

# **Joint Tactical Networking Center Test and Evaluation Laboratory**

## **Software Communications Architecture v2.2.2 Application Requirements List Version 2.4A**

**13 April 2022**



**Prepared for:**

Joint Tactical Networking Center  
33000 Nixie Way, Bldg 50, Suite 339  
San Diego, CA 92147-5110

**Prepared by:**

Joint Tactical Networking Center Test and Evaluation Laboratory

DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited (13 April 2022). JTNC 2022-1011

**Release Notes:**

The Software Communications Architecture (SCA) version 2.2.2 Application Requirements List version 2.4A supersedes previous versions of the document. It was created by referencing the following items:

- a. SCA Specification version 2.2.2 FINAL / 15 May 2006, Version 2.2.2.
- b. SCA Appendix B: AEP AMENDED / 22 October 2008, Version 2.2.2A.
- c. SCA Extensions FINAL / 22 December 2006, Version 2.2.2.
- d. SCA Networking Application Environment Profile / 19 March 2010, Version 2.2.2A.
- e. JTEL SCA v2.2.2 Manual Application Software Test Description (MASTD) / 13 April 2022, Version 2.3A.
- f. JTEL SCA v2.2.2 Automated Application Software Test Description (AASTD) / 13 April 2022, Version 3.14A.

The test capabilities applicable to this release are reflected in JTAP 3.14A.

**Description of Modifications:**

Index	SCA Requirements	Reasons for Modifications
1	AP0069-C002, AP0080, AP0082, AP0083, AP0084, AP0102, AP0613-C148, AP0613-C149, AP0613-C150, AP0613-C151, AP0613-C152, AP0613-C153, AP0613-C155, AP0613-C156, AP0613-C157, AP0613-C158, AP0613-C159, AP0613-C160, AP0613-C161, AP0613-C162, AP0613-C163, AP0613-C164, AP0613-C165, AP0613-C166, AP0614, AP0615, AP0628, AP0630, AP0741, AP0742, AP0743.	<p>The requirements were withdrawn from the SCA compliance test scope because of the following reasons:</p> <ul style="list-style-type: none"><li>• Redundant or untestable conditions</li><li>• Deprecated from a newer SCA specification</li><li>• None to minimal impact to functionality, interoperability, or portability.</li></ul>

**Legend for the Requirements List:**

Column Name	Description
Requirement Tag	This column displays SCA requirement unique identifier. The meaning of the tag prefix: AP = Application
Criterion Tag	This column displays the JTEL SCA criterion unique identifier. The letter “C” is the tag prefix for criterion.
Requirement/Criterion Text	This column displays the requirement/criterion statements extracted from the SCA v2.2.2 specification.
Section Number	This column displays the section where the statement is located in respect to the SCA specification, the SCA Application Environment Profile (AMENDED) or the SCA Extensions.
Test Method	The definition of each method is as follows: <ul style="list-style-type: none"><li>• Automated: The requirement can be verified by JTAP.</li><li>• Manual: The requirement must be verified by using the MASTD.</li><li>• Semi-automated: The requirement must be verified by JTAP and MASTD.</li></ul>
JTAP Test Case Name	This column displays the automated test(s) required to verify the requirement.
Manual Test Case Number	This column displays the manual test number for each requirement or criterion.

**Disclaimer:**

The following list of requirements is a guide for SCA 2.2.2 Compliance tests and should be used for reference only. The SCA 2.2.2 Specification released by JTNC Standards (previously known as JTRS Standards) is the official governing document.

### SCA 2.2.2 Application Requirements List Version 2.4

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
SCA 2.2.2 Specification - Main Body						
AP0011		A log producer shall only output log records that contain an enabled CosLwLog::LogLevel value.	3.1.2.2.1	Manual		APP_TC_001
AP0012		Log producers shall use their component identifier attribute in the producerId field of the CosLwLog::ProducerLogRecord.	3.1.2.2.1	Manual		APP_TC_001
AP0013		Log producers and CF components that are required by this specification to write log records shall operate normally in the absence of a log service or in the case where the connections to a log are nil or an invalid reference.	3.1.2.2.1	Manual		APP_TC_001
AP0063		A component (e.g., Resource, DomainManager, etc.) that consumes events shall implement the CosEventCommPushConsumer interface.	3.1.2.3.1	Manual		APP_TC_029
AP0064		A component (e.g., Resource, Device, DomainManager, etc.) that produces events shall implement the CosEventCommPushSupplier interface and use the CosEventCommPushConsumer interface for generating the events.	3.1.2.3.1	Manual		APP_TC_029
AP0065		A producer component shall not forward or raise any exceptions when the connection to a CosEventCommPushConsumer is a nil or invalid reference.	3.1.2.3.1	Manual		APP_TC_029
AP0069		The connectPort operation shall make a connection to the component identified by its input parameters.	3.1.3.1.1.5.1.3	Automated	ConnectPort	APP_TC_015
AP0070		The connectPort operation shall raise the InvalidPort exception when the input connection parameter is an invalid connection for this port.	3.1.3.1.1.5.1.5	Automated	ConnectPort InvalidPort Exception	APP_TC_014

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0070	C004	The InvalidPort exception indicates one of the following errors has occurred in the specification of a Port association: 1. errorCode 1 means the Port component is invalid (unable to narrow object reference) or illegal object reference.	3.1.3.1.1.3.1	Manual		APP_TC_014
AP0071		The connectPort operation shall raise the OccupiedPort exception when unable to accept the connections because the port is already fully occupied.	3.1.3.1.1.5.1.5	Automated	ConnectPort Occupied Port Exception	APP_TC_015
AP0072		The disconnectPort operation shall break the connection to the component identified by the input connectionId parameter.	3.1.3.1.1.5.2.3	Automated	DisconnectPort Test	APP_TC_012
AP0073		The disconnectPort operation shall raise the InvalidPort exception when the input connectionId parameter is not a known connection to the Port component.	3.1.3.1.1.5.2.5	Automated	DisconnectPort Test	APP_TC_012
AP0073	C003	The InvalidPort exception indicates one of the following errors has occurred in the specification of a Port association: 2. errorCode 2 means the Port name is not found (not used by this Port).	3.1.3.1.1.3.1	Manual		APP_TC_012
AP0074		The initialize operation shall raise an InitializeError exception when an initialization error occurs.	3.1.3.1.2.5.1.5	Manual		APP_TC_002
AP0075		The releaseObject operation shall release all internal memory allocated by the component during the life of the component.	3.1.3.1.2.5.2.3	Manual		APP_TC_019
AP0076		The releaseObject operation shall tear down the component and release it from the CORBA environment.	3.1.3.1.2.5.2.3	Automated	ReleaseObject Test	APP_TC_019
AP0078		The releaseObject operation shall raise a ReleaseError exception when a release error occurs.	3.1.3.1.2.5.2.5	Manual		APP_TC_019

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0079		The runTest operation shall use the input testId parameter to determine which of its predefined test implementations should be performed.	3.1.3.1.3.5.1.3	Automated	RunTestRuns Correctly	APP_TC_010
AP0081		The runTest operation shall return the result(s) of the test in the testValues parameter.	3.1.3.1.3.5.1.3	Automated	RunTestRuns Correctly	APP_TC_010
AP0085		The runTest operation shall raise the UnknownTest exception when there is no underlying test implementation that is associated with the input testId given.	3.1.3.1.3.5.1.5	Automated	RunTestInvalidTest ID	APP_TC_016
AP0086		The runTest operation shall raise the CF UnknownProperties exception when the input parameter testValues contains any CF DataTypes that are not known by the component's test implementation or any values that are out of range for the requested test.	3.1.3.1.3.5.1.5	Automated	RunTestApplication Additional Parameters, RunTestApplication out of range parameters, RunTestinvalid values	APP_TC_016
AP0087		The exception parameter invalidProperties shall contain the invalid testValues properties id(s) that are not known by the component or the value(s) are out of range.	3.1.3.1.3.5.1.5	Automated	RunTestApplication Additional Parameters, RunTestApplication out of range parameters	APP_TC_016
AP0089		The getPort operation shall return the CORBA object reference that is associated with the input port name.	3.1.3.1.4.5.1.4	Semi-Automated	getPort Valid CORBA Object	APP_TC_013
AP0090		The getPort operation shall raise an UnknownPort exception if the port name is invalid.	3.1.3.1.4.5.1.5	Automated	getPort Unknown Port Exceptions	APP_TC_003
AP0091		The configure operation shall assign values to the properties as indicated in the input configProperties parameter.	3.1.3.1.5.5.1.3	Automated	Resource Configuration Test With Defined Properties	APP_TC_011
AP0092		Valid properties for the configure operation shall at a minimum be the configure readwrite and writeonly properties referenced in the component's SPD.	3.1.3.1.5.5.1.3	Automated	Resource Configuration Test With Defined Properties	APP_TC_011

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0093		The configure operation shall raise a PartialConfiguration exception when some configuration properties were successfully set and some configuration properties were not successfully set.	3.1.3.1.5.5.1.5	Automated	Resource Partial and Invalid Configuration Exceptions	APP_TC_011
AP0094		The configure operation shall raise an InvalidConfiguration exception when a configuration error occurs and no configuration properties were successfully set.	3.1.3.1.5.5.1.5	Automated	Resource Partial and Invalid Configuration Exceptions	APP_TC_011
AP0095		The query operation shall return all component properties when the inout parameter configProperties is zero size.	3.1.3.1.5.5.2.3	Automated	Query Test	APP_TC_025
AP0096		The query operation shall return only those id/value pairs specified in the configProperties parameter if the parameter is not zero size.	3.1.3.1.5.5.2.3	Automated	Query Test	APP_TC_025
AP0097		Valid properties for the query operation shall be all configure properties (simple properties whose kind element's kindtype attribute is "configure") whose mode attribute is "readwrite" or "readonly" and any allocation properties with an action value of "external" as referenced in the component's SPD.	3.1.3.1.5.5.2.3	Automated	Query Test	APP_TC_025
AP0098		The query operation shall raise the CF UnknownProperties exception when one or more properties being requested are not known by the component.	3.1.3.1.5.5.2.5	Automated	Query Test	APP_TC_025
AP0099		The errorNumber parameter shall indicate a CF ErrorNumberType value.	3.1.3.1.6.3.1	Manual		APP_TC_032
AP0100		The errorNumber parameter shall indicate a CF ErrorNumberType value.	3.1.3.1.6.3.2	Manual		APP_TC_033
AP0101		The readonly identifier attribute shall contain the unique identifier for a Resource instance.	3.1.3.1.6.4.1	Automated	Verify ResourceID Test	APP_TC_017

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0103		The stop operation shall raise the StopError exception if an error occurs while stopping the resource.	3.1.3.1.6.5.2.5	Manual		APP_TC_033
AP0105		The start operation shall raise the StartError exception if an error occurs while starting the resource.	3.1.3.1.6.5.1.5	Manual		APP_TC_032
AP0107		The shutdown operation shall raise the ShutdownFailure exception when processing errors prevent the release of the resource factory from the CORBA environment or when all resources have not been released from the resource factory.	3.1.3.1.7.5.3.5	Manual		APP_TC_007
AP0108		The error number shall indicate a CF ErrorNumberType value.	3.1.3.1.7.3.3	Automated	Create a Resource from an Invalid ID Test	APP_TC_035
AP0111		The createResource operation shall return a reference to the created resource.	3.1.3.1.7.5.1.4	Automated	Create a Resource from a Valid ID Test	APP_TC_027
AP0115		The createResource operation shall raise the CreateResourceFailure exception when it cannot create the resource.	3.1.3.1.7.5.1.5	Automated	Create a Resource from an Invalid ID Test	APP_TC_035
AP0116		The releaseResource operation shall decrement the reference count for the specified resource, as indicated by the resourceId parameter.	3.1.3.1.7.5.2.3	Manual		APP_TC_006
AP0117		The releaseResource operation shall release the resource from the CORBA environment and make the resource no longer available when the resource's reference count is zero.	3.1.3.1.7.5.2.3	Manual		APP_TC_006
AP0118		The releaseResource operation shall raise the InvalidResourceId exception if an invalid resourceId is received.	3.1.3.1.7.5.2.5	Automated	Release Resource and Shutdown Resource Test	APP_TC_006
AP0119		The shutdown operation shall release the resource factory from the CORBA environment and make it unavailable to any subsequent calls to its object reference.	3.1.3.1.7.5.3.3	Automated	Release Resource and Shutdown Resource Test	APP_TC_007



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0588		Domain Profile files shall be compliant to the Document Type Definitions (DTDs) provided in Appendix D.	3.1.3.5	Automated	DomainProfile Tests	APP_TC_018
AP0590		All XML files shall have as the first two lines as an XML declaration (?xml) and a document type declaration (!DOCTYPE).	3.1.3.5	Automated	DomainProfile Tests	APP_TC_018
AP0591		A Software Package Descriptor file shall have a ".spd.xml" extension.	3.1.3.5.1	Automated	DomainProfile Tests	APP_TC_018
AP0592		A Software Component Descriptor file shall have a ".scd.xml" extension.	3.1.3.5.2	Automated	DomainProfile Tests	APP_TC_018
AP0593		A Software Assembly Descriptor file shall have a ".sad.xml" extension.	3.1.3.5.3	Automated	DomainProfile Tests	APP_TC_018
AP0594		A Properties File shall have a ".prf.xml" extension.	3.1.3.5.4	Automated	DomainProfile Tests	APP_TC_018
AP0603		Applications shall be limited to using the OS services that are designated as mandatory in the SCA Application Environment Profile (Appendix B).	3.2.1.1	Manual		APP_TC_008
AP0604		Applications shall perform file access through the CF File interfaces.	3.2.1.1	Manual		APP_TC_041
AP0606		All application processes shall have a handler registered for the POSIX-defined SIGQUIT signal.	3.2.1.1	Manual		APP_TC_042
AP0607		Applications shall be limited to using CORBA and CORBA services defined in the referenced minimum CORBA specification [5].	3.2.1.2	Manual		APP_TC_024
AP0608		Applications shall implement the Base Application Interfaces as specified in section 3.1.3.1 using the corresponding IDL in Appendix C.	3.2.1.3	Manual		APP_TC_026
AP0609		Each application component shall support the mandatory Naming Context IOR, Name Binding, and the identifier execute parameters as described in 3.1.3.2.2.5.1, in addition to their user-defined execute properties in the component's SPD.	3.2.1.3	Manual		APP_TC_028

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0610		Each application component shall bind its object reference to the Naming Context IOR using the Name Binding parameter.	3.2.1.3	Manual		APP_TC_028
AP0611		Each executable component of an application shall set its identifier attribute using the component identifier execute parameter.	3.2.1.3	Automated	Executable Identifier Tests	APP_TC_028
AP0612		Each executable component of an application shall accept the standard argv arguments of the POSIX exec family of functions [4].	3.2.1.3	Manual		APP_TC_037
AP0613		An application, each application component, and each device manager shall be accompanied by the appropriate Domain Profile files per section 3.1.3.5.	3.2.1.3	Manual		APP_TC_036
AP0613	C145	The software profile for an application consists of one SAD file that references (directly or indirectly) one or more SPD, SCD, and properties (PRF) files.	D.1	Manual		APP_TC_036
AP0613	C154	When the name attribute is a simple name, the file exists in the same directory as the SPD file. A relative directory indication begins either with "../" meaning parent directory and "/" means current directory in the name attribute. Multiple "../" and directory names can follow the initial "../" in the name attribute.	D.2.1.4.1	Manual		APP_TC_036
AP0613	C167	The software assembly element's id attribute is a DCE UUID, as specified in section D.2.1, which uniquely identifies the assembly.	D.6.1	Manual		APP_TC_036
AP0613	C168	The componentfiles element is used to indicate that an assembly is made up of 1..n component files. The componentfile element contains a reference to a local file, which is a Software Package Descriptor file.	D.6.1.2.1	Manual		APP_TC_036

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0613	C169	The componentfileref element is used to reference a particular Software PackageDescriptor file. The componentfileref element's refid attribute corresponds to the componentfile element's id attribute.	D.6.1.3.2	Manual		APP_TC_036
AP0613	C170	The componentinstantiation's id attribute is a DCE UUID that uniquely identifies the component. The id is a DCE UUID value as specified in section D.2.1. .	D.6.1.3.3	Manual		APP_TC_036
AP0613	C171	The optional findcomponent element should be specified except when there is no CORBA object reference for the component instance (e.g., DSP code).	D.6.1.3.3	Manual		APP_TC_036

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0613	C172	The optional findcomponent element (see Figure D-25) is used to obtain the CORBA object reference for the component instance. The two sources for obtaining a CORBA object reference are: 1. The componentresourcefactoryref element, which refers to a particular CF ResourceFactory componentinstantiation element found in the SAD, which is used to obtain a CF Resource instance for this componentinstantiation element. The refid attribute refers to a unique componentinstantiation id attribute. The componentresourcefactoryref element contains an optional resourcefactoryproperties element (see Figure D-26), which specifies the properties "qualifiers", for the CF ResourceFactory create call." 2. The CORBA Naming Service, which is used to find the component's CORBA object reference. The name specified in the namingservice element is a partial name that is used by the CF ApplicationFactory to form the complete context name.	D.6.1.3.3	Manual		APP_TC_036
AP0613	C173	The usesidentifier element identifies which "uses port" on the component is to participate in the connection relationship. This identifier will correspond with an id for one of the component ports specified in the Software Component Descriptor.	D.6.1.5.1.1.1	Manual		APP_TC_036
AP0613	C174	The devicethatloadedthiscomponentref element refers to a specific component found in the assembly, which is used to obtain the logical CF Device that was used to load the referenced component from the CF ApplicationFactory.	D.6.1.5.1.1.6	Manual		APP_TC_036

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0613	C175	The findby element by itself is used when the object reference is not a CF Resource type.	D.6.1.5.1.2	Manual		APP_TC_036
AP0613	C176	The supportedidentifier element identifies which supported interface on the component is to participate in the connection relationship. This identifier will correspond with the repid attribute of one of the component's supportsinterface elements, specified in the Software Component Descriptor.	D.6.1.5.1.3.1	Manual		APP_TC_036
AP0616		An application interface shall be referenced in the application's SAD externalports element, and thus declared "external", if the interface provides a service that is used by other applications.	3.2.2	Manual		APP_TC_005
AP0704		The stop operation shall not inhibit subsequent configure, query, and start operations.	3.1.3.1.6.5.2.3	Automated	Resource Stop Operation Test	APP_TC_034
AP0705		The readonly identifier attribute shall contain the unique identifier for a ResourceFactory instance.	3.1.3.1.7.4.1	Automated	Verify ResourceFactory ID Test	APP_TC_031
AP0706		If the resource already exists, the createResource operation shall return a reference to the existing resource.	3.1.3.1.7.5.1.4	Manual		APP_TC_027
AP0709		Upon execution of a software module by the create operation, a Resource or a ResourceFactory component shall register with the Naming Service.	3.1.3.2.5.1.3	Manual		APP_TC_043
AP0740		Applications shall not utilize static stringified IORs.	3.2.1.2	Manual		APP_TC_044
AP0747		Log producers shall implement a configure property which is a CF Properties type with an id of "PRODUCER_LOG_LEVEL" and a value that is a CosLwLog::LogLevelSequence.	3.1.2.2.1	Automated	ProducerLogLevel Tests	APP_TC_001
AP0748		The createResource operation shall create a resource if no resource exists for the given resourceId and shall assign the given resourceId to a new resource.	3.1.3.1.7.5.1.3	Manual		APP_TC_027

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
AP0753		The createResource operation shall set a reference count to one, when the resource is initially created, or increment the reference count by one, when the resource already exists.	3.1.3.1.7.5.1.3	Manual		APP_TC_027
<b>SCA 2.2.2 Appendix B - SCA Application Environment Profile</b>						
AP0655		Any Ada application shall be restricted to using the equivalent Ada functionality, as defined in POSIX Ada language binding (ISO/IEC 14519:2001), designated as mandatory by the AEP or may use the C interface.	B.4	Manual		APP_TC_020
AP0662		An application that conforms to the AEP shall not call the kill( ) function with a negative argument because this profile does not require process group functionality.	B.4.1.4	Manual		APP_TC_030
<b>SCA 2.2.2 Extensions</b>						
AP0789		An Application Deployment Descriptor file shall have an ".add.xml" extension.	2.9 Extension	Manual		APP_TC_051
<b>SCA 2.2.2 Appendix B - SCA Application Environment Profile (AMENDED / 22 October 2008, Version 2.2.2A)</b>						
AP0792		Any Ada application shall be restricted to using the equivalent Ada functionality, as defined in POSIX Ada language binding (ISO/IEC 14519:2001), designated as mandatory by the AEP or may use the C interface.	B.4	Manual		APP_TC_047
AP0799		An application that conforms to the AEP shall not call the kill( ) function with a negative argument because this profile does not require process group functionality.	B.4.1.4	Manual		APP_TC_049