44Impl.c Service44Impl.componentType(system under test generated)
Expected output	Negative test: “exception”

Test Case ID	C_A015_TestCase
Test Assertion	C-TA-A004
Description	Tests that @WebService generates same SCDL as equivalent, but missing @Remotable.
Artifacts	C_A015_TestCase.java Test_C_A015.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service45.h Service1Impl52.c Service1Impl52.componentType Service45Impl.c Service45Impl.componentType (system under test generated)
Expected output	Positive test: “C_A015 request service1 operation1 invoked service2 operation 1 invoked”

5.2 Appendix B

Test Case ID	C_B001_TestCase
Test Assertion	C-TA-A001
Description	Tests that @Requires generates correct SCDL.
Artifacts	C_B001_TestCase.java

	Test_C_B001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1_Requires.h Service1Impl2.c Service1Impl2.componentType Service1Impl.c Service1Impl_requires.componentType (system under test generated)
Expected output	Positive test: “C_B001 request service1 operation1 invoked service2 operation 1”

Test Case ID	C_B002_TestCase
Test Assertion	C-TA-A001
Description	Tests that @<Intent> generates correct SCDL.
Artifacts	C_B002_TestCase.java Test_C_B002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1_Intent.h Service1Impl2.c Service1Impl2.componentType Service1Impl.c Service1Impl_Intent.componentType (system under test generated)
Expected output	Positive test: “C_B002 request service1 operation1 invoked service2 operation 1”

5.3 Appendix C

Test Case ID	C_C001_TestCase
Test Assertion	C-TA-C005, C-TA-F033
Description	Tests that @WebService generates the correct WSDL.
Artifacts	C_C001_TestCase.java Test_C_C001.composite TestInterface.h TestInvocation.wsdl C_0005_Client.c

	C_0005_Client.componentType Service1.h Service1Impl.c Service1Impl.componentType
Expected output	Positive test: “C_C001 request service1 operation1 invoked”

Test Case ID	C_C002_TestCase
Test Assertion	C-TA-C005
Description	Tests that @WebParam generates the correct WSDL.
Artifacts	C_C002_TestCase.java Test_C_C002.composite TestInterface.h TestInvocation.wsdl C_0005_Client.c C_0005_Client.componentType Service1.h Service25.h Service25.wsdl Service1Impl33.c Service1Impl33.componentType Service25Impl.c Service25Impl.componentType
Expected output	Positive test: “C_C002 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_C003_TestCase
Test Assertion	C-TA-C005
Description	Tests that @WebResult generates the correct WSDL.
Artifacts	C_C003_TestCase.java Test_C_C003.composite TestInterface.h TestInvocation.wsdl C_0005_Client.c C_0005_Client.componentType Service1.h Service26.h Service26.wsdl Service1Impl34.c

	Service1Impl34.componentType Service26Impl.c Service26Impl.componentType
Expected output	Positive test: “C_C003 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_C004_TestCase
Test Assertion	C-TA-C001
Description	Tests that @WebService results in a remotable SCDL interface definition.
Artifacts	C_C004_TestCase.java Test_C_C004.composite TestInterface.h TestInvocation.wsdl C_0006_Client.c C_0006_Client.componentType (system under test generated) Service1.h Service1Impl.c Service1Impl.componentType
Expected output	Positive test: “C_C004 request service1 operation1 invoked”

Test Case ID	C_C005_TestCase
Test Assertion	C-TA-C004
Description	Tests that a struct that is listed in a @WebThrows annotation has a @WebFault annotation.
Artifacts	C_C005_TestCase.java Test_C_C005.composite TestInterface1.h TestInvocation.wsdl C_0007_Client.c C_0007_Client.componentType Service1.h Service1Impl.c Service1Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_C006_TestCase

Test Assertion	C-TA-C006
Description	Tests that @WebParam simple type errors are detected.
Artifacts	C_C006_TestCase.java Test_C_C006.composite TestInterface1.h C_0001_Client.c C_0001_Client.componentType Service1.h Service27.h Service1Impl35.c Service1Impl35.componentType Service27Impl.c Service27Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_C007_TestCase
Test Assertion	C-TA-C010
Description	Tests that @WebParam complex type errors are detected.
Artifacts	C_C007_TestCase.java Test_C_C007.composite TestInterface1.h C_0001_Client.c C_0001_Client.componentType Service1.h Service28.h Service1Impl36.c Service1Impl36.componentType Service28Impl.c Service28Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_C008_TestCase
Test Assertion	C-TA-C007
Description	Tests that @WebResult simple type errors are detected.
Artifacts	C_C008_TestCase.java Test_C_C008.composite

	TestInterface1.h C_0001_Client.c C_0001_Client.componentType Service1.h Service29.h Service1Impl37.c Service1Impl37.componentType Service29Impl.c Service29Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_C009_TestCase
Test Assertion	C-TA-C009
Description	Tests that @WebParam matches a named parameter.
Artifacts	C_C009_TestCase.java Test_C_C009.composite TestInterface1.h C_0001_Client.c C_0001_Client.componentType Service1.h Service30.h Service1Impl38.c Service1Impl38.componentType Service30Impl.c Service30Impl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_C010_TestCase
Test Assertion	C-TA-C005
Description	Tests that the OneWay MEP maps correctly to WSDL.
Artifacts	C_C010_TestCase.java Test_C_C010.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service24OW.h

	Service24OW.wsdl Service1Impl29OW.c Service1Impl29OW.componentType Service24Impl.c Service24ImplOW.componentType
Expected output	Positive test: “C_C010 request service1 operation1 invoked service2 operation1 invoked”

5.4 Appendix F

Test Case ID	C_F004_TestCase
Test Assertion	C-TA-F026
Description	Tests that functions with a non-void return type but annotated OneWay are detected.
Artifacts	C_F003_TestCase.java Test_C_F003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service23.h Service23OW.h Service23.wsdl Service1Impl28.c Service1Impl28.componentType Service23Impl.c Service23ImplOW.componentType
Expected output	Negative test: “exception”

5.5 Assembly

Test Case ID	C_1001_TestCase
Test Assertion	ASM-TA-8005
Description	Tests that the callback interface of a @Remotable interface is also @Remotable.
Artifacts	C_1001_TestCase.java Test_C_1001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h

	Service6.h Service6Callback.h Service1Callback6Impl.c Service1Callback6Impl.componentType (system under test generated) Service6Impl.c Service6Impl.componentType (system under test generated)
Expected output	Negative test: “exception”

Test Case ID	C_1002_TestCase
Test Assertion	ASM-TA-8006
Description	Tests that the callback interface of a local interface is not @Remotable.
Artifacts	C_1002_TestCase.java Test_C_1002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service7.h Service7Callback.h Service1Callback7Impl.c Service1Callback7Impl.componentType (system under test generated) Service7Impl.c Service7Impl.componentType (system under test generated)
Expected output	Negative test: “exception”

6 Test Cases for Annotations and Program-Based Component Implementation Support

The test cases in this section apply to the combination of optional program-based component implementation support and C++ SCA annotation support and are only required if an implementation supports both program-based component implementations and C++ SCA annotations.

6.1 Appendix A

Test Case ID	C_A012_TestCase
Test Assertion	C-TA-A001, C-TA-C003
Description	Tests that @Operation generates correct SCDL.
Artifacts	C_ A012_TestCase.java Test_C_ A012.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service11AP.h Service11WP.wsdl Service1Impl15AP.c Service1Impl15AP.componentType (system under test generated) Service11ImplAP.c Service11ImplAP.componentType (system under test generated)
Expected output	Positive test: “C_ A012 request Test service got an error during execution: Incorrect parameter value”

Test Case ID	C_A013_TestCase
Test Assertion	C-TA-A006, C-TA-C005
Description	Tests that @WebOperation generates same SCDL as equivalent, but missing @Operation.
Artifacts	C_ A013_TestCase.java Test_C_ A013.composite TestInvocationP.h C_0003_Client.c C_0003_Client.componentType Service1P.h Service11WP.h

	Service11WP.wsdl Service1Impl15WP.c Service1Impl15WP.componentType (system under test generated) Service11ImplWP.c Service11ImplWP.componentType (system under test generated)
Expected output	Positive test: “C_ A013 request Test service got an error during execution: Incorrect parameter value”

7 Test Cases for WSDL Extension Support

The test cases in this section apply to the optional C++ WSDL annotation support and are only required if an implementation supports these annotations.

7.1 Appendix D

Test Case ID	C_D001_TestCase
Test Assertion	C-TA-D001, C-TA-F003
Description	Tests the mapping of a function WSDL extension.
Artifacts	C_D001_TestCase.java Test_C_D001.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service37.h (system under test generated) Service37.wsdl Service1Impl44.c Service1Impl44.componentType Service37Impl.c Service37Impl.componentType
Expected output	Positive test: “C_D001 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_D002_TestCase
Test Assertion	C-TA-D001, C-TA-F003
Description	Tests the mapping of a parameter WSDL extension.
Artifacts	C_D002_TestCase.java Test_C_D002.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service38.h (system under test generated) Service38.wsdl Service1Impl45.c

	Service1Impl45.componentType Service38Impl.c Service38Impl.componentType
Expected output	Positive test: "C_D002 request service1 operation1 invoked service2 operation1 invoked"

165

Test Case ID	C_D003_TestCase
Test Assertion	C-TA-D001, C-TA-F003
Description	Tests the mapping of a struct WSDL extension.
Artifacts	C_D003_TestCase.java Test_C_D003.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service39.h (system under test generated) Service39.wsdl Service1Impl46.c Service1Impl46.componentType Service39Impl.c Service39Impl.componentType
Expected output	Positive test: "C_D003 request service1 operation1 invoked service2 operation1 invoked"

166

Test Case ID	C_D004_TestCase
Test Assertion	C-TA-D001, C-TA-F003, C-TA-F013
Description	Tests the mapping of an enableWrapperStyle WSDL extension.
Artifacts	C_D004_TestCase.java Test_C_D004.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service40.h (system under test generated) Service40.wsdl Service1Impl47.c Service1Impl47.componentType

	Service40Impl.c Service40Impl.componentType
Expected output	Positive test: “C_D004 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_D005_TestCase
Test Assertion	C-TA-D001, C-TA-F003
Description	Tests the mapping of a prefix WSDL extension.
Artifacts	C_D005_TestCase.java Test_C_D005.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service41.h (system under test generated) Service41.wsdl Service1Impl48.c Service1Impl48.componentType Service41Impl.c Service41Impl.componentType
Expected output	Positive test: “C_D005 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_D006_TestCase
Test Assertion	C-TA-D005
Description	Tests that a second function WSDL extension in one binding is detected.
Artifacts	C_D006_TestCase.java Test_C_D006.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service37.h (previously system under test generated) Service37Bad.wsdl Service1Impl44.c Service1Impl44.componentType Service37Impl.c

	Service37mpl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_D007_TestCase
Test Assertion	C-TA-D002
Description	Tests the mapping of a parameter WSDL extension with a simple type.
Artifacts	C_D007_TestCase.java Test_C_D007.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service42.h (system under test generated) Service42.wsdl Service1Impl49.c Service1Impl49.componentType Service42Impl.c Service42Impl.componentType
Expected output	Positive test: “C_D007 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_D008_TestCase
Test Assertion	C-TA-D002
Description	Tests the mapping of a parameter WSDL extension with a complex type.
Artifacts	C_D008_TestCase.java Test_C_D008.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service43.h (system under test generated) Service43.wsdl Service1Impl50.c Service1Impl50.componentType Service43Impl.c Service43Impl.componentType

Expected output	Positive test: “C_D008 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_D009_TestCase
Test Assertion	C-TA-D006
Description	Tests that a second struct WSDL extension in one binding is detected.
Artifacts	C_D009_TestCase.java Test_C_D009.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service39.h (previously system under test generated) Service39Bad.wsdl Service1Impl46.c Service1Impl46.componentType Service39Impl.c Service39mpl.componentType
Expected output	Negative test: “exception”

Test Case ID	C_D010_TestCase
Test Assertion	C-TA-D004
Description	Tests that a second enableWrapperStyle WSDL extension in one binding is detected.
Artifacts	C_D010_TestCase.java Test_C_D010.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service40.h (previously system under test generated) Service40Bad.wsdl Service1Impl47.c Service1Impl47.componentType Service40Impl.c Service40mpl.componentType

Expected output	Negative test: “exception”

173

Test Case ID	C_D011_TestCase
Test Assertion	C-TA-D003
Description	Tests that a second prefix WSDL extension in one binding is detected.
Artifacts	C_D011_TestCase.java Test_C_D011.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service41.h (previously system under test generated) Service41Bad.wsdl Service1Impl48.c Service1Impl48.componentType Service41Impl.c Service41mpl.componentType
Expected output	Negative test: “exception”

174

Test Case ID	C_D012_TestCase
Test Assertion	C-TA-D007
Description	Tests the mapping of a JAX-WS WSDL extension.
Artifacts	C_D012_TestCase.java Test_C_D0012.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service37.h (previously system under test generated) Service37J.wsdl Service1Impl44.c Service1Impl44.componentType Service37Impl.c Service37Impl.componentType

Expected output	Positive test: “C_D012 request service1 operation1 invoked service2 operation1 invoked”

Test Case ID	C_D013_TestCase
Test Assertion	C-TA-D008
Description	Tests that a C WSDL extension overrides a JAX-WS WSDL extension.
Artifacts	C_D013_TestCase.java Test_C_D0013.composite TestInvocation.h C_0001_Client.c C_0001_Client.componentType Service1.h Service1.wsdl Service37.h (system under test generated) Service37JC.wsdl Service1Impl44.c Service1Impl44.componentType Service37Impl.c Service37Impl.componentType
Expected output	Positive test: “C_D013 request service1 operation1 invoked service2 operation1 invoked”

8 Cross Mapping of Test Assertions to Test Cases

Test Assertion	Test Cases
ASM-TA-8001	N/A for C
ASM-TA-8005	C_1001_TestCase
ASM-TA-8006	C_1002_TestCase

Test Assertion	Test Cases
C-TA-2001	C_2001_TestCase
C-TA-2003	C_2004_TestCase
C-TA-2006	C_2014_TestCase
C-TA-2007	C_2005_TestCase
C-TA-2008	C_2006_TestCase
C-TA-2009	C_2010_TestCase
C-TA-2010	C_2011_TestCase
C-TA-2011	C_2007_TestCase
C-TA-2012	C_2008_TestCase
C-TA-2017	C_2009_TestCase
C-TA-2018	Optional – Not Tested
C-TA-2019	C_2012_TestCase C_2013_TestCase

Test Assertion	Test Cases
C-TA-3001	C_3001_TestCase
C-TA-3002	C_3002_TestCase

Test Assertion	Test Cases
C-TA-4001	C_4001_TestCase

Test Assertion	Test Cases
C-TA-5001	Untestable

Test Assertion	Test Cases
C-TA-6002	Optional – Not tested Requires an SDO Implementation
C-TA-6003	C_6009_TestCase C_6010_TestCase C_6011_TestCase C_6012_TestCase C_6013_TestCase C_6014_TestCase

183

Test Assertion	Test Cases
C-TA-7001	C_7001_TestCase
C-TA-7002	C_7002_TestCase

184

Test Assertion	Test Cases
C-TA-8001	C_8001_TestCase
C-TA-8002	C_8002_TestCase
C-TA-8003	C_8003_TestCase C_8004_TestCase

185

Test Assertion	Test Cases
C-TA-10001	C_10020_TestCase
C-TA-10002	C_10010_TestCase
C-TA-10003	C_10016_TestCase
C-TA-10004	C_10011_TestCase
C-TA-10005	C_10017_TestCase
C-TA-10006	C_10010_TestCase
C-TA-10007	C_10016_TestCase
C-TA-10008	C_10013_TestCase
C-TA-10009	C_10018_TestCase
C-TA-10010	C_10012_TestCase
C-TA-10011	Untestable – Not a runtime behavior
C-TA-10012	C_10001_TestCase
C-TA-10013	C_10001_TestCase

C-TA-10014	C_10002_TestCase
C-TA-10015	C_10001_TestCase
C-TA-10016	C_10003_TestCase
C-TA-10017	C_10001_TestCase
C-TA-10018	C_10004_TestCase
C-TA-10019	C_6009_TestCase
C-TA-10020	C_10002_TestCase
C-TA-10021	C_10005_TestCase
C-TA-10022	C_10014_TestCase
C-TA-10023	C_10006_TestCase
C-TA-10024	C_10014_TestCase
C-TA-10025	C_10007_TestCase
C-TA-10026	C_10008_TestCase
C-TA-10027	C_10015_TestCase
C-TA-10028	C_10019_TestCase
C-TA-10029	C_10010_TestCase
C-TA-10030	C_10009_TestCase

Test Assertion	Test Cases
C-TA-11001	C_11001_TestCase C_11003_TestCase C_11004_TestCase
C-TA-11002	C_11002_TestCase
C-TA-11004	C_11005_TestCase
C-TA-11005	C_11006_TestCase

Test Assertion	Test Cases
C-TA-A001	C_A001_TestCase C_A002_TestCase C_A003_TestCase C_A004_TestCase C_A005_TestCase C_A007_TestCase C_A010_TestCase C_A012_TestCase

	C_B001_TestCase C_B002_TestCase
C-TA-A002	C_A008_TestCase
C-TA-A003	C_A009_TestCase
C-TA-A004	C_A015_TestCase
C-TA-A005	C_A011_TestCase
C-TA-A006	C_A013_TestCase
C-TA-A007	Optional – Not tested
C-TA-A008	C_A014_TestCase

Test Assertion	Test Cases
C-TA-C001	C_C004_TestCase
C-TA-C002	C_A010_TestCase
C-TA-C003	C_A012_TestCase
C-TA-C004	C_C005_TestCase
C-TA-C005	C_10001_TestCase C_A004_TestCase C_C001_TestCase C_C002_TestCase C_C003_TestCase C_C010_TestCase C_A011_TestCase C_A013_TestCase
C-TA-C006	C_C006_TestCase
C-TA-C007	C_C008_TestCase
C-TA-C009	C_C009_TestCase
C-TA-C010	C_C007_TestCase

Test Assertion	Test Cases
C-TA-D001	C_D001_TestCase C_D002_TestCase C_D003_TestCase C_D004_TestCase C_D005_TestCase
C-TA-D002	C_D007_TestCase C_D008_TestCase

C-TA-D003	C_D011_TestCase
C-TA-D004	C_D010_TestCase
C-TA-D005	C_D006_TestCase
C-TA-D006	C_D009_TestCase
C-TA-D007	C_D012_TestCase
C-TA-D008	C_D013_TestCase

Test Assertion	Test Cases
C-TA-F001	C_10010_TestCase
C-TA-F002	C_10010_TestCase
C-TA-F003	C_D001_TestCase C_D002_TestCase C_D003_TestCase C_D004_TestCase C_D005_TestCase
C-TA-F004	Not tested – this is base WSDL handling, not SCA specific.
C-TA-F005	C_10010_TestCase
C-TA-F006	C_10010_TestCase
C-TA-F007	C_10010_TestCase
C-TA-F008	C_F005_TestCase
C-TA-F009	C_F008_TestCase
C-TA-F010	C_10010_TestCase
C-TA-F011	C_10010_TestCase
C-TA-F012	C_10011_TestCase
C-TA-F013	C_D004_TestCase
C-TA-F014	C_F006_TestCase
C-TA-F015	C_10012_TestCase
C-TA-F016	C_F007_TestCase
C-TA-F018	C_10010_TestCase
C-TA-F019	C_F008_TestCase
C-TA-F020	Not tested – This is an assertion about the web services binding behavior
C-TA-F021	Not tested – not all platforms allow application access to header content

C-TA-F022	C_10001_TestCase
C-TA-F023	C_F001_TestCase
C-TA-F024	Not Tested
C-TA-F025	C_F002_TestCase
C-TA-F026	C_F003_TestCase C_F004_TestCase
C-TA-F027	Not tested – not all platforms allow application access to header content
C-TA-F028	Not tested – not all platforms allow application access to header content
C-TA-F029	Not tested – This is an assertion about the web services binding behavior
C-TA-F030	C_10001_TestCase
C-TA-F031	C_10001_TestCase
C-TA-F032	C_10001_TestCase
C-TA-F033	C_C001_TestCase
C-TA-F034	Not tested – This is an assertion about the web services binding behavior

9 Catalog of Test Artifacts

9.1 C Interfaces

Name	Description
Service1.h Service1P.h	Remotable service interface 1 operation with 1 input, 1 output parameter Same for a program implementation
Service1Superset.h	Remotable service interface which is a superset of Service1 interface 2 operations with 1 input,1 output parameter
Service4.h	Remotable service interface 1 operation with 1 input, 1 output parameter - Input parameter is not const
Service5.h Service5P.h	Remotable service interface with a Callback interface (Service5Callback) 1 operation with 1 input, 1 output parameter Same for a program implementation
Service5Callback.h Service5CallbackP.h	Remotable callback interface for Service5 1 operation with 1 input, 1 output parameter Same for a program implementation
Service6.h	Remotable service interface with a local Callback interface (Service6Callback) 1 operation with 1 input, 1 output parameter
Service6Callback.h	local callback interface for Service6 1 operation with 1 input, 1 output parameter
Service7.h	Local service interface with a remotable Callback interface (Service7Callback) 1 operation with 1 input, 1 output parameter
Service7Callback.h	Remotable callback interface for 7 1 operation with 1 input, 1 output parameter
Service11.h Service11P.h	Remotable interface with a function in the file that is not part of the interface - it is also a superset of Service1 Same for a program implementation

Service12.h Service12P.h	Remotable service interface with a Callback interface (Service12Callback) 1 operation with 1 input, 1 output parameter Same for a program implementation
Service12Callback.h Service12CallbackP.h	Remotable callback interface for Service12 with a function in the file that is not part of the callback interface, 1 operation with 1 input, 1 output parameter Same for a program implementation
Service13.h Service13P.h	Remotable interface containing no function declarations Remotable program implementation interface containing no struct declarations
Service15.h Service15Exp.h	Local service interface with a macro and a typedef used in the declaration. Same interface with macro and typedef expanded
Service16.h	Remotable service interface with a union parameter
Service17.h	Remotable service interface with a defined fault
Service18.h	Remotable service interface with an unnamed parameter
Service19.h	Remotable service interface with a function with multiple parameter passing styles
Service20P.h	Remotable service interface for program implementation with a function with request and response message structs having a member of the same name but different types
Service21.h	Remotable service interface with functions with multiple parameter and return types
Service22.h Service22P.h	Remotable service interface with a function with parameter that is a struct Same for a program implementation (parameter is in the request message struct)
Service23.h Service23OW.h	Remotable service interface with a function with parameters names a_x and xml Remotable service interface with a function with on non-void return type that is annotated OneWay
Service24.h Service24OW.h	Remotable service interface with a function with only an in parameter and a void return type Remotable service interface with a function with

	only an in parameter and a void return type designated oneWay
Service25.h	Remotable service interface with function annotated with @WebParam
Service26.h	Remotable service interface with function annotated with @WebResult
Service27.h	Remotable service interface with function annotated with @WebParam but incorrect simple type value
Service28.h	Remotable service interface with function annotated with @WebParam but incorrect complex type value
Service29.h	Remotable service interface with function annotated with @WebResult but incorrect simple type value
Service30.h	Remotable service interface with function annotated with @WebParam parameter is unnamed
Service31.h (system under test generated from Service31.wsdl)	Remotable service interface with a function that can throw a fault.
Service32.h (system under test generated from Service32.wsdl) Service32.Fh (system under test generated from Service32F.wsdl)	Remotable service interface with functions with multiple parameter and return types Same but two functions that throw a fault
Service33.h (system under test generated from Service33.wsdl)	Remotable service interface with function with parameter that is a struct
Service34.h (system under test generated from Service34_35.wsdl)	Remotable service interface
Service34.h (system under test generated from Service34_35.wsdl)	Remotable service interface
Service36.h (system under test generated from Service36.wsdl)	Remotable service interface
Service37.h (system under test generated from Service37.wsdl)	Remotable service interface based on WSDL with a function extension
Service38.h (system under test generated from Service38.wsdl)	Remotable service interface based on WSDL with a parameter extension
Service39.h (system under test generated from Service39.wsdl)	Remotable service interface based on WSDL with a struct extension
Service40.h (system under test generated from Service40.wsdl)	Remotable service interface based on WSDL with an enableWrapperStyle extension
Service41.h (system under test generated from Service41.wsdl)	Remotable service interface based on WSDL with a prefix extension
Service42.h (system under test generated from	Remotable service interface based on WSDL with a

Service42.wsdl)	parameter extension with a simple type
Service43.h (system under test generated from Service43.wsdl)	Remotable service interface based on WSDL with a parameter extension with a complex type
Service44.h	Remotable service interface with multiple functions
Service45.h	Service interface annotated with @WebService
Service46.h Service46P.h	Remotable service with operation1 that sleeps before returning Same for a program implementation
Service47.h	Remotable service interface with a Callback interface (Service47Callback)
Service47Callback.h	Remotable callback interface for Service47 with callback1 that sleeps before returning

9.2 C Implementations

Name	Description
Implementations xxxP.c are program-based and use yyyP.h interfaces. Implementations zzzX.c use proxy functions	*Services and References use interface Service1 unless described otherwise*
Service1Impl.c Service1Impl.componentType Service1ImplP.c Service1ImplP.componentType	1 service 0 references
Service1Impl2X.c Service1Impl2X.componentType	1 service 1 reference
Service1Impl3.c Service1Impl3.componentType	1 service 1 reference (1..n)
Service1Impl7.c Service1Impl7.componentType	1 service 1 reference (1..1) with interface Service 4
Service1Impl14.c Service1Impl14.componentType	1 service 0 references Init function
Service1Impl15.c Service1Impl15.componentType Service1Impl15P.c Service1Impl15P.componentType	1 service 1 reference (1..1) with interface Service11(P) Incorrect operation invoked
Service1Impl16.c Service1Impl16.componentType	1 service 1 reference (1..1) with interface Service15Exp
Service1Impl17.c	1 service

Service1Impl17.componentType	1 reference (1..1) with interface Service16
Service1Impl18.c Service1Impl18.componentType	1 service 0 references 5 integral properties
Service1Impl19.c Service1Impl19.componentType	1 service 0 references 1 boolean and 2 floating-point properties
Service1Impl20.c Service1Impl20.componentType	1 service 0 references 5 string/character properties
Service1Impl21.c Service1Impl21.componentType	1 service 0 references 1 complex type property
Service1Impl22.c Service1Impl22.componentType	1 service 1 reference Initial response buffer too small
Service1Impl23.c Service1Impl23.componentType	1 service 1 reference (1..1) with interface Service18
Service1Impl24.c Service1Impl24.componentType Service1Impl24P.c Service1Impl24P.componentType	1 service 1 reference (1..1) with interface Service19(P)
Service1Impl25P.c Service1Impl25P.componentType	1 service 1 reference (1..1) with interface Service20P
Service1Impl26.c Service1Impl26.componentType	1 service 1 reference (1..1) with interface Service21
Service1Impl27.c Service1Impl27.componentType Service1Impl27P.c Service1Impl27P.componentType	1 service 1 reference (1..1) with interface Service22(P)
Service1Impl28.c Service1Impl28.componentType Service1Impl28OW.c Service1Impl28OW.componentType	1 service 1 reference (1..1) with interface Service23(OW)
Service1Impl29.c Service1Impl29.componentType Service1Impl29OW.c Service1Impl29OW.componentType	1 service 1 reference (1..1) with interface Service24(OW)

Service1Impl30.c	1 service 1 reference
Service1Impl31.c Service1Impl31.componentType	1 service 0 references Composite scope with static variable and Init function
Service1Impl32.c Service1Impl32.componentType	1 service 0 references 1 multi-valued integral property
Service1Impl33.c Service1Impl33.componentType	1 service 1 reference (1..1) with interface Service25
Service1Impl34.c Service1Impl34.componentType	1 service 1 reference (1..1) with interface Service26
Service1Impl35.c Service1Impl35.componentType	1 service 1 reference (1..1) with interface Service27
Service1Impl36.c Service1Impl36.componentType	1 service 1 reference (1..1) with interface Service28
Service1Impl37.c Service1Impl37.componentType	1 service 1 reference (1..1) with interface Service29
Service1Impl38.c Service1Impl38.componentType	1 service 1 reference (1..1) with interface Service30
Service1Impl39.c Service1Impl39.componentType	1 service 1 reference (1..1) with interface Service31
Service1Impl40.c Service1Impl40.componentType Service1Impl40F.componentType	1 service 1 reference (1..1) with interface Service32 with interface Service32F
Service1Impl41.c Service1Impl41.componentType Service1Impl41P.c Service1Impl41P.componentType	1 service 1 reference (1..1) with interface Service33(P)
Service1Impl42.c Service1Impl42.componentType	1 service 1 reference (1..1) with interface Service34
Service1Impl43.c Service1Impl43.componentType	1 service 1 reference (1..1) with interface Service36
Service1Impl44.c Service1Impl44.componentType	1 service 1 reference (1..1) with interface Service37
Service1Impl45.c Service1Impl45.componentType	1 service 1 reference (1..1) with interface Service38

Service1Impl46.c Service1Impl46.componentType	1 service 1 reference (1..1) with interface Service39
Service1Impl47.c Service1Impl47.componentType	1 service 1 reference (1..1) with interface Service40
Service1Impl48.c Service1Impl48.componentType	1 service 1 reference (1..1) with interface Service41
Service1Impl49.c Service1Impl49.componentType	1 service 1 reference (1..1) with interface Service42
Service1Impl50.c Service1Impl50.componentType	1 service 1 reference (1..1) with interface Service43
Service1Impl51.c Service1Impl51.componentType	1 service 1 reference (1..1) with interface Service44
Service1Impl52.c Service1Impl52.componentType	1 service 1 reference (1..1) with interface Service45
Service1Impl53.c Service1Impl53.componentType Service1Impl53P.c Service1Impl53P.componentType Service1Impl53X.c Service1Impl53X.componentType	1 service 1 reference (1..1) with interface Service46(P) SCAInvokeAsync used
Service1Impl54.c Service1Impl54.componentType Service1Impl54X.c Service1Impl54X.componentType	1 service 1 reference (1..1) with interface Service46 SCAInvokePoll used
Service1Impl55.c Service1Impl55.componentType	1 service 1 reference (1..1) with interface Service46 SCACancelAsync used
Service1Impl56.c Service1Impl56.componentType	1 service 1 reference (1..1) Two threads created each invokes operation on reference
Service1Impl57.c Service1Impl57.componentType	1 service 0 references Uses a global variable
Service1Callback5Impl.c Service1Callback5Impl.componentType Service1Callback5ImplP.c Service1Callback5ImplP.componentType Service1Callback5ImplX.c	1 service 1 reference with Service5(P)/Service5Callback(P) interfaces

Service1Callback5ImplX.componentType	
Service1Callback6Impl.c Service1Callback6Impl.componentType	1 service 1 reference with Service6/Service6Callback interfaces
Service1Callback7Impl.c Service1Callback7Impl.componentType	1 service 1 reference with Service7/Service7Callback interfaces
Service1Callback12Impl.c Service1Callback12Impl.componentType Service1Callback12ImplP.c Service1Callback12ImplP.componentType	1 service 1 reference with Service12/Service12Callback interfaces Incorrect operation invoked
Service1Callback47Impl.c Service1Callback47Impl.componentType Service1Callback47ImplX.c Service1Callback47ImplX.componentType	1 service 1 reference with Service47/Service47Callback interfaces
Service1I_16mpl.c Service1_16Impl.componentType	2 services, one with interface Service1 and one with interface Service16 0 references - Operation of service1 invokes operation of service 16
Service1SupersetImpl2.c Service1SupersetImpl2.componentType	1 service with interface Service1Superset
Service4Impl.c Service4Impl.componentType	1 service with interface Service4
Service5ImplP.c Service5ImplP.componentType Service5ImplX.c Service5ImplX.componentType	1 service with interface Service5P
Service6Impl.c Service6Impl.componentType	1 service with interface Service6
Service7Impl.c Service7Impl.componentType	1 service with interface Service6
Service11Impl.c Service11Impl.componentType Service11ImplP.c Service11ImplP.componentType	1 service with interface Service11
Service12Impl.c Service12Impl.componentType Service12ImplP.c Service12ImplP.componentType	1 service with interface Service12(P)

Service15Impl.c Service15Impl.componentType	1 service with interface Service15
Service16Impl.c Service16Impl.componentType	1 service with interface Service16
Service17Impl.c Service17Impl.componentType	1 service with interface Service17
Service18Impl.c Service18Impl.componentType	1 service with interface Service18
Service19Impl.c Service19Impl.componentType Service19ImpPl.c Service19ImplP.componentType	1 service with interface Service19(P)
Service20ImplP.c Service20ImplP.componentType	1 service with interface Service20P
Service21Impl.c Service21Impl.componentType	1 service with interface Service21
Service22Impl.c Service22Impl.componentType Service22ImplP.c Service22ImplP.componentType	1 service with interface Service22(P)
Service23Impl.c Service23Impl.componentType Service23ImplOW.componentType	1 service with interface Service23 One function marked oneWay
Service24Impl.c Service24Impl.componentType Service24ImplP.c Service24ImplP.componentType Service24ImplOW.componentType	1 service with interface Service24(P) One function marked oneWay
Service25Impl.c Service25Impl.componentType	1 service with interface Service25
Service26Impl.c Service26Impl.componentType	1 service with interface Service26
Service27Impl.c Service27Impl.componentType	1 service with interface Service27
Service28Impl.c Service28Impl.componentType	1 service with interface Service28
Service29Impl.c Service29Impl.componentType	1 service with interface Service29

Service30Impl.c Service30Impl.componentType	1 service with interface Service30
Service31Impl.c Service31Impl.componentType	1 service with interface Service31
Service32Impl.c Service32Impl.componentType Service32ImplF.componentType	1 service with interface Service32 with interface Service32F
Service33Impl.c Service33Impl.componentType Service33ImplP.c Service33ImplP.componentType	1 service with interface Service33(P)
Service3435Impl.c Service3435Impl.componentType	1 service with interface Service34 1 service with interface Service35
Service36Impl.c Service36Impl.componentType	1 service with interface Service36
Service37Impl.c Service37Impl.componentType	1 service with interface Service37
Service38Impl.c Service38Impl.componentType	1 service with interface Service38
Service39Impl.c Service39Impl.componentType	1 service with interface Service39
Service40Impl.c Service40Impl.componentType	1 service with interface Service40
Service41Impl.c Service41Impl.componentType	1 service with interface Service41
Service42Impl.c Service42Impl.componentType	1 service with interface Service42
Service43Impl.c Service43Impl.componentType	1 service with interface Service43
Service44Impl.c Service44Impl.componentType	1 service with interface Service44
Service45Impl.c Service45Impl.componentType	1 service with interface Service45
Service46Impl.c Service46Impl.componentType Service46ImplP.c Service46ImplP.componentType	1 service with interface Service46(P)

Service47Impl.c Service47Impl.componentType Service47ImplX.c Service47ImplX.componentType	1 service with interface Service47

10 Conformance

There are no conformance statements relating to the TestCases.

A.Acknowledgements

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

Participants:

Participant Name	Affiliation
Bryan Aupperle	IBM
Andrew Borley	IBM
Jean-Sebastien Delfino	IBM
Mike Edwards	IBM
David Haney	Individual
Mark Little	Red Hat
Jeff Mischkinsky	Oracle Corporation
Peter Robbins	IBM

B.Revision History

[optional; should not be included in OASIS Standards]

Revision	Date	Editor	Changes Made
			•