



WINNF Approved Issues and Resolutions
concerning the November 2010 draft of the
SCA Next Specification
Document WINNF-11-R-0004

Version V1.0.0
14 September 2011

TERMS, CONDITIONS & NOTICES

This document has been prepared by the SCA Next WG to assist The Software Defined Radio Forum Inc. (or its successors or assigns, hereafter “the Forum”). It may be amended or withdrawn at a later time and it is not binding on any member of the Forum or of the SCA Next WG.

Contributors to this document that have submitted copyrighted materials (the Submission) to the Forum for use in this document retain copyright ownership of their original work, while at the same time granting the Forum a non-exclusive, irrevocable, worldwide, perpetual, royalty-free license under the Submitter’s copyrights in the Submission to reproduce, distribute, publish, display, perform, and create derivative works of the Submission based on that original work for the purpose of developing this document under the Forum's own copyright.

Permission is granted to the Forum’s participants to copy any portion of this document for legitimate purposes of the Forum. Copying for monetary gain or for other non-Forum related purposes is prohibited.

THIS DOCUMENT IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY USE OF THIS SPECIFICATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE FORUM, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS DOCUMENT.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the specification set forth in this document, and to provide supporting documentation.

This document was developed following the Forum's policy on restricted or controlled information (Policy 009) to ensure that that the document can be shared openly with other member organizations around the world. Additional Information on this policy can be found here: http://www.wirelessinnovation.org/page/Policies_and_Procedures

Although this document contains no restricted or controlled information, the specific implementation of concepts contain herein may be controlled under the laws of the country of origin for that implementation. Readers are encouraged, therefore, to consult with a cognizant authority prior to any further development.

Wireless Innovation Forum TM and SDR Forum TM are trademarks of the Software Defined Radio Forum Inc.

Table of Contents

TERMS, CONDITIONS & NOTICES	i
Preface	iv
Contributors	v
1 Overview	1
2 Issues	1
2.1 Issue 3: Citing OMG references	2
2.2 Issue 4: multiple use of application component term	3
2.3 Issue 5: DomainProfile properties for Resources	4
2.4 Issue 6: Add model associations	6
2.5 Issue 7: Limitation of a ComponentFactoryComponent	7
2.6 Issue 8: connectinterface of usesport with findby is superfluous	8
2.7 Issue 9: ComponentFactory is not a Framework Services Interface	10
2.8 Issue 10: DeviceManager uses ComponentFactory	11
2.9 Issue 11: InvalidPort exception issues	12
2.10 Issue 12: How to use a transport not currently listed in App. E	13
2.11 Issue 13: DeviceManager vs DeviceManagerComponent registration process	15
2.12 Issue 14: listing the "exact number of ports is optional"	16
2.13 Issue 15: CORBA one-ways	17
2.14 Issue 16: Conditional IDL inheritance	19
2.15 Issue 18: ESSOR on CORBA Profile	21
2.16 Issue 19: Component profile attribute should only return a filename	22
2.17 Issue 20: PIM Notation	27
2.18 Issue 21: Appendix E.3 Heading Typo	28
2.19 Issue 22: Figure 2-4: Conceptual Model of Resources	29
2.20 Issue 23: SCA PIM makes reference to CORBA PSM specification	30
2.21 Issue 24: PIM should use UML textual notation	31
2.22 Issue 25: 3.1.3.1.2.1 PortAccessor Description	32
2.23 Issue 26: UML Diagrams using C++ pointer notation	33
2.24 Issue 27: 3.1.3.1.2.3.5 InvalidPort errorCodes	34
2.25 Issue 28: Style of connectUsesPort and disconnectPorts	36
2.26 Issue 29: 3.1.3.1.2.5.2 disconnectPorts uses and provides	37
2.27 Issue 30: InvalidConfiguration Exception	38
2.28 Issue 31: 3.1.3.1.2.3.1 ConnectionType	39
2.29 Issue 32: 3.1.3.3 Framework Control Interfaces	40
2.30 Issue 33: Application::releaseObject disconnect port	41
2.31 Issue 34: Event Channels and Application::releaseObject behaviour p37-38	42
2.32 Issue 35: Figure 3-12 collaboration diagram	43
2.33 Issue 36: 3.1.3.3.3.3.1 CreateApplicationRequestError Exception	44
2.34 Issue 37: 3.1.3.3.5.1 ApplicationFactory::create and "start-up"	45
2.35 Issue 38: 3.1.3.3.5.1.4 ApplicationManagerComponent duplication	46
2.36 Issue 39: CORBA Profile support for IIOP/GIOP	47
2.37 Issue 40: restrict the IDL used in the PIM	48
2.38 Issue 41: allow further Application Environment Profiles	49
2.39 Issue 42: Term for CORBA and alternatives	50
2.40 Issue 43: Lightweight profile	51

2.41 Issue 44: Inclusion of narrow and unchecked_narrow in the SCA CORBA profiles....54
2.42 Issue 45: SCA Next AggregateDevice interface does not support device types (e.g.
Loadable, executable.....56

Preface

The SCA Next WG has cooperated with the JTRS SCA Next Working Panel to help define the next release of the SCA and was requested by that Panel to serve as the public liaison to coordinate input from international and non-military users of the SCA. The WG has spent more and one hundred hours in session (teleconferences and face-to-face meetings) and members have spent countless hours between sessions working to make the next release of the SCA a better product.

Many of our contributions were incorporated into the first draft of SCA Next, released by JTRS in November 2011. Since that time, the WG has operated an on-line Issue site where members and other requesting access, could raise issues, submit solutions and discuss solutions. A total of forty-three issues were submitted for consideration and this document contains a report of the recommended resolutions for 41 for which the WG reached consensus. The resolution of one other is in a separate document and one additional issue is still pending. This report will become an official resolution of the WINNF and be forwarded to the JTRS panel as our contribution to the next SCA version.

Contributors

Many members of the SCA Next WG contributed to this effort by submitting issues, submitting possible solutions and participating in the over 50 teleconferences and face-to-face meetings the WG held. The WG policy required “active participation” by attendance at two of the most recent five meetings to be allowed to vote on an issue at the WG level. The following member companies qualified to vote on one or more of the WG votes:

Communications Research Centre of Canada	PrismTech
DataSoft Corp	Raytheon
General Dynamics	Rockwell Collins
Harris Corp	Rohde & Schwarz
Indra Sistemas	SCA Technica
ITT	SELEX
L-3	Thales
MIT Lincoln Laboratories	Ultra Electronics
MITRE	Virginia Tech
Objective Interface Systems	

The following member companies should be particularly thanked for attending more than half of all meetings (in order of attendance):

ITT
Rohde & Schwarz
Ultra Electronics
Communications Research Centre of Canada
PrismTech
Raytheon
SELEX

WINNF Approved Issues and Resolutions concerning the November 2010 draft of the SCA Next Specification

1 Overview

This document contains the WINNF's recommended solutions to 41 issues (and references one whose resolution is in a separate document) raised against the 30 Nov 2011 draft of the SCA Next specification, including its appendixes. What follows are the issues raised and their recommended solution. The discussion threads and alternative solutions considered are documented in the on-line Issue tool at:

<http://groups.winnforum.org/p/is/ty/type=2>

2 Issues

Note that issues 1 and 2 do not exist, and issue 17 has been dealt with in another document.

2.1 Issue 3: Citing OMG references

Issue	Title		contributor
3	Citing OMG references		ITT
	Document	Page	Paragraph
	Main	various	various
	Date submitted		Date resolved
	2 Dec 2010		5 Apr 2011
Description			
Update normative and informative references			
OMG should be spelled out somewhere and possibly a URL to OMG documents or other indication of how they can be obtained.			
Resolution			
add more details to citation such as			
"OMG (Object Management Group)"			
and possibly include document URLs			
The Normative References section 1.5 and Informative References 1, need to be updated, eg Normative ref #5 still cites minimumCORBA rather than CORBA/e as in the Appendix and the cited document is no longer available on the OMG website.			
Do we have a position on how to deal with cited references that become unavailable, such as minimumCORBA? Should we inquire of OMG and IEEE, whether we have to right to make old documents available on our website after they have disappeared from theirs? IEEE is a bigger problem because their documents are not free.			

2.2 Issue 4: multiple use of application component term

Issue	Title		contributor
4	multiple use of application component term		ITT
Document		Page	Paragraph
Main		Various	various
Date submitted			Date resolved
2 Dec 2010			20 Jun 2011
Description			
<p>The term "application component" is used in multiple ways in the spec. As a formal component object representing the application; maybe the same as the Core Framework component "Base Application Components" 2.2.2 p12; or "ApplicationComponent pg 11 or as one of the Resources in an Application (eg, Forward pg 10, 2.2.4 pg 11,12, 14 and figure 2-1 pg 12, 3.1.3.3.1.6.1.3 pg 37 and dozens of other places.</p> <p>When spelled without a space ("ApplicationComponent") it seems to always refer to the formal component in CF but this is a very subtle distinction</p> <p>The concern is the ambiguity of Component that represents the Application from Components that are IN the Application both seems to be referred to as "Application Component"</p>			
Resolution			
<p>1) the component (Resource) within an Application should be referred to as "Resource Component" and avoid "application" as part of the name.</p> <p>2) consider using a different term for the Application object in CoreFramework, perhaps ApplicationProxy or ApplicationManager so that the formal component representation would be ApplicationProxyComponent or ApplicationManagerComponent and not be confused with the Application created by createApplication.</p>			

2.3 Issue 5: DomainProfile properties for Resources

Issue	Title		contributor
5	DomainProfile properties for Resources		ITT
	Document	Page	Paragraph
	Main	45	3.1.3.3.3.5.1.3 Behavior (Application Factory) & 3.1.3.3.1.1, first paragraph
	Date submitted		Date resolved
	2 Dec 2010		3 May 2011
Description			
<p>In 2.2.2 CF configures each device with all configure properties specified for that device in the domain profil, but for applications. only the configure properties specified for the AssemblyController are sent to the Application - the configure properties of Application Resource are ignored. This makes the values of these properties in the domain profile rather useless. This does not seem to be fixed in SCA Next.</p> <p>Specify some method for these properties to be sent to the application. Possibilities seem to be</p> <ol style="list-style-type: none"> 1) send them directly to the Resources 2) send them to the AssemblyController but wrap the propertySet into a nested single property with the component name or <component-name>Properties etc as the propertyId so that AC can forward. 			
Resolution			
<p>3.1.3.3.3.5.1.3 Behavior (Application Factory)</p> <p>The paragraph on pg 45 now reading:</p> <p>The create operation shall, in order, initialize all ApplicationResourceComponents, then establish connections for those components, and finally configure ApplicationResourceComponent (s) as identified in the assemblycontroller element in the SAD.</p> <p>Should be changed to:</p> <p>The create operation shall, in order, initialize all ApplicationResourceComponents, then establish connections for those components, and configure all</p>			

ApplicationResourceComponent(s) (except the AssemblyControllerComponent) with their profile properties and finally configure the AssemblyControllerComponent.

3.1.3.3.1.1, first paragraph, should be clarified by rewording:

The *Application* class provides the interface for all control, configuration, and status of an instantiated application in the domain, after initial configuration by the Core Framework.

2.4 Issue 6: Add model associations

Issue	Title		contributor
6	Add model associations		MITRE
Document		Page	Paragraph
Main		Various	Various
Date submitted			Date resolved
23 Dec 2010			20 Jun 2011
Description			
The component representataion UML models contain the structure of the component but they are not formal views in that they don't include other identifiers such as cardinality and associations. These extra elements need to be added in order the make the views more helpful.			
Resolution			
The WINNF does not agree to take on the editing task to update/augment these diagrams.			

2.5 Issue 7: Limitation of a ComponentFactoryComponent

Issue	Title		contributor
7	Limitation of a ComponentFactoryComponent		Rohde & Schwarz
Document		Page	Paragraph
Main		124	3.1.3.8.4.4
Date submitted			Date resolved
30 Dec 2010			5 Apr 2011
Description			
A ComponentFactoryComponent used to launch ApplicationResourceComponents is stated to be limited to SCA AEP.			
Resolution			
In 3.1.3.8.4.4, replace "SCA AEP" by "one of the approved SCA AEPs. In most cases the ComponentFactory should follow the same AEP profile as the components it creates." Or something similar.			

2.6 Issue 8: connectinterface of usesport with findby is superfluous

Issue	Title		contributor
8	connectinterface of usesport with findby is superfluous		Rohde & Schwarz
	Document	Page	Paragraph
	App D.1 Psm_Dtd	46	D.1.5.1.7.1
	Date submitted		Date resolved
	30 Dec 2010		3 May 2011
Description			
<p>The connectinterface of usesport can be performed with one of the following:</p> <ul style="list-style-type: none"> • providesport • componentsupportedinterface • findby <p>The latter is superfluous since a findby mechanism is already defined for both providesport and componentsupportedinterface.</p>			
Resolution			
<p>Remove the connectinterface of usesport with findby.</p> <p>Note that a service is now treated as a component as well!</p> <p>We also recommend the related changes suggested by Hughes Latour in his Jan 18 comment.</p> <p>findby is redundant as destination connection.</p> <p>componentsupportedinterface->findby perform just that.</p> <p>remove of findby connection element at the connectinterface element level.</p> <p>Furthermore:</p> <p>findby should be collapse to domainfinder so connection elements refer directly to domainfinder</p>			

The xml elements will look as follows:
usesport->domainfinder
providesport->domainfinder
componentsupportedinterface->domainfinder

Lastly:

supportedinterface should have
devicethatloadedthiscomponentref/deviceusedbythiscomponentref to allowed direct connection
the devices when connection by port is not necessary

The componentsupportedinterface will look as follows:
componentsupportedinterface->devicethatloadedthiscomponentref
componentsupportedinterface->deviceusedbythiscomponentref
componentsupportedinterface->domainfinder

With these changes there will be 2 type of destination

providesport: for port connection
componentsupportedinterface: for direct component connection

One more optimization change:

the useport element has a usesidentifier element
the providesport has a providesidentifier element
the componentsupportedinterface have uses supportedidentifier

why are they not just called identifier it is already scoped in
their element: useport, providesport and componentsupportedinterface

With all the changes above it would simplify the XML reader code logic since
the parsing of the subelements would be identical.

This would also simplify the graphical modelling of connections in SCA assemblies.

Since this involves changes in several places, we will not give detailed editorial changes here.

2.7 Issue 9: ComponentFactory is not a Framework Services Interface

Issue	Title		contributor
9	ComponentFactory is not a Framework Services Interface		Rohde & Schwarz
Document		Page	Paragraph
Main		10	2.2.2
Date submitted			Date resolved
30 Dec 2010			05 Apr 2011
Description			
The list of Framework Services Interfaces contains the ComponentFactory which in fact is both a Base Application Interface and a Base Device Interface (since it is used by the ApplicationFactoryComponent and the DeviceManagerComponent).			
Resolution			
We find the current hierarchy difficult to handle components and interfaces such as ComponentFactory that should appear in more than one location in the hierarchy. ComponentFactory is both a Base Application and a Base Device. Perhaps there can be a new section for such components or at least for such interfaces that allows a single entry. It could be justified to have two entries in the component sections if the entries differ, eg, different requirements in the two roles.			

2.8 Issue 10: DeviceManager uses ComponentFactory

Issue	Title		contributor
10	DeviceManager uses ComponentFactory		Rohde & Schwarz
Document		Page	Paragraph
Main		13	2.2.4
Date submitted			Date resolved
30 Dec 2010			5 Apr 2011
Description			
In Figure 2-2, only the Application uses the ComponentFactory.			
Resolution			
The figure needs updating and the notation explained in the text, but the entire chapter needs updating as well.			

2.9 Issue 11: InvalidPort exception issues

Issue	Title		contributor
11	InvalidPort exception issues		Raytheon
	Document	Page	Paragraph
	Main	24	3.1.3.1.2.3.5
	Date submitted		Date resolved
	23 Jan 2011		7 Jun 2011
Description			
<p>1. The invalidPort exception supports the reporting of only 1 connection error reason (i.e. 1,2,3) for 1.n connections. Each failed connection may have failed for different reasons.</p> <p>2. The connectUsesPort method can throw either the invalidPort or occupiedPort exception. What is the expected behavior if the requested connections contains both invalidPorts and occupiedPorts? related comment: connect mechanism implies that all connections are attempted and failed connections are returned, its implied (not a shall</p> <p>3. The invalidPort exception returns a sequence of connectionType (connections) which contains a object Type. The exeption does not need the object type to be returned to the caller for error processing and adds a performance penalty during the connection error case.</p>			
Resolution			
<p>Accept all the changes in the revised Raytheon comment and slides of 23 May, with the option 5b of slide 5, maintaining the errorCode but adding the new struct ConnectionErrorType to associate an errorCode with each failure and other corrections.</p>			

2.10 Issue 12: How to use a transport not currently listed in App. E

Issue	Title		contributor
12	How to use a transport not currently listed in App. E		Rohde & Schwarz
	Document	Page	Paragraph
	Main	75	3.1.3.4.3.4
	Date submitted		Date resolved
	28 Jan 2011		17 May 2011
Description			
<p>In APPENDIX E - PLATFORM SPECIFIC MODEL (PSM) - TRANSPORTS AND TECHNOLOGIES, it is stated that "SCA products can be realized using a variety of transports and technologies (e.g. CORBA, C++, SOAP, Data Distribution Service (DDS), MHAL Communication Service, etc.)."</p> <p>In my opinion, this would mean that a platform specific model can be defined and used by a radio provider which is not listed in Appendix E.</p> <p>On the other hand, the main document contains statements like "An ApplicationResourceComponent shall be limited to using transfer mechanisms features specified in Appendix E for the specific platform technology implemented."</p> <p>I would propose to rework the statements in the main document, e.g. to "An ApplicationResourceComponent shall be limited to using transfer mechanism features specified for the specific platform technology implemented."</p> <p>Rationale: it does not make sense to add each platform specific model to Appendix E, just to fulfil a requirement that it is specified in this document.</p>			
Resolution			
<p>App E - Platform Specific Model (PSM) - Transports and Technologies, contains the following in section E.0.2 (last paragraph)</p> <p>SCA products can be realized using a variety of transports and technologies (e.g. CORBA, C++, SOAP, Data Distribution Service (DDS), MHAL Communication Service, etc.).</p> <p>To be more explicitly consistent with 3.1.2 and 3.1.3.4.3.4 in the main specification, this should be changed to add:</p>			

An allowed transfer mechanisms shall have a PSM documenting language mappings in an attachment of Appendix E and be listed in this appendix. Newly approved transfer mechanisms shall be added to new versions of this appendix without requiring new versions of the main SCA specification.

2.11 Issue 13: DeviceManager vs DeviceManagerComponent registration process

Issue	Title		contributor
13	DeviceManager vs DeviceManagerComponent registration process		SAIC
Document		Page	Paragraph
Main		78	Figure 3-32
Date submitted			Date resolved
4 Feb 2011			20 Jun 2011
Description			
<p>Figure 3-32</p> <p>Upon start-up of the DeviceManager, Is it the DeviceManagerComponent that does the registering with ComponentRegistry or the DeviceManager interface?</p> <p>Or both?</p> <p>According to Pg 75 "The <i>components</i> launched this way register with the launching <i>DeviceManagerComponent</i> via <i>ComponentRegistry::registerComponent operation</i>." it seems like the DeviceManagerComponent registers with the ComponentRegistry.</p> <p>Is the figure inconsistent, or mis-guiding?</p>			
Resolution			
<p>Add the following some appropriate place:</p> <p>Requirements shown in sequence diagrams for Interfaces to create shall be met by each component that realizes that interface.</p> <p>In addition the list above diagram 3-32 should be edited to include only those steps shown in the diagram.</p>			

2.12 Issue 14: listing the "exact number of ports is optional"

Issue	Title		contributor
14	listing the "exact number of ports is optional"		SAIC
	Document	Page	Paragraph
	Main	26	3.1.3.1.2.5.3.1
	Date submitted		Date resolved
	18 Feb 2011		5 Apr 2011
Description			
Section 3.1.3.1.2.5.3.1 Pg 26 (line #707) -- "exact number of ports are specified"			
<p>The SCD is optional for the SDR developer, therefore, my question is what is the impact on the SDR developer if the exact number of ports is not known? From implementaton perspective, the exact number of ports is key to connect all ports at once which is more effificent for the developer. From a JTEL perspective, this information is also key and helpful to understand the exact number of ports.</p>			
Resolution			
<p>Add to 3.1.3.9.2, a sentence such as:</p> <p>"An SCD shall be supplied except for a non-SCA component launched by Core Framework."</p> <p>To make it clear that SCD is not optional in other cases in spite of diagram 3-62 listing it as 0..1</p> <p>Several XML elements in SCD have 0..n definitions, but are not intended to be "optional" and so the sentence:</p> <p>"An SCD file contains information about the interfaces that a component provides and/or uses"</p> <p>should be changed to:</p> <p>"An SCD file shall contain information about all the interfaces that the component provides and/or uses"</p>			

2.13 Issue 15: CORBA one-ways

Issue	Title		contributor
15	CORBA one-ways		Raytheon
Document	Page	Paragraph	
App D Psm DomainProfileDescriptorsFile		D.5.1.4.2	
Date submitted		Date resolved	
23 Feb 2011		23 Jun 2011	
Description			
<p>Issue: The SCA and the referenced CORBA specifications do not directly address the desired order of delivery behavior or requirements for one-way invocations.</p> <p>Two methods are utilized which affect the behavior of one-way calls:</p> <ul style="list-style-type: none"> • Via implementation-defined behavior of a particular CORBA ORB. For example, many ORBs (such as TAO) use TCP/IP to deliver oneway requests. Since TCP/IP provides ordered delivery semantics and ORB that uses TCP/IP—together with a single-threaded server configuration—can ensure ordered delivery of requests. • Via the CORBA Messaging SyncScopePolicy interface. Setting a SyncScope policy value of SYNC_WITH_TARGET will produce a behavior identical to a synchronous twoway request, but may not be desirable since it increases latency. Setting the value to SYNC_WITH_SERVER has somewhat lower latency than SYNC_WITH_TARGET since it will cause the client ORB to block only until the server ORB receives the request and sends a reply, but before the request is dispatched to a servant implementation. The client ORB will wait for any location forwarding that may occur, but this policy value is especially useful when the server is judged to be more reliable than the network <p>While these 2 methods provide implementation specific features to affect the behavior of one-way invocations, there is no requirement at the specification level for order of delivery.</p> <p>Recommendation: Investigate setting requirement(s) for the order of delivery of data in one-way calls.</p>			
Resolution			
<p>Add a way for the provider interface to specify the requirement to preserve message order. How the implementer or porter chooses to accomplish this is left open. The recommendation is to add an attribute to the domain profile specification of the provider port (but it is NOT expected that</p>			

CoreFramework will enforce this) by adding the following to Appendix D psm DomainProfileDescriptorsFile and with “yes” as the default.

D.5.1.4.2 ports

Each provides port element has a preserveinvocationorder notional attribute to indicate if the order of the invocations made to this port must be preserved or not.

```
<!ELEMENT provides  
( porttype* )>  
<!ATTLIST provides  
  repid CDATA #REQUIRED  
  providesname CDATA #REQUIRED  
  preserveinvocationorder (yes | no) ?yes?>
```


2.14 Issue 16: Conditional IDL inheritance

Issue	Title		contributor
16	Conditional IDL inheritance		PrismTech
	Document	Page	Paragraph
	main	17	Fig 2-4
	Date submitted		Date resolved
	1 Mar 2011		3 May 2011
Description			
<p>I have concerns over the mapping of optional inheritance (Figure 2-4: Conceptual Model of Resources) in PIM to CORBA PSM.</p> <p>You cannot conditionally inherit IDL interfaces, all parts of the system must be built with the same IDL. Otherwise stubs and skeletons will be out of sync and any type interrogation operations like <code>is_a()</code> and <code>narrow()</code> will not work as expected.</p> <p>Its permissible for ORBs to evaluate type information on the client side, based purely on the static typing in the stub. This could give to the situation where <code>is_a</code> will return TRUE but is actually FALSE. This will arise if you compile the client and the server with different IDL.</p> <p>Having optional inheritance in the PIM is fine, but an alternative mapping for IDL PSM needs to be discussed.</p> <p>This issue causes further problems like how to colocate components compiled from different versions of the IDL either by static linking or dynamic linking and loading (e.g dl shared libs).</p>			
Resolution			
<p>Consensus is that this is a significant problem with the proposed conditional inheritance and further that even a limited number of interface combinations (say full and lightweight) pose issues that are likely more trouble than the slight gain from reduced component size and testing.</p> <p>A suggested alternative appears to have fewer problems. This is to remove the inheriting idl interfaces, such as, Resource, and simply have components inherit or realize the interfaces they desire (eg, PropertySet, LifeCycle) and omit those they do not. If CF narrows to the interface it</p>			

is using, this should not pose the problems that conditional inheritance at the IDL level cause.

We recommend removing the concept of Conditional IDL inheritance, but considering permitting the alternative.

2.15 Issue 18: ESSOR on CORBA Profile

Issue	Title		contributor
18	ESSOR on CORBA Profile		ESSOR/SELEX
Document		Page	Paragraph
App E.1 Psm CORBA		Various	various
Date submitted			Date resolved
8 Nov 2010			21 Jun 2011
Description			
See document WINNF-11-R-0006			
Resolution			
See document WINNF-11-R-0006			

2.16 Issue 19: Component profile attribute should only return a filename

Issue	Title		contributor
19	Component profile attribute should only return a filename		CRC
Document		Page	Paragraph
Main		45, 51, 58, 62, 97	3.1.3.3.1.4.1, 3.1.3.3.3.4.2, 3.1.3.3.4.4.5, 3.1
Date submitted			Date resolved
2 Mar 2011			24 May 2011
Description			
<p>Since SCA v2.2.2 the profile attribute only returns a profile element based on the Profile Descriptor.</p> <p>The requirement to return the content of the file reference has been removed.</p> <p>The profile attribute should only return the filename.</p> <p>Enhancements:</p> <ol style="list-style-type: none"> 1-The components won't need to prepend and append the xml profile element parts. 2-The one calling the profile does not need to have any parsing code to extract the filename buried within the profile element string. 			
Resolution			
Application			
3.1.3.3.1.4.1 profile			
<i>From:</i>			
SCA40 The readonly profile attribute shall return the to the application's SAD file. Files referenced within the profile are obtained via a FileManagerComponent.			
<i>To:</i>			
SCA40 The readonly profile attribute shall return the Software Assembly Descriptor filename. The filename is absolute and is used to obtain the file via the DomainManagerComponent's FileManagerComponent.			

ApplicationFactory

3.1.3.3.3.4.2 softwareProfile

From:

softwareProfile

The softwareProfile attribute contains the Profile Descriptor for the application that is created by the ApplicationFactoryComponent. SCA67 The readonly softwareProfile attribute shall return a profile element (Profile Descriptor) with a file reference to the ApplicationComponent's SAD file. Files referenced within the profile are obtained via a FileManagerComponent.

readonly attribute string softwareProfile;

To:

The profile attribute contains the Software Assembly Descriptor filename use to create applications.

SCA67 The readonly profile attribute shall return a the Software Assembly Descriptop (SAD) filename. The filename is absolute and is used to obtain the file via the DomainManagerComponent's FileManagerComponent.

readonly attribute string profile;

DomainManager

3.1.3.3.4.4.5 domainManagerProfile

From:

domainManagerProfile

The domainManagerProfile attribute contains the DomainManagerComponent's Profile Descriptor. SCA112 The readonly domainManagerProfile attribute shall return a profile element (Profile Descriptor) with a file reference to the DomainManager Configuration Descriptor (DMD) file. Files referenced within the profile are obtained via the DomainManagerComponent's FileManagerComponent.

readonly attribute string profile;

To:

profile

The profile attribute contains the DomainManagerComponent's Profile Descriptor filename. SCA112 The readonly profile attribute shall return the DomainManager Configuration Descriptor (DMD) filename. The filename is absolute and is used to obtain the file via the DomainManagerComponent's FileManagerComponent.

readonly attribute string profile;

DeviceManager

3.1.3.3.7.4.2 deviceConfigurationProfile

From:

deviceConfigurationProfile

The readonly deviceConfigurationProfile attribute contains the device manager's profile descriptor. SCA129 The readonly deviceConfigurationProfile attribute shall return a profile element (Profile Descriptor) with a file reference to the DeviceManagerComponent's Device Configuration Descriptor (DCD) file. Files referenced within the profile are obtained via the FileSystemComponent.

readonly attribute string deviceConfigurationProfile;

To:

profile

The readonly profile attribute contains the device manager's profile descriptor filename. SCA129 The readonly profile attribute shall return the DeviceManagerComponent's Device Configuration Descriptor (DCD) filename. The filename is absolute and is used to obtain the file via the DeviceManagerComponent's FileSystemComponent.

readonly attribute string profile;

Device

3.1.3.5.4.4.2 softwareProfile

From:

softwareProfile

The softwareProfile attribute contains the profile descriptor for this device. SCA265 The readonly softwareProfile attribute shall return a profile element (Profile Descriptor) with a file

reference to the SPD file. Files referenced within the profile are obtained via the FileManagerComponent.

readonly attribute string softwareProfile;

To:

profile

The profile attribute contains the Software Profile Descriptor filename for this device. SCA265 The readonly profile attribute shall return the Software Profile Descriptor (SPD) filename. The filename is absolute and is used to obtain the file via the DeviceManagerComponent's FileManagerComponent the device is registered to.

readonly attribute string profile;

3.1.3.5.4.1 Description

From:

1. Software Profile Attribute - The SPD referenced by this profile element (Profile Descriptor) defines the logical device capabilities ...

To:

1. Profile Attribute - The filename of the SPD file which defines the logical device capabilities ...

ComponentType

3.1.3.10.2.17 ComponentType

From:

The ComponentType structure defines the basic elements of a component. The identifier field is the id of the component as specified through execparams. The softwareProfile field is either the component's SPD filename or the SPD itself. The type field is the type of component. The componentObject field is the object reference of the component. The providesPorts field is a sequence of static ports provided by the Component.

```
struct ComponentType {
    string identifier;
    string softwareProfile;
    ComponentEnumType type;
    Object componentObject;
    Ports providesPorts;
```

```
};
```

To:

The ComponentType structure defines the basic elements of a component. The identifier field is the id of the component as specified through execparams. The profile field is either the component's SPD filename or the SPD itself. The type field is the type of component. The componentObject field is the object reference of the component. The providesPorts field is a sequence of static ports provided by the Component.

```
struct ComponentType {  
    string identifier;  
    string profile;  
    ComponentEnumType type;  
    Object componentObject;  
    Ports providesPorts;  
};
```

The ApplicationType, the ApplicationFactoryType and the ManagerType all use the profile attribute and does not make any reference to the profile element (Profile Descriptor)

2.17 Issue 20: PIM Notation

Issue	Title		contributor
20	PIM Notation		PrismTech
Document		Page	Paragraph
App E.3 PSM-OMG IDL		2	
Date submitted			Date resolved
9 Mar 2011			3 May 2011
Description			
<p>"OMG IDL is the standard representation for the SCA technology independent model as well as the technology used by the OMG IDL PSM."</p> <p>The statement above appears in the last paragraph of PSM - OMG IDL (Appendix D(sic).3 PSM - OMG IDL). It would be better to put this at the start of Section 3 SCA PLATFORM INDEPENDENT MODEL (PIM) main spec. to avoid confusion between PIM notation and IDL PSM.</p>			
Resolution			
<p>Move the sentence from the end of Appendix E.3 (not D.3) PSM - OMG IDL to the start of Section 3 Platform Independent Model (PIM) in main SCA spec, since it primarily relates to the PIM.</p>			

2.18 Issue 21: Appendix E.3 Heading Typo

Issue	Title		contributor
21	Appendix E.3 Heading Typo		PrismTech
Document		Page	Paragraph
App E.3 Psm_OmgIdl.pdf			Heading
Date submitted			Date resolved
9 Mar 2011			5 Apr 2011
Description			
SCA_NextDraft_20101130_App_E.3_Psm_OmgIdl.pdf			
Heading in this document says Appendix D.3.			
Resolution			
Correct the heading.			

2.19 Issue 22: Figure 2-4: Conceptual Model of Resources

Issue	Title		contributor
22	Figure 2-4: Conceptual Model of Resources		PrismTech
Document		Page	Paragraph
main		17	Fig 2-4
Date submitted			Date resolved
9 Mar 2011			5 Apr 2011
Description			
Rather than denoting optional interfaces using comment placed over the inheritance line, it should be modelled as a constraint associated with generalisation relationship. Comments and their placement on diagrams carry no semantics in the UML.			
Resolution			
We recommend the change suggested, at least in the UML model files, including Figures 2-4, 3-11, 3-15, 3-17, 3-33, 3-42, 3-44. The diagrams shown as figures in the text might show less detail.			

2.20 Issue 23: SCA PIM makes reference to CORBA PSM specification

Issue	Title		contributor
23	SCA PIM makes reference to CORBA PSM specification		PrismTech
Document		Page	Paragraph
main		8	1.5
Date submitted			Date resolved
9 Mar 2011			5 Apr 2011
Description			
<p>Unlike the OMG Lightweight Log Service specification, the OMG Event Service (formal/04-10-02 and Event Service IDL, v1.2) is not specified as a PIM. It raises an issue with the platform independent nature of SCA PIM. At the very least we should note that fact in text, ideally there should be a PIM for Event.</p>			
Resolution			
<p>We recommend that a note be added to explain the inconsistency of the SCA PIM not being able to reference PIMs for all referenced elements such as the Event Service (only the Lightweight Event Service current has a PIM document 04-10-02). We further recommend that we suggest to OMG the need for PIMs for these so that they can be referenced in the future.</p>			

2.21 Issue 24: PIM should use UML textual notation

Issue	Title		contributor
24	PIM should use UML textual notation		PrismTech
Document		Page	Paragraph
main		Various	Various
Date submitted			Date resolved
9 Mar 2011			10 May 2011
Description			
<p>The UML defines a textual notation for operations, parameters, exceptions, attributes etc. It would be more appropriate to use this notation to express the PIM rather than IDL. Using IDL to express The PIM can cause confusion with the CORBA IDL PSM. It would also be more consistent with the usage of the UML graphical notation.</p> <p>See OMG Lightweight Log Service Specification: OMG Document formal/05-02-02: v1.1 as an example.</p>			
Resolution			
<p>While this change is desirable it is considered too difficult for this release of the spec. So we do not recommend the change for this release.</p>			

2.22 Issue 25: 3.1.3.1.2.1 PortAccessor Description

Issue	Title		contributor
25	3.1.3.1.2.1 PortAccessor Description		PrismTech
Document		Page	Paragraph
main		23	3.1.3.1.2.1
Date submitted			Date resolved
10 Mar 2011			5 Apr 2011
Description			
<p>"A component defines a specific port type by specifying an interface that inherits the PortAccessor interface."</p> <p>I would remove the above sentence from PortAccessor description. The first sentence adequately describes its purpose. This sentence actually confuses it. A component implements PortAccessor in order to provide access to and connection of ports, the component doesn't define port type as suggested in sentence.</p>			
Resolution			
We recommend removing the sentence.			

2.23 Issue 26: UML Diagrams using C++ pointer notation

Issue	Title		contributor
26	UML Diagrams using C++ pointer notation		PrismTech
Document		Page	Paragraph
Main		Various	Various
Date submitted			Date resolved
10 Mar 2011			5 Apr 2011
Description			
Diagrams should use standard UML parameter direction notation in PIM			
<direction> ::= 'in' 'out' 'inout' (defaults to 'in' if omitted).			
Resolution			
We recommend that the UML notation for direction be added to each parameter in the UML model files, but the diagrams in the text may show less detail.			

2.24 Issue 27: 3.1.3.1.2.3.5 InvalidPort errorCodes

Issue	Title		contributor
27	3.1.3.1.2.3.5 InvalidPort errorCodes		PrismTech
Document		Page	Paragraph
Main		24	3.1.3.1.2.3.5
Date submitted			Date resolved
10 Mar 2011			31 May 2011
Description			
<p>"The InvalidPort exception indicates one of the following errors has occurred in the specification of a connection:</p> <ol style="list-style-type: none"> 1. errorCode of a value of 1 indicates the provides port component is invalid (e.g. unable to narrow object reference) or illegal object reference, 2. errorCode of a value of 2 indicates the connectionId is invalid, 3. errorCode of a value of 2 indicates uses or provides port name does is invalid for the given connectionId ..." <p>General typo and text tidy up required.</p> <p>a) Add space between 1 and indicates in list element 1. Also remove bracketed text, it sort of refers to PSM technologies.</p> <p>b) Is list element 3. correct? Should the errorCode be value 3 to differentiate from errorCode 2 associated with connectionId is invalid (2).</p> <p>c) Drop "does" from list element 3.</p>			
Resolution			
<p>Correct as recommended. Should now read:</p> <ol style="list-style-type: none"> 1. errorCode of a value of 1 indicates the provides port component is invalid or illegal object reference, 2. errorCode of a value of 2 indicates the connectionId is invalid, 3. errorCode of a value of 3 indicates uses or provides port name is invalid for the given connectionId 			

[reference to narrow is removed since it is language mapping specific and covered by "invalid"

Note related recommendation in issue 11

2.25 Issue 28: Style of connectUsesPort and disconnectPorts

Issue	Title		contributor
28	Style of connectUsesPort and disconnectPorts		PrismTech
Document	Page	Paragraph	
main	25, 26	3.1.3.1.2.5.1 , 3.1.3.1.2.5.2	
Date submitted		Date resolved	
10 Mar 2011		31 May 2011	
Description			
<p>By combining multiple connection establishments into one operation invocation, each component implementation is now required to support more logic, whereas that logic was previously found in the CF management component.</p> <p>Currently in SCA 2.2/2.2.2 implementing the logic (looping around connection creation and dealing with error conditions) in the framework means we don't have inconsistency issues with components implementing different connection creation semantics. It also reduces the overall deployment size as this is done once in the framework and not in each component implementation. Its also less effort and simpler for the component implementer.</p> <p>In addition the specification is rather vague on what should happen if a connection or disconnection fails. Should all successful connections made up to the point of failure be disconnected, or should the operation continue making the remaining connections but still report the erroneous connections(s)? Whatever semantics are decided upon, they must be implemented repeatedly in every component, therefore placing more burden on the component implementor compared to SCA 2.2/2.2.2.</p>			
Resolution			
<p>To clarify the behavior when one or more connections fail, add:</p> <p>If one or more connections fail, the operation shall continue attempting the remaining connections and all successful connections shall remain connected until explicitly disconnected.</p>			

2.26 Issue 29: 3.1.3.1.2.5.2 disconnectPorts uses and provides

Issue	Title		contributor
29	3.1.3.1.2.5.2 disconnectPorts uses and provides		PrismTech
Document		Page	Paragraph
main		26	3.1.3.1.2.5.2
Date submitted			Date resolved
11 Mar 2011			24 May 2011
Description			
<p>Allowing the portName field to refer to either a uses or provides port seems to offer unnecessary flexibility. There is no description of when or why either case may be used. In the case where it specifies a provides port, I'm assuming that the disconnectPorts() call is being made on the component that owns the named provides port. In this case, in order for the connection to be disconnected at both ends, the provides port must have a reference to the uses port end. It seems an unnecessary burden for the component to maintain this backward reference just to enable this capability to disconnect from the provides end. By contrast to this, in 2.2.2 the disconnection was initiated at the uses port end, which naturally maintains the connection as a reference to the provides port. There was no need to store or manipulate any state, relating to connections, at the provides port. It's putting more burden onto the implementor of components and increasing the size and complexity.</p> <p>Its not an API call that in general is called by user code, therefore its not necessary to provide convenience of disconnecting at either end.</p>			
Resolution			
No change, but the new dynamic vs static port connections needs to be explained much better than currently in the draft.			

2.27 Issue 30: InvalidConfiguration Exception

Issue	Title		contributor
30	InvalidConfiguration Exception		PrismTech
	Document	Page	Paragraph
	main	30	3.1.3.1.5.3.1
	Date submitted		Date resolved
	18 Mar 2011		31 May 2011
Description			
<p>The InvalidConfiguration exception indicates that configuration of a component has failed. I'm not sure what the string msg parameter adds as we already know its an invalid configuration. If there was a string msg associated with each failed Property then that adds value as you can report further on each property. Currently the msg field does not really add any further informational value over an InvalidConfiguration.</p> <p>Would it not be better to associate an informational msg field with each failed property?</p> <p>See PartialConfiguration, it returns invalid properties without the top level string msg. How is the usage pattern different for this exception?</p>			
Resolution			
<p>While a standard way to associate a msg with a failed property would be good, it is not worth breaking backward compatibility and so the recommendation is "no change".</p>			

2.28 Issue 31: 3.1.3.1.2.3.1 ConnectionType

Issue	Title		contributor
31	3.1.3.1.2.3.1 ConnectionType		PrismTech
Document		Page	Paragraph
main		24	3.1.3.1.2.3.1
Date submitted			Date resolved
18 Mar 2011			31 May 2011
Description			
<p>The description for ConnectionType does not really explain usage scenarios where portName is set to a "uses" or "provides" port. You have to read on further and understand that it is an overloaded use of ConnectionType in operation connectUsersPort and getProvidedPorts. It's sort of left as a hanging question as to the usage scenarios until you have read further down. Not sure that this reads well.</p> <p>How much do we save by overloading the type usage here?</p> <p>At a minimum it maybe worth improving the text slightly for readability.</p>			
Resolution			
no change recommended.			

2.29 Issue 32: 3.1.3.3 Framework Control Interfaces

Issue	Title		contributor
32	3.1.3.3 Framework Control Interfaces		PrismTech
Document		Page	Paragraph
Main		26	3.1.3.3
Date submitted			Date resolved
18 Mar 2011			24 May 2011
Description			
<p>"The implementation of the Application, ApplicationFactory, and DomainManager interfaces are coupled together and are delivered together as a complete domain management implementation and service."</p> <p>Not sure I follow this sentence correctly. Is this suggesting that Application, ApplicationFactory and DomainManager are implemented and realized as one implementation delivered together? As this is PIM, should we not steer clear of prescribing a particular style of implementation. A CORBA PSM does not preclude this being implemented as multiple location transparent distributed objects.</p>			
Resolution			
<p>Remove the sentence from 3.1.3.3 and edit the similar sentence in 3.1.3.4 as follows:</p> <p>The implementation of the ApplicationManagerComponent, ApplicationFactoryComponent, and DomainManagerComponent components are logically coupled together to provide a complete domain management implementation and service.</p>			

2.30 Issue 33: Application::releaseObject disconnect port

Issue	Title		contributor
33	Application::releaseObject disconnect port		SAIC
Document		Page	Paragraph
Main		37	3.1.3.3.1.6.1.3
Date submitted			Date resolved
18 Mar 2011			14 Jun 2011
Description			
<p>Section 3.1.3.3.1.6.1.3 Behavior</p> <p>SCA next -- "The Application:releaseObject operation SHOULD disconnect ports that were previously connected based upon the application's software profile.</p> <p>SCA 2,2,2 -- "The releaseObject operation SHALL disconnect ports that were previously connected based upon the application's software profile."</p> <p>What is the impact on the SDR if the releaseObject no longer needs to disconnect is ports?</p>			
Resolution			
<p>change wording in Section 3.1.3.3.1.6.1.3 Behavior</p> <p>The Application:releaseObject operation SHALL disconnect all ports on other objects that are currently connected to the object based upon the application's software profile, except ports on other objects that are also being released, in which case disconnecting them is optional."</p>			

2.31 Issue 34: Event Channels and Application::releaseObject behaviour p37-38

Issue	Title		contributor
34	Event Channels and Application::releaseObject behaviour p37-38		PrismTech
Document		Page	Paragraph
Main		37-38	3.1.3.3.1.6.1
Date submitted			Date resolved
21 Mar 2011			31 May 2011
Description			
<p>"The Application::releaseObject operation may destroy an Event Service's event channel when no more consumers and producers are connected to it."</p> <p>Does this need a slight reword.</p> <p>Application::releaseObject may destroy an Event Service channel. When no more consumers and producers are connected, it WILL destroy the event channel.</p> <p>The "may" introduces a level of ambiguity as to whether it will be destroyed or not when no more consumers/producers are attached.</p>			
Resolution			
No change recommended			

2.32 Issue 35: Figure 3-12 collaboration diagram

Issue	Title		contributor
35	Figure 3-12 collaboration diagram		PrismTech
Document		Page	Paragraph
Main		38, 39	Figure 3-12
Date submitted			Date resolved
21 Mar 2011			3 May 2011
Description			
Figure 3-12 is sequence diagram not collaboration.			
Resolution			
Change the sentence before the diagram to call it a sequence diagram.			

2.33 Issue 36: 3.1.3.3.3.1 CreateApplicationRequestError Exception

Issue	Title		contributor
36	3.1.3.3.3.1 CreateApplicationRequestError Exception		PrismTech
Document		Page	Paragraph
Main		42	3.1.3.3.3.1
Date submitted			Date resolved
22 Mar 2011			14 Jun 2011
Description			
<p>Minor editorial to this 1st paragraph.</p> <p>"The CreateApplicationRequestError exception is raised when the parameter CF DeviceAssignmentSequence contains one (1) or more invalid application component-to-device assignment(s)."</p> <p>Issue 1) Should there be a ':' between CF and DeviceAssignmentSequence?</p> <p>Issue 2) Very minor feedback here, but are we using a pattern of bracketing the numbers "(1)" for there textual counterparts throughout?</p>			
Resolution			
<p>resolved as suggested:</p> <p>Edit the 1st paragraph.</p> <p>"The CreateApplicationRequestError exception is raised when the parameter CF DeviceAssignmentSequence contains one (1) or more invalid application component-to-device assignment(s)."</p> <p>We are not sure what "CF" is referring too. formal parameter name? None exist in UML diagram.</p> <p>Suggest removing the "(1)".</p>			

2.34 Issue 37: 3.1.3.3.3.5.1 ApplicationFactory::create and "start-up"

Issue	Title		contributor
37	3.1.3.3.3.5.1 ApplicationFactory::create and "start-up"		PrismTech
Document		Page	Paragraph
Main		47	3.1.3.3.3.5.1.3
Date submitted			Date resolved
24 Mar 2011			3 May 2011
Description			
<p>"The TestableObject::runTest operation (3.1.3.1.4.5.1), Resource::stop operation (3.1.3.1.6.5), and Resource::start operation (3.1.3.1.6.5) are not called at start-up." p47</p> <p>This is in the context of ApplicationFactory::create operation. What is the definition of start-up in this sentence ? Did we mean creation time?</p>			
Resolution			
<p>In section 3.1.3.3.3.5.1.3 Behavior, reword the sentence (pg 47)</p> <p>The TestableObject::runTest operation (3.1.3.1.4.5.1), Resource::stop operation (3.1.3.1.6.5), and Resource::start operation (3.1.3.1.6.5) are not called at start-up.</p> <p>as</p> <p>The TestableObject::runTest operation (3.1.3.1.4.5.1), Resource::stop operation (3.1.3.1.6.5), and Resource::start operation (3.1.3.1.6.5) shall not be called during the create operation.</p>			

2.35 Issue 38: 3.1.3.3.3.5.1.4 ApplicationManagerComponent duplication

Issue	Title		contributor
38	3.1.3.3.3.5.1.4 ApplicationManagerComponent duplication		PrismTech
Document		Page	Paragraph
Main		47	3.1.3.3.3.5.1.4
Date submitted			Date resolved
24 Mar 2011			7 Jun 2011
Description			
<p>"The create operation returns a duplicated ApplicationManagerComponent for the created application."</p> <p>Is "duplicated" something that has meaning at the PIM level? Is this a deep copy or shallow of the application? Or is this more for a PSM to deal with and not to be mentioned in a PIM given that some languages may not require duplication due to language semantics.</p>			
Resolution			
<p>The consensus resolution is that the sentence should be removed. The previous sentence is sufficient (and may have been intended as a replacement for this sentence rather than an addition).</p>			

2.36 Issue 39: CORBA Profile support for IIOP/GIOP

Issue	Title		contributor
39	CORBA Profile support for IIOP/GIOP		ITT
	Document	Page	Paragraph
	App E.1 Psm CORBA		Table
	Date submitted		Date resolved
	18 Apr 2011		14 Jun 2011
Description			
<p>A question has arisen about the meaning of the current CORBA profiles omitting IIOP/GIOP. Does this mean that Application/Waveform method calls cannot be transported by IIOP or other GIOP-based transports? Such a restriction was NOT the intention of the recommended profile, but perhaps this and the intended restriction needs to be clearer in the wording.</p>			
Resolution			
<p>Add to D.1.2.1</p> <p>5. "NA" indicates that the identified operation or feature is not applicable for Application use. While the feature may be used by the OE or by the OE on the Application's behalf, the Application shall not assume its presence.</p> <p>Then in the the table of Attachment 1 the following lines should contain NA for both Full and Lw profiles LW Log Service GIOP CDR Transfer Syntax GIOP Messages IIOP</p> <p>[note: an alternative solution recommended in WINNF-11-R-0006 is to simply remove these lines, which removes the need for an "NA" notation, but we believe that this solution is preferable.]</p> <p>Add a new section perhaps D.1.2.5 (pushing the Attachments section to D1.2.6)</p> <p>Some CORBA transports require transport-specific initialization using vendor specific functions. Since these are not standardized, they are considered non-compliant, should be used only where absolutely necessary and some compliance testing organizations may require a "waver" for their use.</p>			

2.37 Issue 40: restrict the IDL used in the PIM

Issue	Title		contributor
40	restrict the IDL used in the PIM		Rohde & Schwarz
	Document	Page	Paragraph
	Main		
	Date submitted		Date resolved
	20 Apr 2011		10 May 2011
Description			
<p>Currently in Appendix E.3 PSM - OMG IDL, the following statement appears:</p> <p>"OMG IDL is the standard representation for the SCA technology independent model ..."</p> <p>It should be clarified that not all syntax elements of OMG IDL are allowed in the PIM.</p> <p>Example: the keyword "oneway" should not be allowed in the PIM although it can be used in the PSM.</p>			
Resolution			
<p>We believe that the current text for the SCA PIM does meet this restriction and that there is not a need to explicitly state requirements on the PIM we are writing in the spec and so there is no need for a change. But this restriction should be kept in mind during future editing.</p> <p>We do recommend adding a Normative Reference to the OMG IDL spec.</p> <p>But a related issue was raised, that if a waveform is intended to be portable across multiple connection-mechanisms, then it should be specified as a Waveform PIM and then multiple Waveform PSMs. If this is done, then the Waveform PIM should also follow the restrictions suggested in this issue.</p>			

2.38 Issue 41: allow further Application Environment Profiles

Issue	Title		contributor
41	allow further Application Environment Profiles		Rohde & Schwarz
Document		Page	Paragraph
App B AEP			
Date submitted			Date resolved
20 Apr 2011			10 May 2011
Description			
<p>In Appendix B, SCA Application Environment Profiles (AEPs), exactly two AEPs are allowed for SCA compliant platforms and applications:</p> <p>"The SCA AEP and LwAEP, are the SCA required profiles referenced in sections 3.1.1, 3.2.1, and 3.3.1 of the main document. The SCA dictates that an Operating Environment provides the options and functions designated as mandatory within the supported profile and constrains an application to only use those services."</p> <p>The first sentence should be opened for further AEPs (which could be POSIX-free, object-oriented abstractions), just like the SCA Next will be opened for further connectivity mechanisms (see future Appendix E, Platform Specific Model - Transports and Technologies).</p> <p>Since the exact wording within Appendix E has to be clarified, I would suggest to use a wording similar to future Appendix E.</p> <p>As a consequence, the text in the main specification should be changed to support this:</p> <p>Section 2.2.4 Structure: delete "which is a subset of the Portable Operating System Interface (POSIX) specification [1]" plus the following sentence;</p> <p>section 3.1.1 Operating System: delete "POSIX specifications are used as a basis for this profile."</p>			
Resolution			
While there was some sharing concerns about this issue, the consensus was to recommend no changes to address it at this time.			

2.39 Issue 42: Term for CORBA and alternatives

Issue	Title		contributor
42	Term for CORBA and alternatives		ITT
Document		Page	Paragraph
Main and App E PSM Transport and Technologies		Various	various
Date submitted			Date resolved
3 May 2011			24 May 2011
Description			
At the San Diego sessions, we proposed that the term "connection mechanism" be used to refer to CORBA or alternatives in the SCA. But the current draft uses "transfer mechanism" in a number of places for this. Do we want to suggest the change to connection mechanism or is transfer mechanism acceptable to us?			
Resolution			
No change to Draft. Continue using the term "transfer mechanism" currently in the draft and NOT change to "connection mechanism"			

2.40 Issue 43: Lightweight profile

Issue	Title		contributor
43	Lightweight profile		ITT
Document		Page	Paragraph
SCA Users Guide			new
Date submitted			Date resolved
3 May 2011			20 Jun 2011
Description			
<p>There needs to be a clearer definition of the intent or use case for "lightweight" and "ultra" CORBA and AEP. It has been our position that the "Lightweight CORBA Profile" and the "Lightweight AEP" were intended to be used together in implementing a "lightweight" component. Is there also a pairing of "ultralightweight CORBA" with an "ultralightweight AEP"?</p> <p>There are some implied characteristics of a "lightweight component" that allow the restrictions imposed by the lightweight CORBA and AEP profiles but this is not currently described and will not be apparent to all readers. We should draft a discussion of this.</p> <p>The current SCA Next spec has avoided including use cases and so this discussion would probably need to go into the User's Guide.</p>			
Resolution			
<p>Additional sections for SCA Users Guide</p> <p>6 Guidance, Rationale, Use Cases and Supplementary Information</p> <p>6.1 CORBA profiles</p> <p>6.1.1 Rationale for restrictions on the use of Any</p> <p>ORB providers can provide insertion and extraction operations for known simple types and transport them without large TypeCodes that can add significantly to message sizes (type information can be larger than the actual data). For complex types, the CORBA compiler must generate code for insertion and extraction and add it to each component using the interface as well as adding the type information to each message. Even when using an ORB that supports complex-types in Any, most of the resource savings is achieved if the Application does not use them. Some additional savings in size</p>			

will be achieved by use of an ORB that does not support them.

6.1.2 Guidance on the use of Any

On systems with limited resources, the use of Any should be minimized. The use of complex types in Any should especially be avoided due to the significant size of the TypeCode that must be included in method calls using them. In some cases, this can more than double the size of the messages.

6.1.3 Guidance on the availability of commercial ORBs implementing these profiles

Initially there may be few, if any, commercial ORBs available that implement the SCA specified profiles. With few noted exceptions, the Full and Lightweight profiles are proper subsets of the CORBA/e Compact profile. This means that a platform with sufficient resources could use a CORBA/e Compact ORB and support nearly all permitted Application features and require minimal porting effort.

6.1.4 Lightweight Component or Use Case for the Lightweight profile

The Lightweight Profile, is intended for extremely limited platforms, such as most DSPs, and assumes a particular model for implementing an SCA component (Resource or Device) that we will call a “Lightweight” component. In order to avoid resource intensive features of the SCA for component management, such as the Resource interface and its inherited PropertySet interface, the model for the Lightweight profile assumes components that are not full SCA components or that the implementation of a full SCA component is split between the extremely limited platform and some less-limited platform. It is assumed that the component management functions, including the Resource interface are realized on the less-limited platform and that only port implementations (such as traffic data handling) is implemented on the limited processor. An alternative in Applications is for the Assembly Controller to directly manage a Lightweight component, not using a Resource port. So the permitted data types and method calls are restricted to those necessary for these port implementations. Note that some current standard APIs such as, Audio Port Device and GPS Device would need to be modified to follow these restrictions. Coordination between the lightweight and management parts of this component is outside the scope of this recommendation and is not required to use CORBA.

Where lightweight components might need to be deployed on even more limited processors such as FPGAs or where they have interfaces to other components on such processors, compatibility will be enhanced if data types are restricted to those realizable on such processors. So components implementing the lightweight profile are encourage to avoid the data types discouraged in the Permitted Data Types Section and marked with * in the table of Attachment 1 to Appendix E.1.

6.1.5 Guidance on restriction interface data types

It is recommended that data types be restricted in any interface to modules implemented on extremely limited platforms such as FPGAs and most DSPs.

Interfaces to code modules implemented on extremely limited platforms, such as FPGAs and

most DSPs, whether or not they are implemented in CORBA, are encouraged not to use the types discouraged in the Lightweight CORBA profile and marked with * in the table.

This recommendation is intended to permit easier porting between CORBA and non-CORBA implementations and to ensure that data can be easily passed among CORBA and non-CORBA components. Since this statement restricts implementation that do not use CORBA, it should be placed somewhere in the SCA specification outside of a CORBA profile section.

6.1.6 Rationale for CORBA feature inclusion in the profiles

The choice to include CORBA features in the profiles was driven by use cases. Some of these use cases are listed along with columns comparing Full with minimumCORBA and CORBA/e Compact in Appendix 1.

2.41 Issue 44: Inclusion of narrow and unchecked_narrow in the SCA CORBA profiles

Issue	Title		contributor
44	Inclusion of narrow and unchecked_narrow in the SCA CORBA profiles		Raytheon
Document	Page	Paragraph	
App E.1 Psm CORBA	Various	Various	
Date submitted		Date resolved	
5 May 2011		31 May 2011	
Description			
<p><u>Change Request:</u> Inclusion of narrow and unchecked_narrow in the SCA CORBA profiles.</p> <p>The SCA Next CORBA profiles contained in Appendix E (PLATFORM SPECIFIC MODEL (PSM) - COMMON OBJECT REQUEST BROKER ARCHITECTURE (CORBA)) does not contain information regarding the use of operations such as _narrow. Narrow behavior is widely used in SCA OE and Application implementations and should be specified in some manner as mandatory.</p> <p>This change request additionally suggests the addition of unchecked_narrow functionality. The behavior for type down - casting (narrow) typically is provided via synchronous 2-way server invocations which can be associated with performance penalties during product startup and application deployment.</p> <p><u>Additional information:</u></p> <p>The CORBA/e specification contains the following statement:</p> <p>9.2.7 Type Coercion Considerations</p> <p>Many programming languages map Object to programming constructs that support inheritance. Mappings to languages (such as C++ and Java) typically provide a mechanism for narrowing (down-casting) an object reference from a base interface to a more derived interface.</p> <p>So apparently while the concept of narrow IS part of CORBA, the method “_narrow()” is not defined in CORBA standards but in the language specific mappings. So this makes it a little different from the other methods in the current SCA Next CORBA Profile.</p>			

Resolution

Do NOT add an specific mention of "narrow" or "unchecked narrow" but add the following in an appropriate place:

Applications are permitted to use any feature of the standard CORBA language-mapping being used, unless specifically disallowed in this appendix.

2.42 Issue 45: SCA Next AggregateDevice interface does not support device types (e.g. Loadable, executable)

Issue	Title		contributor
45	SCA Next AggregateDevice interface does not support device types (e.g. Loadable, executable)		Raytheon
Document		Page	Paragraph
Main		97	3.1.3.5.8 AggregateDevice
Date submitted			Date resolved
9 May 2011			20 May 2011
Description			
<p>The AggregateDevice interface contains the methods addDevice and removeDevice and the attribute devices. The methods and attribute utilizes the Device type (interface) which is not mandatory for the Loadable and Executable Device.</p> <p><u>Issue:</u> Aggregate interface won't support aggregation of the Loadable and Executable Devices.</p> <p>Several different options could be utilized:</p> <ol style="list-style-type: none"> 1. Change add, remove and devices to utilize CORBA object type. This would require additional requirements for type checking <p>or</p> <ol style="list-style-type: none"> 2. Change add, remove and devices to utilize CF::Component type. This introduces potential of redundant information in the ComponentType structure and the device interfaces. <p>or</p> <ol style="list-style-type: none"> 3. Use base Device interface for both the Loadable and Executable Devices. Current base Device interface has no operations or attributes and is only used for Device interface (not Loadable and Executable). 			
Resolution			
Change add(), remove() and devices() to utilize CORBA object type. This is a new requirement on aggregateDevice to check type and keep list of object type.			

