

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

## **Software Communications Architecture v2.2.2 Operating Environment 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) v2.2.2 Operating Environment (OE) Requirements List version 2.4A supersedes the 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 OE Software Test Description (MOESTD) / 13 April 2022, Version 3.3A.
- f. JTEL SCA v2.2.2 Automated Operating Environment Software Test Description (AOESTD) / 13 April 2022, Version 3.14A

The test capabilities applicable to this release are reflected in JTAP v3.14A.

**Description of Modifications from previous requirement list:**

Index	SCA Requirements	Reasons for Modifications
1	OE0069-C002, OE0080, OE0082, OE0083, OE0084, OE0102, OE0218, OE0442, OE0443, OE0613-C148, OE0613-C149, OE0613-C150, OE0613-C151, OE0613-C152, OE0613-C153, OE0613-C155, OE0613-C156, OE0613-C157, OE0613-C158, OE0613-C159, OE0613-C160, OE0613-C161, OE0613-C162, OE0613-C163, OE0613-C164, OE0613-C165, OE0613-C166, OE0613-C178, OE0613-C179, OE0613-C180, OE0613-C181, OE0614, OE0615, OE0620, OE0627, OE0628, OE0630, OE0633, OE0741, OE0742, OE0743, OE0803.	<p>These requirements were withdrawn from the SCA compliance test scope because of the following reasons: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: OE = Operating Environment
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 MOESTD.</li><li>• Semi-automated: The requirement must be verified by JTAP and MOESTD.</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 OE 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						
OE0001		The OE shall provide the functions and options designated as mandatory by the AEP defined in Appendix B.	3.1.1	Manual		OE_TC_059
OE0002		The OE and related file systems shall support a filename length of 40 characters and a pathname length of 1024 characters.	3.1.1	Semi-Automated	FileSystemcreate remove, FileSystemmkdir rmdir	OE_TC_060
OE0003		The OE shall include middleware that, at a minimum, provides the services and capabilities of minimumCORBA as specified by the OMG Document in reference [5].	3.1.2	Manual		OE_TC_061
OE0007		The "kind" element of each NameComponent shall be "" (null string).	3.1.2.1	Automated	DomainManager Naming Service Register	OE_TC_001
OE0011		A log producer shall only output log records that contain an enabled CosLwLog::LogLevel value.	3.1.2.2.1	Semi-Automated	Application PRODUCER_LOG_LEVEL, DeviceManager PRODUCER_LOG_LEVEL, DomainManager PRODUCER_LOG_LEVEL	OE_TC_002
OE0012		Log producers shall use their component identifier attribute in the producerId field of the CosLwLog::ProducerLogRecord.	3.1.2.2.1	Semi-Automated	Application PRODUCER_LOG_LEVEL, DeviceManager PRODUCER_LOG_LEVEL, DomainManager PRODUCER_LOG_LEVEL	OE_TC_002
OE0013		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		OE_TC_002

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0061		The OE shall provide an implementation of the CORBA Event Service.	3.1.2.3.1	Automated	EventService IDM_Channel	OE_TC_063
OE0062		The Event Service shall implement the PushConsumer and PushSupplier interfaces of the CosEventComm module as described in OMG Event Service Specification [8] using the IDL found in that specification.	3.1.2.3.1	Automated	EventService disconnect_push_consumer CORBA::OBJECT_NOT_EXIST, EventService disconnect_push_consumer, EventService IDM_Channel, Eventservice push CosEventComm::Disconnected	OE_TC_063
OE0063		A component (e.g., Resource, DomainManager, etc.) that consumes events shall implement the CosEventCommPushConsumer interface.	3.1.2.3.1	Manual		OE_TC_063
OE0064		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	Semi-Automated	DomainManager registerWithEventChannel unregisterFromEventChannel	OE_TC_063
OE0065		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		OE_TC_063
OE0066		The Incoming Domain Management Channel name shall be "IDM_Channel".	3.1.2.3.1	Automated	DomainManager registerWithEventChannel unregisterFromEventChannel, EventService Event Channel Function	OE_TC_003
OE0067		The Outgoing Domain Management Channel name shall be "ODM_Channel".	3.1.2.3.1	Automated	DomainManager registerWithEventChannel unregisterFromEventChannel, EventService Event Channel Function	OE_TC_003
OE0069		The connectPort operation shall make a connection to the component identified by its input parameters.	3.1.3.1.1.5.1.3	Manual		OE_TC_020

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0070		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	Manual		OE_TC_020
OE0071	C003	A port may support several connections. The input connectionId is a unique identifier to be used by the disconnectPort operation when breaking a specific connection.	3.1.3.1.1.5.1.7	Manual		OE_TC_020
OE0071		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	Manual		OE_TC_032
OE0072		The disconnectPort operation shall break the connection to the component identified by the input connectionId parameter.	3.1.3.1.1.5.2.3	Manual		OE_TC_035
OE0073		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	Manual		OE_TC_035
OE0073	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		OE_TC_035
OE0074		The initialize operation shall raise an InitializeError exception when an initialization error occurs.	3.1.3.1.2.5.1.5	Manual		OE_TC_037
OE0075		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		OE_TC_038
OE0076		The releaseObject operation shall tear down the component and release it from the CORBA environment.	3.1.3.1.2.5.2.3	Automated	Application releaseObject, Device releaseObject	OE_TC_214
OE0078		The releaseObject operation shall raise a ReleaseError exception when a release error occurs.	3.1.3.1.2.5.2.5	Manual		OE_TC_039

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0079		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	Manual		OE_TC_041
OE0081		The runTest operation shall return the result(s) of the test in the testValues parameter.	3.1.3.1.3.5.1.3	Manual		OE_TC_043
OE0085		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	Device runtest UnknownTest	OE_TC_215
OE0086		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	Device runtest UnknownProperties	OE_TC_215
OE0087		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	Application runTest UnknownProperties, Device runTest UnknownProperties, Application query UnknownProperties	OE_TC_093
OE0089		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	Automated	Device getPort, DeviceManager getPort	OE_TC_216
OE0090		The getPort operation shall raise an UnknownPort exception if the port name is invalid.	3.1.3.1.4.5.1.5	Automated	Device getPort UnknownPort, DeviceManager getPort UnknownPort	OE_TC_216
OE0091		The configure operation shall assign values to the properties as indicated in the input configProperties parameter.	3.1.3.1.5.5.1.3	Semi-Automated	Device configure, DomainManager configure	OE_TC_047
OE0092		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	Manual		OE_TC_048

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0093		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	Device configure PartialConfiguration, DeviceManager configure PartialConfiguration, DomainManager configure PartialConfiguration	OE_TC_049
OE0094		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	Device configure InvalidConfiguration, DeviceManager configure InvalidConfiguration, DomainManager configure InvalidConfiguration	OE_TC_288
OE0095		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	Device query, DeviceManager query, DomainManager query	OE_TC_186
OE0096		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	Device query, DeviceManager query, DomainManager query	OE_TC_186
OE0097		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	Device query, DeviceManager query, DomainManager query	OE_TC_186
OE0098		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	Semi-Automated	DeviceManager query UnknownProperties, DomainManager query UnknownProperties	OE_TC_187
OE0099		The errorNumber parameter shall indicate a CF ErrorNumberType value.	3.1.3.1.6.3.1	Semi-Automated	Application start StartError	OE_TC_050
OE0100		The errorNumber parameter shall indicate a CF ErrorNumberType value.	3.1.3.1.6.3.2	Semi-Automated	Application stop StopError	OE_TC_064



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0101		The readonly identifier attribute shall contain the unique identifier for a Resource instance.	3.1.3.1.6.4.1	Automated	Application identifier Attribute, Device identifier Attribute	OE_TC_232
OE0103		The stop operation shall raise the StopError exception if an error occurs while stopping the resource.	3.1.3.1.6.5.2.5	Semi-Automated	Application stop StopError	OE_TC_064
OE0105		The start operation shall raise the StartError exception if an error occurs while starting the resource.	3.1.3.1.6.5.1.5	Semi-Automated	Application start StartError	OE_TC_050
OE0120		Framework Control Interfaces shall be implemented using the CFIDL presented in Appendix C.	3.1.3.2	Manual		OE_TC_116
OE0121		The readonly profile attribute shall contain a profile element (Profile Descriptor) with a file reference to the application's SAD file.	3.1.3.2.1.4.1	Automated	Application profile Attribute	OE_TC_233
OE0122		This readonly name attribute shall contain the name of the created application.	3.1.3.2.1.4.2	Automated	Application name Attribute	OE_TC_233
OE0123		The componentNamingContexts attribute shall contain the list of components' Naming Service Context within the application for those components using CORBA Naming Service.	3.1.3.2.1.4.3	Automated	Application componentNamingContexts Attribute	OE_TC_233
OE0124		The componentProcessIds attribute shall contain the list of components' process IDs within the Application for components that are executing on a device.	3.1.3.2.1.4.4	Automated	Application componentProcessIds Attribute	OE_TC_233
OE0125		The componentDevices attribute shall contain a list of devices, which each component either uses, is loaded on or is executed on.	3.1.3.2.1.4.5	Automated	Application componentDevices Attribute	OE_TC_233
OE0126		The componentImplementations attribute shall contain the list of components' SPD implementation IDs within the application for those components created.	3.1.3.2.1.4.6	Automated	Application componentImplementations Attribute	OE_TC_233

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0127		The application shall delegate the implementation of the inherited Resource operations (runTest, start, stop, configure, and query) to the Application Resource component identified by the application's SAD assembly controller element (Assembly Controller).	3.1.3.2.1.5	Automated	Application start, Application stop, Application configure query, Application query empty set, Application runTest	OE_TC_189
OE0128		The application shall propagate exceptions raised by the application's Assembly Controller's operations.	3.1.3.2.1.5	Automated	Application query UnknownProperties, Application releaseObject ReleaseError, Application configure InvalidConfiguration, Application configure PartialConfiguration, Application runTest UnknownTest, Application runTest UnknownProperties, Application start StartError, Application stop StopError	OE_TC_189
OE0129		The initialize operation shall not be propagated to the application's components or its Assembly Controller.	3.1.3.2.1.5	Automated	Application initialize	OE_TC_189
OE0131		The Application::releaseObject operation shall release each application component not created by a resource factory by utilizing the component's Resource::releaseObject operation.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_190
OE0132		The Application::releaseObject operation shall release each component created by a resource factory via the ResourceFactory::releaseResource operation.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_190
OE0133		The Application::releaseObject operation shall terminate a resource factory when no more resources are managed by the resource factory via the ResourceFactory::shutdown operation.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_190

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0134		The Application::releaseObject operation shall terminate the processes / tasks on allocated executable devices belonging to each application component by utilizing the ExecutableDevice::terminate operation.	3.1.3.2.1.6.1.3	Automated	DomainManager registerService unregisterService	OE_TC_190
OE0135		The releaseObject operation shall de-allocate the memory associated with each application component instance from its allocated device by utilizing the LoadableDevice::unload operation.	3.1.3.2.1.6.1.3	Automated	DomainManager registerService unregisterService	OE_TC_191
OE0136		The releaseObject operation shall deallocate the device capacities that were allocated during application creation.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_191
OE0137		The application shall release all object references to the components making up the application.	3.1.3.2.1.6.1.3	Manual		OE_TC_108
OE0138		The releaseObject operation shall disconnect ports that were previously connected based upon the application's software profile.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_192
OE0139		The releaseObject operation shall disconnect consumers and producers from a CORBA Event Service's event channel based upon the software profile.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_192
OE0140		For components (e.g., Resource, ResourceFactory) that are registered with Naming Service, the releaseObject operation shall unbind those components and destroy the associated naming contexts as necessary from the Naming Service.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_193
OE0141		The releaseObject operation for an application shall disconnect ports first, then release the resources and the resource factories, then call the terminate operation, and lastly call the unload operation on the devices.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_194

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0142		The releaseObject operation shall, upon successful application release, write an ADMINISTRATIVE_EVENT log record.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_195
OE0143		The releaseObject operation shall, upon unsuccessful application release, write a FAILURE_ALARM log record.	3.1.3.2.1.6.1.3	Automated	Application releaseObject ReleaseError	OE_TC_195
OE0144		The releaseObject operation shall send a DomainManagementObjectRemovedEventType event to the Outgoing Domain Management event channel upon successful release of an application.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	OE_TC_195
OE0144	C018	The producerId is the identifier attribute of the released application.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	
OE0144	C019	The sourceId is the identifier attribute of the released application.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	
OE0144	C020	The sourceName is the name attribute of the released application.	3.1.3.2.1.6.1.3	Automated	Application releaseObject	
OE0144	C021	The sourceCategory is "APPLICATION".	3.1.3.2.1.6.1.3	Automated	Application releaseObject	
OE0149		The releaseObject operation shall raise a ReleaseError exception when internal processing errors prevent the successful release of any application component.	3.1.3.2.1.6.1.5	Automated	Application releaseObject ReleaseError	OE_TC_234
OE0150		The getPort operation shall return object references only for input port names that match the port names that are in the application SAD externalports element.	3.1.3.2.1.6.2.4	Automated	Application getPort	OE_TC_235
OE0151		The getPort operation shall raise an UnknownPort exception if the port is invalid.	3.1.3.2.1.6.2.5	Automated	Application getPort UnknownPort	OE_TC_235
OE0152		The error number shall indicate a CF ErrorNumberType value.	3.1.3.2.2.3.2	Automated	ApplicationFactory create CreateApplicationError	OE_TC_125, OE_TC_182, OE_TC_183

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0153		The name attribute shall be identical to the softwareassembly element name attribute of the application's Software Assembly Descriptor file.	3.1.3.2.2.4.1	Automated	ApplicationFactory name Attribute	OE_TC_236
OE0154		The readonly softwareProfile attribute shall contain a profile element (Profile Descriptor) with a file reference to the application's SAD file.	3.1.3.2.2.4.2	Automated	ApplicationFactory softwareProfile Attribute	OE_TC_236
OE0155		The readonly identifier attribute shall contain the unique identifier for an ApplicationFactory instance.	3.1.3.2.2.4.3	Automated	ApplicationFactory identifier Attribute	OE_TC_236
OE0156		The identifier shall be identical to the softwareassembly element id attribute of the application factory's Software Assembly Descriptor file.	3.1.3.2.2.4.3	Automated	ApplicationFactory identifier Attribute	OE_TC_236
OE0157		The create operation shall use the SPD implementation element to locate candidate devices capable of loading and executing application software modules.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create with deviceAssignments	OE_TC_237
OE0158		The create operation shall use the allocateCapacity operation to perform the comparison of allocation properties of the application to those of each candidate device for those application component properties whose kindtype is allocation and whose action element is external.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_196
OE0161		The create operation shall execute the application software modules as specified in the application's Software Assembly Descriptor (SAD) file.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_197
OE0162		The create operation shall use each software module's SPD implementation code's stack size and priority elements, when specified, for the execute options parameters.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_197

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0163		The create operation shall include the mandatory execute parameters Naming Context IOR, Name Binding, and Component Identifier, as described in this section, in the parameters parameter of the ExecutableDevice::execute operation when the CORBA instance's componentinstantiation element of the SAD contains a findcomponent element with a namingservice sub-element.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_197
OE0166		The create operation shall create any naming contexts that do not exist but which are required for successful binding to the Naming Context IOR.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_198
OE0167		The structure of the naming context path shall be "/ DomainName / [optional naming context sequences]".	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_198
OE0172		The Application_Name field shall be identical to the create operation's input name parameter.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_161
OE0173		The create operation shall pass the values of the "execparam" properties of the componentinstantiation componentproperties element contained in the SAD, as parameters to the execute operation.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_200
OE0174		The create operation shall, in order, initialize all application resources, then establish connections for those resources, and finally configure the application component indicated by the assemblycontroller element in the SAD.	3.1.3.2.2.5.1.3	Semi-Automated	ApplicationFactory create PseudoWaveform	OE_TC_165
OE0176		The create operation shall configure the application component indicated by the assemblycontroller element in the SAD if that component has properties with a kindtype of "configure" and a mode of "readwrite" or "writeonly".	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_200

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0177		The create operation shall use the union of the properties contained in the input initConfiguration parameter of the create operation and the assembly controller's componentinstantiation element properties with a kindtype of "configure" and a mode of "readwrite" or "writeonly".	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_200
OE0178		Values contained in the input initConfiguration parameter shall have precedence over the values of the assembly controller's componentinstantiation element properties when they reference the same property.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_200
OE0179		The create operation, when creating a resource from a resource factory, shall pass the componentinstantiation componentresourcefactoryref element properties whose kindtype element is factoryparamas the qualifiers parameter to the referenced ResourceFactory component's createResource operation.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_201
OE0183		For connections to an event channel, the create operation shall connect a CosEventComm::PushConsumer or CosEventComm::PushSupplier object to the event channel as specified in the SAD's domainfinder element.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_202
OE0184		The create operation shall create the specified event channel if the event channel does not exist.	3.1.3.2.2.5.1.3	Manual		OE_TC_074
OE0185		The create operation shall return an Application object reference for the created application when the application is successfully created.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_203
OE0186		The create operation shall, upon successful application creation, write an ADMINISTRATIVE_EVENT log record.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_203

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0187		The create operation shall, upon unsuccessful application creation, write a FAILURE_ALARM log record.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create CreateApplicationError, ApplicationFactory create CreateApplicationRequestError, ApplicationFactory create InvalidInitConfiguration	OE_TC_203
OE0190		The create operation shall send a DomainManagementObjectAddedEventType event to the Outgoing Domain Management event channel upon successful creation of an application.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_204
OE0190	C028	The producerId is the identifier attribute of the application factory.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_204
OE0190	C029	The sourceId is the identifier attribute of the created application.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_204
OE0190	C030	The sourceName is the name attribute of the created application.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_204
OE0190	C031	The sourceIOR is the object reference for the created application.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_204
OE0190	C032	The sourceCategory is "APPLICATION".	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_204
OE0196		The create operation shall raise the CreateApplicationRequestError exception when the input CF DeviceAssignmentSequence parameter contains one (1) or more invalid application component to device assignment(s).	3.1.3.2.2.5.1.5	Automated	ApplicationFactory create CreateApplicationRequestError	OE_TC_205
OE0197		The create operation shall raise the CreateApplicationError exception when the create request is valid but the application cannot be successfully instantiated due to internal processing error(s).	3.1.3.2.2.5.1.5	Automated	ApplicationFactory create CreateApplicationError	OE_TC_125



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0198		The create operation shall raise the InvalidInitConfiguration exception when the input initConfiguration parameter is invalid.	3.1.3.2.2.5.1.5	Automated	ApplicationFactory create InvalidInitConfiguration	OE_TC_205
OE0199		The InvalidInitConfiguration invalidProperties parameter shall identify the invalid properties.	3.1.3.2.2.5.1.5	Automated	ApplicationFactory create InvalidInitConfiguration	OE_TC_205
OE0200		The error number shall indicate a CF ErrorNumberType value.	3.1.3.2.3.3.1	Manual		OE_TC_057
OE0201		The error number shall indicate a CF ErrorNumberType value.	3.1.3.2.3.3.7	Manual		OE_TC_124, OE_TC_126, OE_TC_127
OE0202		The error number shall indicate a CF ErrorNumberType value.	3.1.3.2.3.3.8	Manual		OE_TC_055, OE_TC_128, OE_TC_129
OE0203		The error number shall indicate a CF ErrorNumberType value.	3.1.3.2.3.3.9	Manual		OE_TC_122
OE0204		The readonly deviceManagers attribute shall contain a list of registered device managers that have registered with the domain manager.	3.1.3.2.3.4.1	Automated	DomainManager deviceManagers Attribute	OE_TC_239
OE0205		The domain manager shall write an ADMINISTRATIVE_EVENT log to a domain manager's log, when the deviceManagers attribute is obtained by a client.	3.1.3.2.3.4.1	Automated	DomainManager deviceManagers Attribute	OE_TC_239
OE0206		The readonly applications attribute shall contain the list of Applications that have been instantiated.	3.1.3.2.3.4.2	Automated	DomainManager applications Attribute, ApplicationFactory create PseudoWaveform	OE_TC_206
OE0207		The domain manager shall write an ADMINISTRATIVE_EVENT log record to a domain manager's log, when the application's attribute is obtained by a client.	3.1.3.2.3.4.2	Automated	DomainManager applications Attribute	OE_TC_206

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0208		The readonly applicationFactories attribute shall contain a list with one application factory per application (SAD file and associated files) successfully installed (i.e. no exception raised).	3.1.3.2.3.4.3	Automated	DomainManager applicationFactories Attribute	OE_TC_240
OE0209		The domain manager shall write an ADMINISTRATIVE_EVENT log record to a domain manager's log, when the applicationFactories attribute is obtained by a client.	3.1.3.2.3.4.3	Automated	DomainManager applicationFactories Attribute	OE_TC_240
OE0210		The readonly fileMgr attribute shall contain the domain manager file manager.	3.1.3.2.3.4.4	Automated	DomainManager fileMgr Attribute and Components	OE_TC_241
OE0211		The domain manager shall write an ADMINISTRATIVE_EVENT log record to a domain manager's log, when the fileMgr attribute is obtained by a client.	3.1.3.2.3.4.4	Automated	DomainManager fileMgr Attribute and Components	OE_TC_241
OE0212		The readonly domainManagerProfile attribute shall contain a profile element (Profile Descriptor) with a file reference to the DomainManager Configuration Descriptor (DMD) file.	3.1.3.2.3.4.5	Automated	DomainManager domainManagerProfile Attribute	OE_TC_242
OE0213		The readonly identifier attribute shall contain a unique identifier for a DomainManager instance.	3.1.3.2.3.4.6	Automated	DomainManager identifier Attribute	OE_TC_243
OE0214		The identifier shall be identical to the domainmanagerconfiguration element id attribute of the domain manager's Descriptor (DMD) file.	3.1.3.2.3.4.6	Automated	DomainManager identifier Attribute	OE_TC_243
OE0215		The domain manager shall register itself with the CORBA Naming Service during component construction.	3.1.3.2.3.5	Automated	DomainManager Naming Service Register	OE_TC_156
OE0217		The logs utilized by the DomainManager implementation shall be defined in the DMD.	3.1.3.2.3.5	Manual		OE_TC_054
OE0219		The domain manager shall create its own FileManager component that consists of all registered device manager's FileSystems.	3.1.3.2.3.5	Automated	DomainManager fileMgr Attribute and Components	OE_TC_241

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0220		Upon system startup, the domain manager shall restore application factories for applications that were previously installed by the DomainManager::installApplication operation.	3.1.3.2.3.5	Automated	DomainManager Restore ApplicationFactories Verification	OE_TC_226
OE0221		The domain manager shall add the restored application factories to the DomainManager applicationFactories attribute.	3.1.3.2.3.5	Automated	DomainManager Restore ApplicationFactories Verification	OE_TC_226
OE0222		The domain manager shall create the Incoming Domain Management and Outgoing Domain Management event channels.	3.1.3.2.3.5	Automated	DomainManager registerWithEventChannel unregisterFromEventChannel	OE_TC_226
OE0223		The registerDeviceManager operation shall add the device manager indicated by the input deviceMgr parameter to the DomainManager deviceManagers attribute, if it does not already exist.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_208
OE0224		The registerDeviceManager operation shall add the input device manager's registered devices and each registered device's attributes (e.g., identifier, softwareProfile, allocation properties, etc.) to the domain manager.	3.1.3.2.3.6.1.3	Automated	DomainManager Update Attributes List	OE_TC_208
OE0225		The registerDeviceManager operation shall add all the services contained in the registering device manager's registeredServices attribute to the domain manager.	3.1.3.2.3.6.1.3	Automated	DomainManager Update Attributes List	OE_TC_208
OE0226		The registerDeviceManager operation shall establish any connections for the device manager indicated by the input deviceMgr parameter, which are specified in the connections element of the device manager's Device Configuration Descriptor (DCD) file, that are possible with the current set of registered devices and services.	3.1.3.2.3.6.1.3	Manual		OE_TC_157
OE0226	C040	Connections not currently possible are left unconnected pending future device/ service registrations.	3.1.3.2.3.6.1.4	Manual		OE_TC_157

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0228		For connections established for a CORBA Event Service's event channel, the registerDeviceManager operation shall connect a CosEventComm::PushConsumer or CosEventComm::PushSupplier object to the event channel as specified in the DCD's domainfinder element.	3.1.3.2.3.6.1.3	Automated	DomainManager CosEventComm Connect	OE_TC_209
OE0229		If the event channel does not exist, the registerDeviceManager operation shall create the event channel.	3.1.3.2.3.6.1.3	Automated	DomainManager CosEventComm Connect	OE_TC_209
OE0230		The registerDeviceManager operation shall obtain all the software profiles from the registering device manager's file systems.	3.1.3.2.3.6.1.3	Automated	DomainManager Update Attributes List	OE_TC_210
OE0231		The registerDeviceManager operation shall mount the device manager's file system to the domain manager's file manager.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager MountFileSystem	OE_TC_210
OE0232		The mounted FileSystemname shall have the format, "/DomainName/HostName", where DomainName is the name of the domain and HostName is the input deviceMgr's label attribute.	3.1.3.2.3.6.1.3	Automated	DomainManager fileMgr Attribute and Components	OE_TC_210
OE0233		The registerDeviceManager operation shall, upon unsuccessful device manager registration, write a FAILURE_ALARM log record to a domain manager's Log.	3.1.3.2.3.6.1.3	Semi-Automated	DomainManager registerDeviceManager InvalidProfile, DomainManager registerDeviceManager InvalidObjectReference	OE_TC_126
OE0234		The registerDeviceManager operation shall send a DomainManagementObjectAddedEventType event to the Outgoing Domain Management event channel upon successful registration of a device manager.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_154
OE0234	C041	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_154

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0234	C042	The sourceId is the identifier attribute of the registered device manager.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_154
OE0234	C043	The sourceName is the label attribute of the registered device manager.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_154
OE0234	C044	The sourceIOR is the object reference for the registered device manager.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_154
OE0234	C045	The sourceCategory is "DEVICE_MANAGER".	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_154
OE0240		The registerDeviceManager operation shall raise the CF InvalidObjectReference exception when the input parameter deviceMgr contains an invalid reference to a DeviceManager interface.	3.1.3.2.3.6.1.5	Automated	DomainManager registerDeviceManager InvalidObjectReference	OE_TC_244
OE0241		The registerDeviceManager operation shall raise the RegisterError exception when an internal error exists which causes an unsuccessful registration.	3.1.3.2.3.6.1.5	Manual		OE_TC_126
OE0242		The registerDevice operation shall add the device indicated by the input registeringDevice parameter and the device's attributes to the domain manager, if it does not already exist.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_246
OE0244		The registerDevice operation shall write an ADMINISTRATIVE_EVENT log record to a domain manager log upon successful device registration.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_246
OE0245		The registerDevice operation shall write a FAILURE_ALARM log record to a domain manager log, when the CF InvalidProfile exception is raised.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice InvalidProfile	OE_TC_247
OE0246		The registerDevice operation shall write a FAILURE_ALARM log record to a domain manager log when the DeviceManagerNotRegistered exception is raised.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice DeviceManagerNotRegistered	OE_TC_247

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0247		The registerDevice operation shall write a FAILURE_ALARM log record to a domain manager log when the CF InvalidObjectReference exception is raised.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice InvalidObjectReference	OE_TC_247
OE0248		The registerDevice operation shall write a FAILURE_ALARM log record to a domain manager log when the RegisterError exception is raised.	3.1.3.2.3.6.2.3	Manual		OE_TC_124
OE0249		The registerDevice operation shall send a DomainManagementObjectAddedEventType event to the Outgoing Domain Management event channel, upon successful registration of a device.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_247
OE0249	C047	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_247
OE0249	C048	The sourceId is the identifier attribute of the registered device.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_247
OE0249	C049	The sourceName is the label attribute of the registered device.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_247
OE0249	C050	The sourceIOR is the object reference for the registered device.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_247
OE0249	C051	The sourceCategory is "DEVICE".	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_247
OE0255		The registerDevice operation shall raise the CF InvalidProfile exception when: 1. The device's SPD file and the SPD's referenced files do not exist, or 2. The device profile does not reference allocation properties.	3.1.3.2.3.6.2.5	Automated	DomainManager registerDevice InvalidProfile	OE_TC_248
OE0256		The registerDevice operation shall raise a DeviceManagerNotRegistered exception when the input registeredDeviceMgr parameter is not a nil reference and the referenced device manager is not registered with the domain manager.	3.1.3.2.3.6.2.5	Automated	DomainManager registerDevice DeviceManagerNotRegistered	OE_TC_248

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0257		The registerDevice operation shall raise the CF InvalidObjectReference exception when input parameters registeringDevice or registeredDeviceMgr contains an invalid reference.	3.1.3.2.3.6.2.5	Automated	DomainManager registerDevice InvalidObjectReference	OE_TC_248
OE0258		The registerDevice operation shall raise the RegisterError exception when an internal error exists which causes an unsuccessful registration.	3.1.3.2.3.6.2.5	Manual		OE_TC_124
OE0259		The installApplication operation shall verify the existence of the application's SAD file and all files upon which the SAD depends, within the domain manager's file manager.	3.1.3.2.3.6.3.3	Automated	ApplicationFactory create CreateApplicationError, ApplicationFactory create CreateApplicationRequestError, ApplicationFactory create InvalidInitConfiguration, ApplicationFactory create with deviceAssignments, ApplicationFactory identifier Attribute, ApplicationFactory softwareProfile Attribute, DomainManager installApplication InvalidFileName, DomainManager installApplication uninstallApplication	OE_TC_249

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0259	C052	The input profileFileName parameter is the absolute pathname of the application SAD.	3.1.3.2.3.6.3.3	Automated	ApplicationFactory create CreateApplicationError, ApplicationFactory create CreateApplicationRequestError, ApplicationFactory create InvalidInitConfiguration, ApplicationFactory create with deviceAssignments, ApplicationFactory identifier Attribute, ApplicationFactory softwareProfile Attribute, DomainManager installApplication InvalidFileName, DomainManager installApplication uninstallApplication	OE_TC_249
OE0260		The installApplication operation shall write an ADMINISTRATIVE_EVENT log record to a domain manager's log, upon successful application installation.	3.1.3.2.3.6.3.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_249
OE0261		The installApplication operation shall, upon unsuccessful application installation, write a FAILURE_ALARM log record to a domain manager's log.	3.1.3.2.3.6.3.3	Semi-Automated	DomainManager installApplication InvalidFileName, DomainManager installApplication InvalidProfile, DomainManager installApplication ApplicationAlreadyInstalled	OE_TC_057
OE0262		The installApplication operation shall send a DomainManagementObjectAddedEventType event to the Outgoing Domain Management event channel, upon successful installation of an application.	3.1.3.2.3.6.3.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_249
OE0262	C053	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.3.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_249



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0262	C054	The sourceId is the identifier attribute of the installed application factory.	3.1.3.2.3.6.3.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_249
OE0262	C055	The sourceName is the name attribute of the installed application factory.	3.1.3.2.3.6.3.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_249
OE0262	C056	The sourceIOR is the object reference for the installed application factory.	3.1.3.2.3.6.3.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_249
OE0262	C057	The sourceCategory is "APPLICATION_FACTORY".	3.1.3.2.3.6.3.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_249
OE0268		The installApplication operation shall raise the ApplicationInstallationError exception when the installation of the application file(s) was not successfully completed.	3.1.3.2.3.6.3.5	Manual		OE_TC_057
OE0269		The installApplication operation shall raise the CF InvalidFileName exception when the input SAD file or any of the SAD's referenced filenames do not exist in the file system identified by the absolute path of the input profileFileName parameter.	3.1.3.2.3.6.3.5	Automated	DomainManager installApplication InvalidFileName	OE_TC_132
OE0270		The installApplication operation shall log a FAILURE_ALARM log record to a domain manager's Log with a message consisting of "installApplication::invalid file is xxx", where "xxx" is the input or referenced filename, when the CF InvalidFileName exception occurs.	3.1.3.2.3.6.3.5	Automated	DomainManager installApplication InvalidFileName	OE_TC_132
OE0271		The installApplication operation shall raise the CF InvalidProfile exception when any referenced property definition is missing.	3.1.3.2.3.6.3.5	Automated	DomainManager installApplication InvalidProfile	OE_TC_250
OE0272		The installApplication operation shall write a FAILURE_ALARM log record to a domain manager's log when the CF InvalidProfile exception is raised.	3.1.3.2.3.6.3.5	Automated	DomainManager installApplication InvalidProfile	OE_TC_250

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0272	C058	The value of the logData attribute of this record is "installApplication::invalid Profile is yyy", where "yyy" is the input or referenced file name.	3.1.3.2.3.6.3.5	Automated	DomainManager installApplication InvalidProfile	OE_TC_250
OE0273		The unregisterDeviceManager operation shall unregister a DeviceManager component from the DomainManager.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_251
OE0274		The unregisterDeviceManager operation shall release all device(s) and service(s) associated with the device manager that is being unregistered.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_053
OE0275		The unregisterDeviceManager operation shall disconnect the established connections (including those made to the CORBA Event Service event channels) of the unregistering device manager as well as for its registered devices and services.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_251
OE0276		The unregisterDeviceManager operation shall unmount all device manager's file systems from its file manager.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_251
OE0277		The unregisterDeviceManager operation shall, upon the successful unregistration of a device manager, write an ADMINISTRATIVE_EVENT log record to a domain manager's log.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_251
OE0278		The unregisterDeviceManager operation shall, upon unsuccessful unregistration of a device manager, write a FAILURE_ALARM log record to a domain manager's log.	3.1.3.2.3.6.4.3	Automated	DomainManager unregisterDeviceManager InvalidObjectReference	OE_TC_128
OE0279		The unregisterDeviceManager operation shall send a DomainManagementObjectRemovedEventType event to the Outgoing Domain Management event channel, upon successful unregistration of a device manager.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_158

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0279	C059	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_158
OE0279	C060	The sourceId is the identifier attribute of the unregistered device manager.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_158
OE0279	C061	The sourceName is the label attribute of the unregistered device manager.	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_158
OE0279	C062	The sourceCategory is "DEVICE_MANAGER".	3.1.3.2.3.6.4.3	Automated	DomainManager registerDeviceManager unregisterDeviceManager	OE_TC_158
OE0284		The unregisterDeviceManager operation shall raise the CF InvalidObjectReference when the input deviceMgr parameter contains an invalid reference to a DeviceManager interface.	3.1.3.2.3.6.4.5	Automated	DomainManager unregisterDeviceManager InvalidObjectReference	OE_TC_252
OE0285		The unregisterDeviceManager operation shall raise the UnregisterError exception when an internal error exists which causes an unsuccessful unregistration.	3.1.3.2.3.6.4.5	Manual		OE_TC_128
OE0286		The unregisterDevice operation shall remove a device entry from the domain manager.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0287		The unregisterDevice operation shall release (client-side CORBA release) the unregisteringDevice from the domain manager.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0288		The unregisterDevice operation shall disconnect the established connections (including those made to the CORBA Event Service event channels) of the unregistering device.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0289		The unregisterDevice operation shall, upon the successful unregistration of a device, write an ADMINISTRATIVE_EVENT log record to a domain manager's log.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0290		The unregisterDevice operation shall, upon unsuccessful unregistration of a device, write a FAILURE_ALARM log record to a domain manager's log.	3.1.3.2.3.6.5.3	Semi-Automated	DomainManager unregisterDevice InvalidObjectReference	OE_TC_129
OE0291		The unregisterDevice operation shall send a DomainManagementObjectRemovedEventType event to the Outgoing Domain Management event channel, upon successful unregistration of a device.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0291	C063	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0291	C064	The sourceId is the identifier attribute of the unregistered device.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0291	C065	The sourceName is the label attribute of the unregistered device.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0291	C066	The sourceCategory is "DEVICE".	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0296		The unregisterDevice operation shall raise the CF InvalidObjectReference exception when the input parameter contains an invalid reference to a Device interface.	3.1.3.2.3.6.5.5	Automated	DomainManager unregisterDevice InvalidObjectReference	OE_TC_129
OE0297		The unregisterDevice operation shall raise the UnregisterError exception when an internal error exists which causes an unsuccessful unregistration.	3.1.3.2.3.6.5.5	Manual		OE_TC_129

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0299		The uninstallApplication operation shall make the ApplicationFactory unavailable from the domain manager (i.e. its services no longer provided for the application).	3.1.3.2.3.6.6.3	Automated	ApplicationFactory create CreateApplicationError, ApplicationFactory create CreateApplicationRequestError, ApplicationFactory create InvalidInitConfiguration, ApplicationFactory create with deviceAssignments, ApplicationFactory identifier Attribute, ApplicationFactory softwareProfile Attribute, DomainManager installApplication uninstallApplication	OE_TC_255
OE0300		The uninstallApplication operation shall, upon successful uninstall of an application, write an ADMINISTRATIVE_EVENT log record to a domain manager's log.	3.1.3.2.3.6.6.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_255
OE0301		The uninstallApplication operation shall, upon unsuccessful uninstall of an application, write a FAILURE_ALARM log record to a domain manager's log.	3.1.3.2.3.6.6.3	Automated	DomainManager uninstallApplication InvalidIdentifier	OE_TC_122
OE0303		The uninstallApplication operation shall send a DomainManagementObjectRemovedEventType event to the Outgoing Domain Management event channel, upon the successful uninstallation of an application.	3.1.3.2.3.6.6.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_255
OE0303	C067	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.6.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_255
OE0303	C068	The sourceId is the identifier attribute of the uninstalled application factory.	3.1.3.2.3.6.6.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_255
OE0303	C069	The sourceName is the name attribute of the uninstalled application factory.	3.1.3.2.3.6.6.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_255

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0303	C070	The sourceCategory is "APPLICATION_FACTORY".	3.1.3.2.3.6.6.3	Automated	DomainManager installApplication uninstallApplication	OE_TC_255
OE0308		The uninstallApplication operation shall raise the InvalidIdentifier exception when the ApplicationId is invalid.	3.1.3.2.3.6.6.5	Automated	DomainManager uninstallApplication InvalidIdentifier	OE_TC_255
OE0309		The uninstallApplication operation shall raise the ApplicationUninstallationError exception when an internal error causes an unsuccessful uninstallation of the application.	3.1.3.2.3.6.6.5	Manual		OE_TC_122
OE0310		The registerService operation shall verify the input registeringService and registeredDeviceMgr are valid object references.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService InvalidObjectReference	OE_TC_257
OE0311		The registerService operation shall verify the input registeredDeviceMgr has been previously registered with the domain manager.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService DeviceManagerNotRegistered	OE_TC_257
OE0312		The registerService operation shall add the registeringService's object reference and the registeringService's name to the domain manager, if the name for the type of service being registered does not exist within the domain manager.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_257
OE0313		The registerService operation shall return without exception and not register a new service when that service, indicated by the input registeringService parameter, has the same name and type as a previously registered service and the reference to the registered service refers to an existing object.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_257
OE0314		The registerService operation shall associate the input registeringService parameter with the input registeredDeviceMgr parameter in the domain manager, when the registeredDeviceMgr parameter indicates a device manager that is registered with the domain manager.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_065

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0315		The registerService operation shall establish any pending connections from previously registered device managers when the registering service completes these connections.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_238
OE0316		The registerService operation shall, upon successful service registration, write an ADMINISTRATIVE_EVENT log record to a domain manager's log.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256
OE0317		The registerService operation shall, upon unsuccessful service registration, write a FAILURE_ALARM log record to a domain manager's log.	3.1.3.2.3.6.7.3	Semi-Automated	DomainManager registerService InvalidObjectReference, DomainManager registerService DeviceManagerNotRegistered	OE_TC_127
OE0318		The registerService operation shall send a DomainManagementObjectAddedEventType event to the Outgoing Domain Management event channel, upon successful registration of a service.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256
OE0318	C072	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256
OE0318	C073	The sourceId is the identifier attribute of the componentinstantiation element associated with the registered service.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256
OE0318	C074	The sourceName is the input name parameter for the registering service.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256
OE0318	C075	The sourceIOR is the object reference for the registered service.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256
OE0318	C076	The sourceCategory is "SERVICE".	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256
OE0324		The registerService operation shall raise a DeviceManagerNotRegistered exception when the input registeredDeviceMgr parameter is not a nil reference and is not registered with the domain manager.	3.1.3.2.3.6.7.5	Automated	DomainManager registerService DeviceManagerNotRegistered	OE_TC_257

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0325		The registerService operation shall raise the CF InvalidObjectReference exception when input parameters registeringService or registeredDeviceMgr contains an invalid reference.	3.1.3.2.3.6.7.5	Automated	DomainManager registerService InvalidObjectReference	OE_TC_257
OE0326		The registerService operation shall raise the RegisterError exception when an internal error exists which causes an unsuccessful registration.	3.1.3.2.3.6.7.5	Manual		OE_TC_127
OE0327		The unregisterService operation shall remove the unregisteringService entry specified by the input name parameter from the domain manager.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_051
OE0328		The unregisterService operation shall release (client-side CORBA release) the unregisteringService from the domain manager.	3.1.3.2.3.6.8.3	Manual		OE_TC_052
OE0329		The unregisterService operation shall, upon the successful unregistration of a service, write an ADMINISTRATIVE_EVENT log record to a domain manager's log.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258
OE0330		The unregisterService operation shall, upon unsuccessful unregistration of a service, write a FAILURE_ALARM log record to a domain manager's log.	3.1.3.2.3.6.8.3	Automated	DomainManager unregisterService InvalidObjectReference	OE_TC_258
OE0331		The unregisterService operation shall send a DomainManagementObjectRemovedEventType event to the Outgoing Domain Management event channel, upon successful unregistration of a service.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258
OE0331	C077	The producerId is the identifier attribute of the domain manager.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258
OE0331	C078	The sourceId is the identifier attribute of the componentinstantiation element associated with the unregistered service.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258
OE0331	C079	The sourceName is the input name parameter for the unregistering service.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0331	C080	The sourceCategory is "SERVICE".	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258
OE0336		The unregisterService operation shall raise the CF InvalidObjectReference exception when the input parameter contains an invalid reference to a service interface.	3.1.3.2.3.6.8.5	Automated	DomainManager unregisterService InvalidObjectReference	OE_TC_258
OE0337		The unregisterService operation shall raise the UnregisterError exception when an internal error exists which causes an unsuccessful unregistration.	3.1.3.2.3.6.8.5	Manual		OE_TC_055
OE0338		The registerWithEventChannel operation shall connect the object identified by the input registeringObject parameter to an event channel as specified by the input eventChannelName parameter.	3.1.3.2.3.6.9.3	Automated	DomainManager registerWithEventChannel unregisterFromEventChannel	OE_TC_259
OE0339		The registerWithEventChannel operation shall raise the CF InvalidObjectReference exception when the input registeringObject parameter contains an invalid reference to a CosEventComm::PushConsumer interface.	3.1.3.2.3.6.9.5	Automated	DomainManager registerWithEventChannel InvalidObjectReference	OE_TC_259
OE0340		The registerWithEventChannel operation shall raise the InvalidEventChannelName exception when the input eventChannelName parameter contains an invalid event channel name.	3.1.3.2.3.6.9.5	Automated	DomainManager registerWithEventChannel InvalidEventChannelName	OE_TC_259
OE0341		The registerWithEventChannel operation shall raise AlreadyConnected exception when the input parameter contains a connection to the event channel for the input registeringId parameter.	3.1.3.2.3.6.9.5	Automated	DomainManager registerWithEventChannel AlreadyConnected	OE_TC_259
OE0342		The unregisterFromEventChannel operation shall disconnect a registered component from the event channel as identified by the input parameters.	3.1.3.2.3.6.10.3	Automated	DomainManager registerWithEventChannel unregisterFromEventChannel	OE_TC_245

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0343		The unregisterFromEventChannel operation shall raise the InvalidEventChannelName exception when the input eventChannelName parameter contains an invalid reference to an event channel.	3.1.3.2.3.6.10.5	Automated	DomainManager unregisterFromEventChannel InvalidEventChannelName	OE_TC_245
OE0344		The unregisterFromEventChannel operation shall raise the NotConnected exception when the input parameter unregisteringId parameter is not connected to specified input event channel.	3.1.3.2.3.6.10.5	Automated	DomainManager unregisterFromEventChannel NotConnected	OE_TC_245
OE0345		The readonly usageState attribute shall contain the device's usage state (IDLE, ACTIVE, or BUSY).	3.1.3.3.1.4.1	Automated	Device usageState Attribute	OE_TC_079
OE0381		The device shall send a StateChangeEvent event to the Incoming Domain Management event channel, whenever the usageState attribute changes.	3.1.3.3.1.4.1	Manual		OE_TC_079
OE0381	C082	The producerId field is the identifier attribute of the device.	3.1.3.3.1.4.1	Manual		OE_TC_079
OE0381	C083	The sourceId field is the identifier attribute of the device.	3.1.3.3.1.4.1	Manual		OE_TC_079
OE0381	C084	The stateChangeCategory field is "ADMINISTRATIVE_STATE_EVENT".	3.1.3.3.1.4.1	Manual		OE_TC_079
OE0381	C085	The stateChangeFrom field is the value of the adminState attribute before the state change.	3.1.3.3.1.4.1	Manual		OE_TC_079
OE0381	C086	The stateChangeTo field is the value of the adminState attribute after the state change.	3.1.3.3.1.4.1	Manual		OE_TC_079
OE0386		The adminState attribute shall contain the device's admin state value.	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_080

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0387		The adminState attribute shall only allow the setting of LOCKED and UNLOCKED values, where setting "LOCKED" is only effective when the adminState attribute value is UNLOCKED, and setting "UNLOCKED" is only effective when the adminState attribute value is LOCKED or SHUTTING_DOWN.	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_080
OE0388		The adminState attribute, upon being commanded to be LOCKED, shall transition from the UNLOCKED to the SHUTTING_DOWN state and set the adminState to LOCKED for its entire aggregation of devices (if it has any).	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_080
OE0389		The adminState shall then transition to the LOCKED state when the device's usageState is IDLE and its entire aggregation of devices are LOCKED.	3.1.3.3.1.4.2	Manual		OE_TC_080
OE0390		The device shall send a StateChangeEvent event to the Incoming Domain Management event channel, whenever the adminState attribute changes.	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_088
OE0390	C087	The producerId field is the identifier attribute of the device.	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_088
OE0390	C088	The sourceId field is the identifier attribute of the device.	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_088
OE0390	C089	The stateChangeCategory field is "ADMINISTRATIVE_STATE_EVENT".	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_088
OE0390	C090	The stateChangeFrom field is the value of the adminState attribute before the state change.	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_088
OE0390	C091	The stateChangeTo field is the value of the adminState attribute after the state change.	3.1.3.3.1.4.2	Automated	Device adminState Attribute	OE_TC_088
OE0395		The readonly operationalState attribute shall contain the device's operational state (ENABLED or DISABLED).	3.1.3.3.1.4.3	Automated	Device operationalState Attribute	OE_TC_218

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0400		The device shall send a StateChangeEvent event to the Incoming Domain Management event channel, whenever the operationalState attribute changes.	3.1.3.3.1.4.3	Manual		OE_TC_058
OE0400	C092	The producerId field is the identifier attribute of the device.	3.1.3.3.1.4.3	Manual		OE_TC_058
OE0400	C093	The sourceId field is the identifier attribute of the device.	3.1.3.3.1.4.3	Manual		OE_TC_058
OE0400	C094	The stateChangeCategory field is "OPERATIONAL_STATE_EVENT".	3.1.3.3.1.4.3	Manual		OE_TC_058
OE0400	C095	The stateChangeFrom field is the value of the operationalState attribute before the state change.	3.1.3.3.1.4.3	Manual		OE_TC_058
OE0400	C096	The stateChangeTo field is the value of the operationalState attribute after the state change.	3.1.3.3.1.4.3	Manual		OE_TC_058
OE0401		The readonly softwareProfile attribute shall contain a profile element (Profile Descriptor) with a file reference to the SPD file.	3.1.3.3.1.4.4	Automated	Device SoftwareProfile Attribute	OE_TC_218
OE0403		The readonly label attribute shall contain the device's label.	3.1.3.3.1.4.5	Automated	Device Label Attribute	OE_TC_218
OE0404		The readonly compositeDevice attribute shall contain the object reference of the aggregate device when this device is a parent device.	3.1.3.3.1.4.6	Automated	Device compositeDevice Attribute	OE_TC_218
OE0405		The allocateCapacity operation shall reduce the current capacities of the device based upon the input capacities parameter, when the device's adminState is UNLOCKED, device's operationalState is ENABLED, and device's usageState is not BUSY.	3.1.3.3.1.5.1.3	Manual		OE_TC_081
OE0406		The allocateCapacity operation shall set the Device's usageState attribute to BUSY, when the device determines that it is not possible to allocate any further capacity.	3.1.3.3.1.5.1.3	Manual		OE_TC_082

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0407		The allocateCapacity operation shall set the usageState attribute to ACTIVE, when capacity is being used and any capacity is still available for allocation (reference Figure 3-22).	3.1.3.3.1.5.1.3	Manual		OE_TC_004
OE0408		The allocateCapacity operation shall return TRUE, if the capacities have been allocated, or FALSE, if not allocated.	3.1.3.3.1.5.1.4	Manual		OE_TC_083
OE0409		The allocateCapacity operation shall raise the InvalidCapacity exception, when the input capacities parameter contains invalid properties or when attributes of those CF Properties contain an unknown id or a value of the wrong datatype.	3.1.3.3.1.5.1.5	Automated	Device allocateCapacity InvalidCapacity	OE_TC_229
OE0410		The allocateCapacity operation shall raise the InvalidState exception, when the Device's adminState is not UNLOCKED or operationalState is DISABLED.	3.1.3.3.1.5.1.5	Automated	Device allocateCapacity InvalidState	OE_TC_229
OE0411		The deallocateCapacity operation shall adjust the current capacities of the device based upon the input capacities parameter.	3.1.3.3.1.5.2.3	Manual		OE_TC_084
OE0412		The deallocateCapacity operation shall set the usageState attribute to ACTIVE when, after adjusting capacities, any of the device's capacities are still being used.	3.1.3.3.1.5.2.3	Manual		OE_TC_085
OE0413		The deallocateCapacity operation shall set the usageState attribute to IDLE when, after adjusting capacities, none of the device's capacities are still being used.	3.1.3.3.1.5.2.3	Manual		OE_TC_085
OE0414		The deallocateCapacity operation shall set the adminState attribute to LOCKED as specified in 3.1.3.2.4.4.2.	3.1.3.3.1.5.2.3	Manual		OE_TC_085

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0415		The deallocateCapacity operation shall raise the InvalidCapacity exception, when the capacity ID is invalid or the capacity value is the wrong type.	3.1.3.3.1.5.2.5	Automated	Device deallocateCapacity InvalidCapacity	OE_TC_086
OE0417		The deallocateCapacity operation shall raise the InvalidState exception, when the device's adminState is LOCKED or operationalState is DISABLED.	3.1.3.3.1.5.2.5	Automated	Device deallocateCapacity InvalidState	OE_TC_230
OE0418		The releaseObject operation shall call the releaseObject operation on all those devices that are contained within the AggregateDevice devices attribute, when this device is a parent device.	3.1.3.3.1.5.3.3	Automated	Device releaseObject	OE_TC_219
OE0419		The releaseObject operation shall assign the LOCKED state to the Device adminState attribute, when the Device adminState attribute is UNLOCKED.	3.1.3.3.1.5.3.3	Automated	Device releaseObject	OE_TC_219
OE0420		The releaseObject operation shall cause the device to be unavailable and released from the CORBA environment when the Device adminState attribute transitions to LOCKED.	3.1.3.3.1.5.3.3	Automated	Device releaseObject	OE_TC_087
OE0421		The releaseObject operation shall cause the removal of the device from the Device compositeDevice attribute, when this device is a child device.	3.1.3.3.1.5.3.3	Automated	Device releaseObject	OE_TC_219
OE0422		The releaseObject operation shall unregister its device from its device manager.	3.1.3.3.1.5.3.3	Automated	Device releaseObject	OE_TC_219
OE0423		The releaseObject operation shall raise the ReleaseError exception when releaseObject is not successful in releasing a logical device due to internal processing errors that occurred within the device being released.	3.1.3.3.1.5.3.5	Manual		OE_TC_089
OE0424		The error number shall indicate a CF ErrorNumberType.	3.1.3.3.2.3.3	Automated	Device load LoadFail	OE_TC_130

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0426		The load operation shall load the file identified by the input filename parameter on the device based upon the input loadKind parameter.	3.1.3.3.2.5.1.3	Semi-Automated	Device load execute terminate unload	OE_TC_286
OE0427		The load operation shall support the load types as stated in the device's software profile LoadType allocation properties.	3.1.3.3.2.5.1.3	Semi-Automated	Device load execute terminate unload	OE_TC_286
OE0429		The load operation shall raise the InvalidState exception if upon entry the Device's adminState attribute is either LOCKED or SHUTTING_DOWN or its operationalState attribute is DISABLED.	3.1.3.3.2.5.1.5	Semi-Automated	Device load InvalidState	OE_TC_220
OE0430		The load operation shall raise the InvalidLoadKind exception when the input loadKind parameter is not supported.	3.1.3.3.2.5.1.5	Automated	Device load InvalidLoadKind	OE_TC_220
OE0431		The load operation shall raise the CF InvalidFileName exception when the file designated by the input filename parameter cannot be found.	3.1.3.3.2.5.1.5	Automated	Device load InvalidFileName	OE_TC_133
OE0432		The load operation shall raise the LoadFail exception when an attempt to load the device is unsuccessful.	3.1.3.3.2.5.1.5	Automated	Device load LoadFail	OE_TC_130
OE0434		The unload operation shall unload the file identified by the input fileName parameter from the device when the number of unload requests matches the number of load requests for the indicated file.	3.1.3.3.2.5.2.3	Semi-Automated	Device load execute terminate unload	OE_TC_228
OE0435		The unload operation shall raise the InvalidState exception if upon entry the device's adminState attribute is LOCKED or its operationalState attribute is DISABLED.	3.1.3.3.2.5.2.5	Automated	Device unload InvalidState	OE_TC_228
OE0436		The unload operation shall raise the CF InvalidFileName exception when the file designated by the input filename parameter cannot be found.	3.1.3.3.2.5.2.5	Automated	Device unload InvalidFileName	OE_TC_134
OE0438		The errorNumber parameter shall indicate a CF ErrorNumberType value.	3.1.3.3.3.3.1	Automated	Device Terminate InvalidProcess	OE_TC_145

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0444		The error number shall indicate a CF ErrorNumberType value.	3.1.3.3.3.3.8	Automated	Device execute ExecuteFail	OE_TC_222
OE0445		The execute operation shall execute the function or file identified by the input name parameter using the input parameters and options parameters.	3.1.3.3.3.5.1.3	Automated	Device load execute terminate unload	OE_TC_227
OE0446		The execute operation shall convert the input parameters (id/value string pairs) parameter to the standard argv of the POSIX exec family of functions, where argv(0) is the function name.	3.1.3.3.3.5.1.3	Automated	Device load execute terminate unload	OE_TC_227
OE0447		The execute operation shall map the input parameters parameter to argv starting at index 1 as follows, argv (1) maps to input parameters (0) id and argv (2) maps to input parameters (0) value and so forth.	3.1.3.3.3.5.1.3	Automated	Device load execute terminate unload	OE_TC_227
OE0448		The execute operation shall use these options, when specified, to set the operating system's process/thread stack size and priority, for the executable image of the given input name parameter.	3.1.3.3.3.5.1.3	Automated	Device load execute terminate unload	OE_TC_227
OE0449		The execute operation shall return a unique process ID for the process that it created.	3.1.3.3.3.5.1.4	Automated	Device load execute terminate unload	OE_TC_227
OE0450		The execute operation shall raise the InvalidState exception if upon entry the device's adminState attribute is either LOCKED or SHUTTING_DOWN or its operationalState attribute is DISABLED.	3.1.3.3.3.5.1.5	Automated	Device execute InvalidState	OE_TC_222
OE0451		The execute operation shall raise the InvalidFunction exception when the function indicated by the input name parameter does not exist for the device.	3.1.3.3.3.5.1.5	Automated	Device execute InvalidFileName InvalidFunction	OE_TC_222
OE0452		The execute operation shall raise the CF InvalidFileName exception when the file name indicated by the input name parameter does not exist for the device.	3.1.3.3.3.5.1.5	Automated	Device execute InvalidFileName InvalidFunction	OE_TC_135



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0453		The execute operation shall raise the InvalidParameters exception when the input parameter ID or value attributes are not valid strings.	3.1.3.3.3.5.1.5	Automated	Device execute InvalidParameters	OE_TC_222
OE0454		The execute operation shall raise the InvalidOptions exception when the input options parameter does not comply with sections 3.1.3.3.3.6 STACK_SIZE_ID and 3.1.3.3.3.7 PRIORITY_ID.	3.1.3.3.3.5.1.5	Automated	Device execute InvalidOptions	OE_TC_222
OE0455		The execute operation shall raise the ExecuteFail exception when the operating system "execute" function for the device is not successful.	3.1.3.3.3.5.1.5	Automated	Device execute ExecuteFail	OE_TC_222
OE0456		The terminate operation shall terminate the execution of the process/thread designated by the processId input parameter on the device.	3.1.3.3.3.5.2.3	Automated	Device load execute terminate unload	OE_TC_223
OE0457		The terminate operation shall raise the InvalidState exception if upon entry the device's adminState attribute is LOCKED or its operationalState attribute is DISABLED.	3.1.3.3.3.5.2.5	Automated	Device Terminate InvalidState	OE_TC_223
OE0458		The terminate operation shall raise the InvalidProcess exception when the process Id does not exist for the device.	3.1.3.3.3.5.2.5	Automated	Device Terminate InvalidProcess	OE_TC_145
OE0459		The readonly devices attribute shall contain a list of devices that have been added to this device or a sequence length of zero if the device has no aggregation relationships with other devices.	3.1.3.3.4.4.1	Manual		OE_TC_022
OE0460		The addDevice operation shall add the input associatedDevice parameter to the AggregateDevice's devices attribute when the associatedDevice does not exist in the devices attribute.	3.1.3.3.4.5.1.3	Manual		OE_TC_023

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0461		The addDevice operation shall write a FAILURE_ALARM log record, upon unsuccessful adding of an associatedDevice to the AggregateDevice's devices attribute.	3.1.3.3.4.5.1.3	Manual		OE_TC_025
OE0462		The addDevice operation shall raise the CF InvalidObjectReference when the input associatedDevice parameter is a nil CORBA object reference.	3.1.3.3.4.5.1.5	Manual		OE_TC_027
OE0463		The removeDevice operation shall remove the input associatedDevice parameter from the AggregateDevice's devices attribute.	3.1.3.3.4.5.2.3	Manual		OE_TC_028
OE0464		The removeDevice operation shall write a FAILURE_ALARM log record, upon unsuccessful removal of the associatedDevice from the AggregateDevice devices attribute.	3.1.3.3.4.5.2.3	Manual		OE_TC_029
OE0465		The removeDevice operation shall raise the CF InvalidObjectReference when the input associatedDevice parameter is a nil CORBA object reference or does not exist in the AggregateDevice devices attribute.	3.1.3.3.4.5.2.5	Manual		OE_TC_030
OE0466		The readonly identifier attribute shall contain the instance-unique identifier for a device manager.	3.1.3.2.4.4.1	Manual		OE_TC_034
OE0467		The identifier shall be identical to the deviceconfiguration element id attribute of the device manager's Device Configuration Descriptor (DCD) file.	3.1.3.2.4.4.1	Automated	DeviceManager identifier Attribute	OE_TC_034
OE0468		The readonly label attribute shall contain the device manager's label.	3.1.3.2.4.4.2	Automated	DeviceManager label Attribute	OE_TC_224
OE0469		The readonly fileSys attribute shall contain the FileSystem associated with this device manager.	3.1.3.2.4.4.3	Automated	DeviceManager fileSys Attribute	OE_TC_224

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0470		The readonly deviceConfigurationProfile attribute shall contain a profile element (Profile Descriptor) with a file reference to the device manager's Device Configuration Descriptor (DCD) file.	3.1.3.2.4.4.4	Automated	DeviceManager deviceConfigurationProfile Attribute	OE_TC_224
OE0471		The readonly registeredDevices attribute shall contain a list of devices that have registered with this device manager or a sequence length of zero if no devices have registered with the device manager.	3.1.3.2.4.4.5	Automated	DeviceManager registeredDevices Attribute	OE_TC_224
OE0472		The readonly registeredServices attribute shall contain a list of services that have registered with this device manager or a sequence length of zero if no services have registered with the device manager.	3.1.3.2.4.4.6	Automated	DeviceManager registeredServices Attribute	OE_TC_224
OE0473		The device manager upon start up shall register itself with a domain manager.	3.1.3.2.4.5	Manual		OE_TC_033
OE0474		A device manager shall use the information in the device manager's DCD for determining: 1. Services to be deployed for this device manager (for example, log(s)), 2. Devices to be created for this device manager (when the DCD deployondevice element is not specified then the DCD componentinstantiation element is deployed on the same hardware device as the device manager), 3. Devices to be deployed on (executing on) another device, 4. Devices to be aggregated to another device, 5. Mount point names for file systems, 6. The DeviceManager's identifier attribute value which is the DCD's id attribute value, and 7. The DeviceManager's label attribute value which is the DCD's name attribute value.	3.1.3.2.4.5	Semi-Automated	DeviceManager registeredServices Attribute, DeviceManager registeredDevices Attribute, DeviceManager fileSys Attribute, DeviceManager identifier Attribute, DeviceManager label Attribute	OE_TC_034

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0475		The device manager shall create FileSystem components implementing the FileSystem interface for each OS file system.	3.1.3.2.4.5	Automated	DeviceManager fileSys Attribute	OE_TC_283
OE0476		If multiple file systems are to be created, the device manager shall mount created file systems to a FileManager component (widened to a FileSystem through the FileSys attribute).	3.1.3.2.4.5	Semi-Automated	DeviceManager fileSys Attribute	OE_TC_283
OE0476	C109	The mount points used for the created file systems are identical to the values identified in the filesystemnames element of the device manager's Device Configuration Descriptor.	3.1.3.2.4.5	Manual		OE_TC_283

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0478		The device manager shall supply execute operation parameters for a device consisting of: 1. Device manager IOR - The ID is "DEVICE_MGR_IOR" and the value is a string that is the DeviceManager stringified IOR. 2. Profile Name - The ID is "PROFILE_NAME" and the value is a CORBA string that is the full mounted file system file path name. 3. Device Identifier - The ID is "DEVICE_ID" and the value is a string that corresponds to the DCD component instantiation id attribute. 4. Device Label - The ID is "DEVICE_LABEL" and the value is a string that corresponds to the DCD component instantiation usage element. This parameter is only used when the DCD component instantiation usage element is specified. 5. Composite Device IOR - The ID is "Composite_DEVICE_IOR" and the value is a string that is an AggregateDevice stringified IOR. This parameter is only used when the DCD component instantiation element represents the child device of another component instantiation element. 6. The execute ("execparam") properties as specified in the DCD for a component instantiation element.	3.1.3.2.4.5	Automated	DeviceManager Execute Parameters For CompositeDevice, DeviceManager Execute Parameters For Device	OE_TC_225
OE0479		The device manager shall pass the component instantiation element "execparam" properties that have values as parameters.	3.1.3.2.4.5	Automated	DeviceManager Execute Parameters For CompositeDevice, DeviceManager Execute Parameters For Device	OE_TC_225
OE0480		The device manager shall pass "execparam" parameters' IDs and values as string values.	3.1.3.2.4.5	Automated	DeviceManager Execute Parameters For CompositeDevice, DeviceManager Execute Parameters For Device	OE_TC_225

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0481		The device manager shall use the componentinstantiation element's SPD implementation code's stacksize and priority elements, when specified, for the execute operation options parameters.	3.1.3.2.4.5	Automated	DeviceManager Execute Parameters For CompositeDevice, DeviceManager Execute Parameters For Device	OE_TC_285
OE0482		The device manager shall initialize and then configure logical devices that are started by the device manager, after they have successfully registered with the device manager.	3.1.3.2.4.5	Automated	DeviceManager Execute Parameters For CompositeDevice, DeviceManager Execute Parameters For Device	OE_TC_285
OE0483		The device manager shall configure a DCD's componentinstantiation element provided the componentinstantiation element has "configure" readwrite or writeonly properties with values.	3.1.3.2.4.5	Automated	DeviceManager Execute Parameters For CompositeDevice, DeviceManager Execute Parameters For Device	OE_TC_285
OE0484		If a service is deployed by the device manager, the device manager shall supply execute operation parameters consisting of: 1. Device manager IOR - The ID is "DEVICE_MGR_IOR" and the value is a string that is the DeviceManager stringified IOR. 2. Service Name - The ID is "SERVICE_NAME" and the value is a string that corresponds to the DCD componentinstantiation usagename element. 3. The execute ("execparam") properties as specified in the DCD for a componentinstantiation element.	3.1.3.2.4.5	Automated	DeviceManager Execute Parameters For Services	OE_TC_231
OE0485		The registerDevice operation shall add the input registeringDevice to the DeviceManager registeredDevices attribute when the input registeringDevice does not already exist in the registeredDevices attribute.	3.1.3.2.4.6.1.3	Automated	DeviceManager registerDevice unregisterDevice	OE_TC_260

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0486		The registerDevice operation shall register the registeringDevice with the domain manager when the device manager has already registered to the domain manager and the registeringDevice has been successfully added to the DeviceManager registeredDevices attribute.	3.1.3.2.4.6.1.3	Automated	DeviceManager registerDevice unregisterDevice	OE_TC_260
OE0487		The registerDevice operation shall write a FAILURE_ALARM log record to a domain manager's log, upon unsuccessful registration of a device to the DeviceManager registeredDevices attribute.	3.1.3.2.4.6.1.3	Automated	DeviceManager registerDevice InvalidObjectReference	OE_TC_260
OE0488		The registerDevice operation shall raise the CF InvalidObjectReference when the input registeringDevice is a nil CORBA object reference.	3.1.3.2.4.6.1.5	Automated	DeviceManager registerDevice InvalidObjectReference	OE_TC_260
OE0489		The unregisterDevice operation shall remove the input registeredDevice from the DeviceManager registeredDevices attribute.	3.1.3.2.4.6.2.3	Automated	DeviceManager registerDevice unregisterDevice	OE_TC_261
OE0490		The unregisterDevice operation shall unregister the input registeredDevice from the domain manager when the input registeredDevice is registered with the device manager and the device manager is not shutting down.	3.1.3.2.4.6.2.3	Automated	DeviceManager registerDevice unregisterDevice	OE_TC_261
OE0491		The unregisterDevice operation shall write a FAILURE_ALARM log record, when it cannot successfully remove a registeredDevice from the DeviceManager registeredDevices attribute.	3.1.3.2.4.6.2.3	Automated	DeviceManager unregisterDevice InvalidObjectReference	OE_TC_261
OE0492		The unregisterDevice operation shall raise the CF InvalidObjectReference when the input registeredDevice is a nil CORBA object reference or does not exist in the DeviceManager's registeredDevices attribute.	3.1.3.2.4.6.2.5	Automated	DeviceManager unregisterDevice InvalidObjectReference	OE_TC_261

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0493		The registerService operation shall add the input registeringService to the DeviceManager registeredServices attribute when the input registeringService does not already exist in the registeredServices attribute.	3.1.3.2.4.6.3.3	Automated	DeviceManager registerService unregisterService	OE_TC_262
OE0494		The registerService operation shall register the registeringService with the domain manager when the device manager has already registered to the domain manager and the registeringService has been successfully added to the DeviceManager's registeredServices attribute.	3.1.3.2.4.6.3.3	Automated	DeviceManager registerService unregisterService	OE_TC_262
OE0495		The registerService operation shall write a FAILURE_ALARM log record, upon unsuccessful registration of a service to the DeviceManager registeredServices attribute.	3.1.3.2.4.6.3.3	Automated	DeviceManager registerService InvalidObjectReference	OE_TC_262
OE0496		The registerService operation shall raise the CF InvalidObjectReference exception when the input registeringService is a nil CORBA object reference.	3.1.3.2.4.6.3.5	Automated	DeviceManager registerService InvalidObjectReference	OE_TC_262
OE0497		The unregisterService operation shall remove the input registered service specified by the input name parameter from the DeviceManager::registeredServices attribute.	3.1.3.2.4.6.4.3	Automated	DeviceManager registerService unregisterService	OE_TC_263
OE0498		The unregisterService operation shall unregister the input unregistering service from the domain manager when the device manager is not in the SHUTTING_DOWN state.	3.1.3.2.4.6.4.3	Automated	DeviceManager registerService unregisterService	OE_TC_263
OE0499		The unregisterService operation shall write a FAILURE_ALARM log record, when it cannot successfully remove a registeredService from the DeviceManager registeredServices attribute.	3.1.3.2.4.6.4.3	Automated	DeviceManager unregisterService InvalidObjectReference	OE_TC_263



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0500		The unregisterService operation shall raise the CF InvalidObjectReference when the input unregistering service is a nil CORBA object reference or does not exist in the DeviceManager registeredServices attribute.	3.1.3.2.4.6.4.5	Automated	DeviceManager unregisterService InvalidObjectReference	OE_TC_263
OE0501		The shutdown operation shall unregister the device manager from the domain manager.	3.1.3.2.4.6.5.3	Automated	DeviceManager shutdown	OE_TC_264
OE0502		The shutdown operation shall perform releaseObject on all of the device manager's registered devices (DeviceManager registeredDevices attribute).	3.1.3.2.4.6.5.3	Automated	DeviceManager shutdown	OE_TC_264
OE0503		The shutdown operation shall cause the device manager to be unavailable (i.e. released from the CORBA environment and its process terminated on the OS), when all of the device manager's registered devices are unregistered from the device manager.	3.1.3.2.4.6.5.3	Automated	DeviceManager shutdown	OE_TC_264
OE0504		The getComponentImplementationId operation shall return the SPD implementation element's id attribute that matches the SPD implementation element used to create the component identified by the input componentInstantiationId parameter.	3.1.3.2.4.6.6.4	Automated	DeviceManager getComponentImplementationId	OE_TC_090
OE0505		The getComponentImplementationId operation shall return an empty string when the input componentInstantiationId parameter does not match the id attribute of any SPD implementation element used to create the component.	3.1.3.2.4.6.6.4	Automated	DeviceManager getComponentImplementationId	OE_TC_090
OE0506		Framework Services Interfaces shall be implemented using the CF IDL presented in Appendix C.	3.1.3.4	Manual		OE_TC_091
OE0507		The error number shall indicate a CF ErrorNumberType value.	3.1.3.4.1.3.1	Semi-Automated	File write IOException	OE_TC_066, OE_TC_146

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0509		The readonly fileName attribute shall contain the pathname used as the input fileName parameter of the FileSystem::create operation when the file was created .	3.1.3.4.1.4.1	Automated	File fileName Attribute	OE_TC_265
OE0510		The readonly filePointer attribute shall contain the current file position.	3.1.3.4.1.4.2	Automated	File filePointer Attribute	OE_TC_265
OE0511		The read operation shall read, from the referenced file, the number of octets specified by the input length parameter and advance the value of the filePointer attribute by the number of octets actually read.	3.1.3.4.1.5.1.3	Automated	File read write close	OE_TC_266
OE0512		The read operation shall read less than the number of octets specified in the input-length parameter, when an end of file is encountered.	3.1.3.4.1.5.1.3	Automated	File read write close	OE_TC_266
OE0513		The read operation shall return via the out Message parameter a CF OctetSequence that equals the number of octets actually read from the File.	3.1.3.4.1.5.1.4	Automated	File read write close	OE_TC_266
OE0514		If the filePointer attribute value reflects the end of the File, the read operation shall return a zero-length CF OctetSequence.	3.1.3.4.1.5.1.4	Automated	File read write close	OE_TC_266
OE0515		The read operation shall raise the IOException when a read error occurs.	3.1.3.4.1.5.1.5	Manual		OE_TC_066
OE0516		The write operation shall write data to the file referenced.	3.1.3.4.1.5.2.3	Automated	File read write close	OE_TC_267
OE0517		The write operation shall increment the filePointer attribute to reflect the number of octets written, when the operation is successful.	3.1.3.4.1.5.2.3	Automated	File filePointer Attribute	OE_TC_267
OE0518		If the write is unsuccessful, the value of the filePointer attribute shall maintain or be restored to its value prior to the write operation call.	3.1.3.4.1.5.2.3	Automated	File write IOException	OE_TC_267

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0519		The write operation shall raise the IOException when a write error occurs.	3.1.3.4.1.5.2.5	Automated	File write IOException	OE_TC_146
OE0519	C111	If the file was opened using the FileSystem::open operation with an inputread_Only parameter value of TRUE, writes to the file are considered to be in error.	3.1.3.4.1.5.2.3	Manual		OE_TC_146
OE0520		The sizeOf operation shall return the number of octets stored in the file.	3.1.3.4.1.5.3.4	Automated	File sizeOf	OE_TC_268
OE0521		The sizeOf operation shall raise the CFFileException when a file-related error occurs (e.g., file does not exist anymore).	3.1.3.4.1.5.3.5	Automated	File sizeOfFileException	OE_TC_147
OE0522		The close operation shall release any OE file resources associated with the component.	3.1.3.4.1.5.4.3	Manual		OE_TC_067
OE0523		The close operation shall make the file unavailable to the component.	3.1.3.4.1.5.4.3	Automated	File read write close	OE_TC_067
OE0524		The close operation shall raise the CFFileException when it cannot successfully close the file.	3.1.3.4.1.5.4.5	Manual		OE_TC_068
OE0525		The setFilePointer operation shall set the filePointer attribute value to the input filePointer.	3.1.3.4.1.5.5.3	Automated	File setFilePointer	OE_TC_269
OE0526		The setFilePointer operation shall raise the CFFileException when the file pointer for the referenced file cannot be set to the value of the input filePointer parameter.	3.1.3.4.1.5.5.5	Manual		OE_TC_069
OE0527		The setFilePointer operation shall raise the InvalidFilePointer exception when the value of the filePointer parameter exceeds the file size.	3.1.3.4.1.5.5.5	Automated	File setFilePointer InvalidFilePointer	OE_TC_269
OE0528		At a minimum, the file system shall support name, kind, and size information for a file.	3.1.3.4.2.3.3	semi-Automated	FileManager list	OE_TC_031
OE0529		For this property, the identifier is CREATED_TIME_ID and the value shall be an unsigned long long data type containing the number of seconds since 00:00:00 UTC, Jan. 1, 1970.	3.1.3.4.2.3.6	Manual		OE_TC_109

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0530		For this property, the identifier is MODIFIED_TIME_ID and the value shall be an unsigned long long data type containing the number of seconds since 00:00:00 UTC, Jan. 1, 1970.	3.1.3.4.2.3.7	Manual		OE_TC_110
OE0531		For this property, the identifier is LAST_ACCESS_TIME_ID and the value shall be an unsigned long long data type containing the number of seconds since 00:00:00 UTC, Jan. 1, 1970.	3.1.3.4.2.3.8	Manual		OE_TC_111
OE0532		The remove operation shall remove the plain file which corresponds to the input fileName parameter.	3.1.3.4.2.5.1.3	Automated	FileManager create remove, FileSystem create remove	OE_TC_270
OE0533		The remove operation shall raise the CF InvalidFileName exception when the input fileName parameter is not a valid absolute pathname.	3.1.3.4.2.5.1.5	Automated	FileManager remove InvalidFileName, FileSystem remove InvalidFileName	OE_TC_136
OE0534		The remove operation shall raise the CF FileException when a file-related error occurs.	3.1.3.4.2.5.1.5	semi-Automated	FileManager remove FileException	OE_TC_105
OE0535		The copy operation shall copy the source file identified by the input sourceFileName parameter to the destination file identified by the input destinationFileName parameter.	3.1.3.4.2.5.2.3	Automated	FileManager copy, FileSystem copy	OE_TC_106
OE0536		The copy operation shall raise the CF FileException exception when a file-related error occurs.	3.1.3.4.2.5.2.5	Automated	FileManager copy FileException, FileSystem copy FileException	OE_TC_148
OE0537		The copy operation shall raise the CF InvalidFileName exception when the sourceFileName or destinationFileName input parameters are not a valid absolute pathnames.	3.1.3.4.2.5.2.5	Automated	FileManager copy InvalidFileName, FileSystem copy InvalidFileName	OE_TC_137
OE0538		The exists operation shall check to see if a file exists based on the fileName parameter.	3.1.3.4.2.5.3.3	Automated	FileManager exists, FileSystem exists	OE_TC_138
OE0539		The exists operation shall return TRUE if the file exists, or FALSE if it does not.	3.1.3.4.2.5.3.4	Automated	FileManager exists, FileSystem exists	OE_TC_138

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0540		The exists operation shall raise the CF InvalidFileName exception when input fileName parameter is not a valid absolute pathname.	3.1.3.4.2.5.3.5	Automated	FileManagerexists InvalidFileName, FileSystemexists InvalidFileName	OE_TC_159
OE0542		The list operation shall support the "*" and "?" wildcard characters (used to match any sequence of characters (including null) and any single character, respectively).	3.1.3.4.2.5.4.3	semi-Automated	FileManager list	OE_TC_031
OE0543		The list operation shall return a FileInformationSequence for files that match the search pattern specified in the input pattern parameter.	3.1.3.4.2.5.4.4	semi-Automated	FileManager list	OE_TC_031
OE0545		The list operation shall raise the CF InvalidFileName exception when the input pattern parameter is not an absolute pathname or cannot be interpreted due to unexpected characters.	3.1.3.4.2.5.4.5	Automated	FileManager list InvalidFileName, FileSystemlist InvalidFileName,	OE_TC_103
OE0546		The list operation shall raise the CF FileException when a file-related error occurs.	3.1.3.4.2.5.4.5	semi-Automated	FileManager list FileException	OE_TC_104
OE0547		The create operation shall create a new File based upon the input fileName parameter.	3.1.3.4.2.5.5.3	Automated	FileManager create remove, FileSystemcreate remove	OE_TC_107
OE0548		The create operation shall return a file object reference to the opened file.	3.1.3.4.2.5.5.4	Automated	FileManager create remove, FileSystemcreate remove	OE_TC_272
OE0550		The create operation shall raise the CF FileException if the file already exists or another file error occurred.	3.1.3.4.2.5.5.5	Automated	FileManager create FileException, FileSystemcreate FileException	OE_TC_149
OE0551		The create operation shall raise the CF InvalidFileName exception when the input fileName parameter is not a valid absolute pathname.	3.1.3.4.2.5.5.5	Automated	FileManager create InvalidFileName, FileSystemcreate InvalidFileName	OE_TC_139
OE0552		The open operation shall open the file referenced by the input fileName parameter.	3.1.3.4.2.5.6.3	Automated	FileManager open, FileSystemopen	OE_TC_273
OE0555		The open operation shall open the file for write access when the input read_Only parameter is FALSE.	3.1.3.4.2.5.6.3	Automated	FileManager open, FileSystemopen	OE_TC_273

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0556		The open operation shall return a File instance on successful completion.	3.1.3.4.2.5.6.4	Automated	FileManager open, FileSystemopen	OE_TC_273
OE0558		The open operation shall raise the CF FileException if the file does not exist or another file error occurred.	3.1.3.4.2.5.6.5	Automated	FileManager open FileException, Filesystemopen FileException	OE_TC_150
OE0559		The open operation shall raise the CF InvalidFileName exception when the input fileName parameter is not a valid absolute pathname.	3.1.3.4.2.5.6.5	Automated	FileManager open InvalidFileName, FileSystemopen InvalidFileName	OE_TC_140
OE0560		The mkdir operation shall create a file system directory based on the directoryName given.	3.1.3.4.2.5.7.3	Automated	FileManager mkdir rmdir, Filesystemmkdir rmdir	OE_TC_114
OE0561		The mkdir operation shall create all parent directories required to create the directoryName path given.	3.1.3.4.2.5.7.3	semi-Automated	Filesystemmkdir rmdir	OE_TC_114
OE0562		The mkdir operation shall raise the CF FileException if the directory indicated by the input directoryName parameter already exists or if a file-related error occurred during the operation.	3.1.3.4.2.5.7.5	Automated	FileManager mkdir FileException, Filesystemmkdir FileException	OE_TC_151
OE0563		The mkdir operation shall raise the CF InvalidFileName exception when the directoryName is not a valid directory name.	3.1.3.4.2.5.7.5	Automated	FileManager mkdir InvalidFileName, Filesystemmkdir InvalidFileName	OE_TC_141
OE0564		The rmdir operation shall remove the directory identified by the input directoryName parameter.	3.1.3.4.2.5.8.3	Automated	FileManager mkdir rmdir, Filesystemmkdir rmdir	OE_TC_115
OE0565		The rmdir operation shall raise the CF FileException when the directory identified by the input directoryName parameter does not exist, the directory contains files, or an error occurs which prohibits the directory from being deleted.	3.1.3.4.2.5.8.5	Automated	FileManager rmdir FileException, Filesystemrmdir FileException	OE_TC_152
OE0566		The rmdir operation shall raise the CF InvalidFileName exception when the input directoryName parameter is not a valid path prefix.	3.1.3.4.2.5.8.5	Automated	FileManager rmdir InvalidFileName, Filesystemrmdir InvalidFileName	OE_TC_142
OE0567		The query operation shall return file system information to the calling client based upon the given fileSystemProperties' ID.	3.1.3.4.2.5.9.3	Automated	Filesystemquery	OE_TC_275

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0568		The FileSystem::query operation shall recognize and provide the designated return values for the following FileSystemProperties (section 3.1.3.4.2.3.2): 1. SIZE - an ID value of "SIZE" causes the query operation to return an unsigned long long containing the file systems size (in octets). 2. A AVAILABLE SPACE - an ID value of "A AVAILABLE SPACE" causes the query operation to return an unsigned long long containing the available space on the file system (in octets).	3.1.3.4.2.5.9.3	Automated	FileSystemquery	OE_TC_275
OE0569		The query operation shall raise the UnknownFileSystemProperties exception when the given file system property is not recognized.	3.1.3.4.2.5.9.5	Automated	FileSystemquery UnknownFileSystemProperties	OE_TC_275
OE0573		The mount operation shall associate the specified file system with the mount point referenced by the input mountPoint parameter.	3.1.3.4.3.5.1.3	Automated	FileManager mount unmount	OE_TC_276
OE0574		A mount point name shall begin with a "/" (forward slash character).	3.1.3.4.3.5.1.3	Automated	DeviceManager fileSys Attribute, FileManager getMounts, FileManager mount InvalidFileName	OE_TC_276
OE0575		The mount operation shall raise the CF InvalidFileName exception when the input mount point does not conform to the file name syntax in section 3.1.3.4.2.1.	3.1.3.4.3.5.1.5	Automated	FileManager mount InvalidFileName	OE_TC_143
OE0576		The mount operation shall raise the MountPointAlreadyExists exception when the mount point already exists in the file manager.	3.1.3.4.3.5.1.5	Automated	FileManager mount MountPointAlreadyExists	OE_TC_276
OE0577		The mount operation shall raise the InvalidFileSystem exception when the input FileSystem is a null object reference.	3.1.3.4.3.5.1.5	Automated	FileManager mount InvalidFileSystem	OE_TC_276
OE0578		The unmount operation shall remove a mounted file system from the file manager whose mounted name matches the input mountPoint name.	3.1.3.4.3.5.2.3	Automated	FileManager mount unmount	OE_TC_277

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0579		The unmount operation shall raise the NonExistentMount exception when the mount point does not exist.	3.1.3.4.3.5.2.5	Automated	FileManager unmount NonExistentMount	OE_TC_277
OE0580		The getMounts operation shall return a MountSequence that contains the file systems mounted within the file manager.	3.1.3.4.3.5.3.4	Automated	FileManager getMounts	OE_TC_278
OE0581		The FileManager interface shall support a federated, or distributed, file system that may span multiple FileSystem components.	3.1.3.4.3.5.4	Automated	FileManager Distributed FileSystem	OE_TC_279
OE0582		A file manager shall implement the inherited FileSystem operations as required under section 3.1.3.4.2 for each mounted file system.	3.1.3.4.3.5.4	Automated	FileManager copy, FileManager create remove, FileManager exists, FileManager list, FileManager mkdir rmdir, FileManager open	OE_TC_031
OE0583		The FileSystem operations inherited by a file manager shall remove the name of the mounted file system from input pathnames before passing the pathnames to any operation on a mounted file system.	3.1.3.4.3.5.4	Automated	FileManager copy, FileManager create remove, FileManager exists, FileManager list, FileManager mkdir rmdir, FileManager open	OE_TC_031
OE0584		The file manager shall use the FileSystem operations of the file system whose associated mount point exactly matches the input fileName parameter to the lowest matching subdirectory.	3.1.3.4.3.5.4	Automated	FileManager Distributed FileSystem	OE_TC_279
OE0585		The query operation shall return the combined mounted file systems information to the calling client based upon the given input FileSystemProperties' ID elements.	3.1.3.4.3.5.5.3	Automated	FileManager query	OE_TC_280



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0586		As a minimum, the query operation shall support the following input fileSystemProperties ID elements.  SIZE - a property itemID value of "SIZE" causes the query operation to return the combined total size of all the mounted file systems as an unsigned long long property value.  AVAILABLE_SPACE - a property itemID value of "AVAILABLE_SPACE" causes the query operation to return the combined total available space (in octets) of all the mounted file systems as unsigned long long property value.	3.1.3.4.3.5.5.3	Automated	FileManager query	OE_TC_280
OE0587		The query operation shall raise the UnknownFileSystemProperties exception when the input fileSystemProperties parameter contains an invalid property ID element.	3.1.3.4.3.5.5.5	Automated	FileManager query UnknownFileSystemProperties	OE_TC_280
OE0588		Domain Profile files shall be compliant to the Document Type Definitions (DTDs) provided in Appendix D.	3.1.3.5	Automated	DTD Verification, Application Profile Attribute, ApplicationFactory softwareProfile Attribute, Device softwareProfile Attribute, DeviceManager deviceConfigurationProfile Attribute, DomainManager domainManagerProfile Attribute	OE_TC_070
OE0589		DTD files are installed in the domain and shall have ".dtd" as their filename extension.	3.1.3.5	Automated	DTD Verification	OE_TC_117
OE0590		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	Manual		OE_TC_070
OE0591		A Software Package Descriptor file shall have a ".spd.xml" extension.	3.1.3.5.1	Manual		OE_TC_070

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0592		A Software Component Descriptor file shall have a ".scd.xml" extension.	3.1.3.5.2	Manual		OE_TC_070
OE0594		A Properties File shall have a ".prf.xml" extension.	3.1.3.5.4	Manual		OE_TC_070
OE0595		A Device Package Descriptor File shall have a ".dpd.xml" extension.	3.1.3.5.5	Manual		OE_TC_070
OE0596		A Device Configuration Descriptor file shall have a ".dcd.xml" extension.	3.1.3.5.6	Manual		OE_TC_070
OE0597		A DomainManager Configuration Descriptor file shall have a ".dmd.xml" extension.	3.1.3.5.8	Manual		OE_TC_070
OE0598		The error number shall indicate a CF ErrorNumberType value.	3.1.3.6.3	Semi-Automated	File sizeOfFileException, FileManager copy FileException, FileManager create FileException, FileManager list FileException, FileManager mkdir FileException, FileManager open FileException, FileManager remove FileException, FileManager rmdir FileException, FileSystemcopy FileException, FileSystemcreate FileException, FileSystemmkdir FileException, FileSystemopen FileException, FileSystemrmdir FileException	OE_TC_068, OE_TC_069, OE_TC_104, OE_TC_105, OE_TC_147, OE_TC_148, OE_TC_149, OE_TC_150, OE_TC_151, OE_TC_152

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0599		The CF InvalidFileName exception indicates an invalid file name was passed to a file service operation. The error number shall indicate a CF ErrorNumberType value.	3.1.3.6.4	Semi-Automated	DomainManager installApplication InvalidFileName, FileManager copy InvalidFileName, FileManager create InvalidFileName, FileManager exists InvalidFileName, FileManager list InvalidFileName, FileManager mkdir InvalidFileName, FileManager mount InvalidFileName, FileManager open InvalidFileName, FileManager remove InvalidFileName, FileManager rmdir InvalidFileName, FileSystem copy InvalidFileName, FileSystem create InvalidFileName, FileSystem exists InvalidFileName, FileSystem list InvalidFileName, FileSystem mkdir InvalidFileName, FileSystem open InvalidFileName, FileSystem remove InvalidFileName, FileSystem rmdir InvalidFileName, Device unload InvalidFileName	OE_TC_103, OE_TC_132, OE_TC_133, OE_TC_134, OE_TC_135, OE_TC_136, OE_TC_137, OE_TC_139, OE_TC_140, OE_TC_141, OE_TC_142, OE_TC_143, OE_TC_144, OE_TC_159
OE0605		Valid individual filenames and directory names shall be 40 characters or less.	3.1.3.4.2.1	semi-Automated	FileSystem copy InvalidFileName, FileSystem create InvalidFileName, FileSystem exists InvalidFileName, FileSystem list InvalidFileName, FileSystem mkdir InvalidFileName, FileSystem open InvalidFileName, FileSystem remove InvalidFileName, FileSystem rmdir InvalidFileName	OE_TC_060

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0605	C123	Valid characters for a filename or directory name are the 62 alphanumeric characters (Upper, and lowercase letters and the numbers 0 to 9) in addition to the "." (period), "_" (underscore) and "-" (hyphen) characters.	3.1.3.4.2.1	semi-Automated	FileSystemcopy InvalidFileName, FileSystemcreate InvalidFileName, FileSystemexists InvalidFileName, FileSystemlist InvalidFileName, FileSystemmkdir InvalidFileName, FileSystemopen InvalidFileName, FileSystemremove InvalidFileName, FileSystemrmdir InvalidFileName	OE_TC_060
OE0605	C124	The filenames "." ("dot") and ".." ("dot-dot") are invalid in the context of a file system.	3.1.3.4.2.1	semi-Automated	FileSystemcopy InvalidFileName, FileSystemcreate InvalidFileName, FileSystemexists InvalidFileName, FileSystemlist InvalidFileName, FileSystemmkdir InvalidFileName, FileSystemopen InvalidFileName, FileSystemremove InvalidFileName, FileSystemrmdir InvalidFileName	OE_TC_060
OE0613		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		OE_TC_119
OE0613	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		OE_TC_119
OE0618		The executable parameters of a logical device shall accept the standard argv arguments as used in the POSIX exec family of functions [4].	3.3.1	Automated	Device load execute terminate unload	OE_TC_092
OE0619		A logical device shall accept the executable parameters as specified in section 3.1.3.3.3.5.1.3 (ExecutableDevice::execute).	3.3.1	Manual		OE_TC_092

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0621		A logical device shall register itself with a device manager using the value associated with the DEVICE_MGR_IOR parameter per 3.1.3.2.4.5.	3.3.3	Manual		OE_TC_095
OE0622		A child device shall add itself to a parent device using the executable Composite Device IOR parameter per 3.1.3.2.4.5.	3.3.3	Manual		OE_TC_096
OE0623		The values associated with the parameters (PROFILE_NAME, COMPOSITE_DEVICE_IOR, DEVICE_ID and DEVICE_LABEL) as described in 3.1.3.2.4.5 shall be used to set the Device's softwareProfile, compositeDevice, identifier, and label attributes, respectively.	3.3.3	Manual		OE_TC_097
OE0626		Each logical device shall have a SPD, SCD, DPD, and one or more Properties Descriptors as described in section 3.1.3.5.	3.3.4	Automated	Device softwareProfile Attribute	OE_TC_282
OE0700		If a log service is implemented, the log service shall conform to the OMGLightweight Log Service Specification [7].	3.1.2.2	Automated	Log SetMaxSize InvalidParam, Log Clearlog, Log GetAvailableStatus, Log Get/Set Administrative State, Log GetLogFullAction SetLogFullAction, Log GetMaxSize SetmaxSize GetCurrentSize, Log GetNumRecords, Log GetOperationalState, Log GetRecordIDFromTime, Log WriteRecord, Log WriteRecords, Log RetrieveRecords, Log RetrieveRecordsByLevel, Log RetrieveRecordsByProducerId, Log RetrieveRecordsByProducename, Log Destroy	OE_TC_005, OE_TC_006, OE_TC_007, OE_TC_008, OE_TC_009, OE_TC_010, OE_TC_011, OE_TC_012, OE_TC_013, OE_TC_014, OE_TC_015, OE_TC_016, OE_TC_017, OE_TC_018, OE_TC_019

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0700	C182	This operation returns the maximum capacity in bytes of the storage area.	3.3.1.1	Automated	Log GetMaxSize SetMaxSize GetCurrentSize	OE_TC_005
OE0700	C183	The get_current_size operation returns the size in bytes of the log storage area currently occupied by logging records.	3.3.1.2	Automated	Log GetMaxSize SetMaxSize GetCurrentSize	OE_TC_005
OE0700	C184	This operation raises the InvalidParamException if the supplied parameter is invalid.	3.3.4.1	Automated	Log SetMaxSize InvalidParam, Log GetMaxSize SetMaxSize GetCurrentSize	OE_TC_005
OE0700	C185	This operation allows setting of the maximum capacity in bytes of the storage area.	3.3.4.1	Automated	Log GetMaxSize SetMaxSize GetCurrentSize	OE_TC_005
OE0700	C186	The get_n_records operation returns the number of logging records currently stored in the log storage area.	3.3.1.3	Automated	Log GetNumRecords	OE_TC_006
OE0700	C187	The get_log_full_action operation returns the information about which action the LoggingService will take when the storage area becomes full.	3.3.1.4	Automated	Log GetLogFullAction SetLogFullAction	OE_TC_007
OE0700	C188	The possible values are HALT, which means no further logging records are accepted and stored; or WRAP, which means the Log continues by overwriting the oldest records in the storage area.	3.3.1.4	Automated	Log GetLogFullAction SetLogFullAction	OE_TC_007
OE0700	C189	The set_log_full_action operation allows the specification which action should be taken after all free space in the log storage area is depleted.	3.3.4.2	Automated	Log GetLogFullAction SetLogFullAction	OE_TC_007
OE0700	C190	The possible values are HALT, which means no further logging records are accepted and stored; or WRAP, which means the Log continues by overwriting the oldest records in the storage area.	3.3.4.2	Automated	Log GetLogFullAction SetLogFullAction	OE_TC_007
OE0700	C191	When the LogFullAction type is set to WRAP, the Log will set the availability status logFull state to false.	3.3.4.2	Automated	Log GetLogFullAction SetLogFullAction	OE_TC_007

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0700	C192	The returned instance of the AvailabilityStatus type contains two Boolean values: off_duty, which indicates the log is disabled when true; and log_full, which indicates that all free space is depleted in the log storage area.	3.3.1.5	Automated	Log GetAvailabilityStatus	OE_TC_008
OE0700	C193	The get_administrative_state is used to read the administrative state of the Log.	3.3.1.6	Automated	Log Get/Set Administrative State	OE_TC_009
OE0700	C194	If the state is locked, no new records are accepted. Reading of already stored records is not affected.	3.3.1.6	Automated	Log Get/Set Administrative State	OE_TC_009
OE0700	C195	This operation allows one to affect the ability of the logging service to accept and store new logging records by administrative action.	3.3.4.3	Automated	Log Get/Set Administrative State	OE_TC_009
OE0700	C196	If the state is locked, no new records are accepted.	3.3.4.3	Automated	Log Get/Set Administrative State	OE_TC_009
OE0700	C197	Reading of already stored records is not affected.	3.3.4.3	Automated	Log Get/Set Administrative State	OE_TC_009
OE0700	C198	If the state is set to unlocked, the log operates normally.	3.3.4.3	Automated	Log Get/Set Administrative State	OE_TC_009
OE0700	C199	The get_operational_state operation returns the actual operational state of the log.	3.3.1.7	Automated	Log GetOperationalState	OE_TC_010
OE0700	C200	Possible values are enabled, which means the log is fully functional and available to log producer and log consumer clients; or disabled, which indicates the log has encountered a runtime problem and is not available for use by log producers or log consumers.	3.3.1.7	Automated	Log GetOperationalState	OE_TC_010
OE0700	C201	The get_record_id_from_time operation returns the record Id of the first record in the Log with a time stamp that is greater than, or equal to, the time specified in the fromTime parameter.	3.3.2.1	Automated	Log GetRecordIdFromTime	OE_TC_011
OE0700	C202	If the Log does not contain a record that meets the criteria provided, then the RecordId returned corresponds to the next record that will be recorded in the future.	3.3.2.1	Automated	Log GetRecordIdFromTime	OE_TC_011

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0700	C203	The retrieve_records operation returns a LogRecordSequence that begins with the record specified by the currentId parameter.	3.3.2.2	Automated	Log RetrieveRecords	OE_TC_012
OE0700	C204	The number of records in the LogRecordSequence returned by the retrieve_records operation is equal to the number of records specified by the howMany parameter, or the number of records available if the number of records specified by the howMany parameter cannot be met.	3.3.2.2	Automated	Log RetrieveRecords	OE_TC_012
OE0700	C205	The log will update howMany to indicate the number of records returned and will set currentId to either the id of the record following the last examined record or the next record that will be recorded in the future if there are no further records available.	3.3.2.2	Automated	Log RetrieveRecords	OE_TC_012
OE0700	C206	If the record specified by currentId does not exist, but corresponds to the next record that will be recorded in the future, the retrieve_records operation returns an empty list of LogRecords, sets howMany to zero, and leaves the value of currentId unchanged.	3.3.2.2	Automated	Log RetrieveRecords	OE_TC_012
OE0700	C207	If the record specified by currentId does not exist and does not correspond to the next record that will be recorded in the future, or if the Log is empty, the retrieve_records operation returns an empty list of LogRecords, and sets both, currentId and howMany to zero.	3.3.2.2	Automated	Log RetrieveRecords	OE_TC_012
OE0700	C208	The retrieve_records_by_level operation returns a LogRecordSequence of records that correspond to the supplied LogLevels.	3.3.2.3	Automated	Log RetrieveRecordsByLevel	OE_TC_013
OE0700	C209	Candidate records for the LogRecordSequence begin with the record specified by the currentId parameter.	3.3.2.3	Automated	Log RetrieveRecordsByLevel	OE_TC_013



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0700	C210	The number of records in the LogRecordSequence returned by the retrieve_records_by_level operation is equal to the number of records specified by the howMany parameter, or the number of records available if the number of records specified by the howMany parameter cannot be met.	3.3.2.3	Automated	Log RetrieveRecordsByLevel	OE_TC_013
OE0700	C211	The log will update howMany to indicate the number of records returned and will set currentId to either the id of the record following the last examined record or the next record that will be recorded in the future if there are no further records available.	3.3.2.3	Automated	Log RetrieveRecordsByLevel	OE_TC_013
OE0700	C212	If the record specified by currentId does not exist, but corresponds to the next record that will be recorded in the future, the retrieve_records_by_level operation returns an empty list of LogRecords, sets howMany to zero, and leaves the value of currentId unchanged.	3.3.2.3	Automated	Log RetrieveRecordsByLevel	OE_TC_013
OE0700	C213	If the record specified by currentId does not exist and does not correspond to the next record that will be recorded in the future, or if the Log is empty, the retrieve_records_by_level operation returns an empty list of LogRecords, and sets both, currentId and howMany to zero.	3.3.2.3	Automated	Log RetrieveRecordsByLevel	OE_TC_013
OE0700	C214	The retrieve_records_by_producer_id operation returns a LogRecordSequence of records that correspond to the supplied producerIds.	3.3.2.4	Automated	Log RetrieveRecordsByProducerID	OE_TC_014
OE0700	C215	Candidate records for the LogRecordSequence begin with the record specified by the currentId parameter.	3.3.2.4	Automated	Log RetrieveRecordsByProducerID	OE_TC_014
OE0700	C216	The number of records in the LogRecordSequence returned by the retrieve_records_by_producer_id operation is equal to the number of records specified by the howMany parameter, or the number of records available if the number of records specified by the howMany parameter cannot be met.	3.3.2.4	Automated	Log RetrieveRecordsByProducerID	OE_TC_014

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0700	C217	The log will update howMany to indicate the number of records returned and will set currentId to either the id of the record following the last examined record or the next record that will be recorded in the future if there are no further records available.	3.3.2.4	Automated	Log RetrieveRecordsByProducerID	OE_TC_014
OE0700	C218	If the record specified by currentId does not exist, but corresponds to the next record that will be recorded in the future, the retrieve_records_by_producer_id operation returns an empty list of LogRecords, sets howMany to zero, and leaves the value of currentId unchanged.	3.3.2.4	Automated	Log RetrieveRecordsByProducerID	OE_TC_014
OE0700	C219	If the record specified by currentId does not exist and does not correspond to the next record that will be recorded in the future, or if the Log is empty, the retrieve_records_by_producer_id operation returns an empty list of LogRecords, and sets both, currentId and howMany to zero.	3.3.2.4	Automated	Log RetrieveRecordsByProducerID	OE_TC_014
OE0700	C220	The retrieve_records_by_producer_name operation returns a LogRecordSequence of records that correspond to the supplied producerNames.	3.3.2.5	Automated	Log RetrieveRecordsByProducerName	OE_TC_015
OE0700	C221	Candidate records for the LogRecordSequence begin with the record specified by the currentId parameter.	3.3.2.5	Automated	Log RetrieveRecordsByProducerName	OE_TC_015
OE0700	C222	The number of records in the LogRecordSequence returned by the retrieve_records_by_producer_name operation is equal to the number of records specified by the howMany parameter, or the number of records available if the number of records specified by the howMany parameter cannot be met.	3.3.2.5	Automated	Log RetrieveRecordsByProducerName	OE_TC_015
OE0700	C223	The log will update howMany to indicate the number of records returned and will set currentId to either the id of the record following the last examined record or the next record that will be recorded in the future if there are no further records available.	3.3.2.5	Automated	Log RetrieveRecordsByProducerName	OE_TC_015

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0700	C224	If the record specified by currentId does not exist, but corresponds to the next record that will be recorded in the future, the retrieve_records_by_producer_name operation returns an empty list of LogRecords, sets howMany to zero, and leaves the value of currentId unchanged.	3.3.2.5	Automated	Log RetrieveRecordsByProducerName	OE_TC_015
OE0700	C225	If the record specified by currentId does not exist and does not correspond to the next record that will be recorded in the future, or if the Log is empty, the retrieve_records_by_producer_name operation returns an empty list of LogRecords, and sets both, currentId and howMany to zero.	3.3.2.5	Automated	Log RetrieveRecordsByProducerName	OE_TC_015
OE0700	C226	The write_records operation adds the log records supplied in the records parameter to the Log.	3.3.3.1	Automated	Log WriteRecords	OE_TC_016
OE0700	C227	When there is insufficient storage to add one of the supplied log records to the Log, and the LogFullAction is set to HALT, the write_records operation will set the availability status logFull state to true.	3.3.3.1	Automated	Log WriteRecords	OE_TC_016
OE0700	C228	When there is insufficient storage to add one of the supplied log records to the Log, and the LogFullAction is set to WRAP, the write_records operation will overwrite the oldest LogRecords with the newest records, as they are written to the Log, and leave the availability status logFull state unchanged.	3.3.3.1	Automated	Log WriteRecords	OE_TC_016
OE0700	C229	The write_records operation inserts the current UTC time to the time field of each record written to the Log, and assigns a unique record id to the id field of the LogRecord.	3.3.3.1	Automated	Log WriteRecords	OE_TC_016
OE0700	C230	Log records accepted for storage by the write_records will be available for retrieval in the order received.	3.3.3.1	Automated	Log WriteRecords	OE_TC_016
OE0700	C231	The write_record operation adds a log record supplied in the record parameter to the Log.	3.3.3.2	Automated	Log WriteRecord	OE_TC_017

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0700	C232	When there is insufficient storage to add the supplied log record to the Log, and the LogFullAction is set to HALT, the write_record operation will set the availability status logFull state to true.	3.3.3.2	Automated	Log WriteRecord	OE_TC_017
OE0700	C233	When there is insufficient storage to add the supplied log record to the Log, and the LogFullAction is set to WRAP, the write_record operation will overwrite the oldest LogRecords with the new record, and leave the availability status logFull state unchanged.	3.3.3.2	Automated	Log WriteRecord	OE_TC_017
OE0700	C234	The write_record operation inserts the current UTC time to the time field of each record written to the Log, and assigns a unique record id to the id field of the LogRecord.	3.3.3.2	Automated	Log WriteRecord	OE_TC_017
OE0700	C235	Log records accepted for storage by write_record will be available for retrieval in the order received.	3.3.3.2	Automated	Log WriteRecord	OE_TC_017
OE0700	C236	This operation purges all logging records from the log storage area; however, it does not alter the size of the storage area in any way.	3.3.4.4	Automated	Log ClearLog	OE_TC_018
OE0700	C237	The log will set the availability status logFull state to false.	3.3.4.4	Automated	Log ClearLog	OE_TC_018
OE0700	C238	This operation will destroy the associated instance of the Log class.	3.3.4.5	Automated	Log Destroy	OE_TC_019
OE0700	C239	All existing records in the log storage area are irrecoverably lost and the memory resources associated with the storage area are released.	3.3.4.5	Manual		OE_TC_019
OE0700	C240	For example, if 3 records are provided in the records parameter, and while trying to write the second record to the log, the record will not fit, then the log is considered to be full. Therefore, the second and third records will not be stored in the log but the first record would have been successfully stored.	3.3.3.1	Automated	Log WriteRecords	OE_TC_016

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0701		Base Device Interfaces shall be implemented using the CF IDL presented in AppendixC.	3.1.3.3	Manual		OE_TC_062
OE0702		The OE shall provide two standard event channels: Incoming Domain Management and Outgoing Domain Management.	3.1.2.3.1	Automated	EventService Event Channel Function	OE_TC_003
OE0703		Base Application Interfaces shall be implemented using the CF IDL presented in AppendixC.	3.1.3.1	Manual		OE_TC_102
OE0704		The stop operation shall not inhibit subsequent configure, query, and start operations.	3.1.3.1.6.5.2.3	Automated	Application Stop, Device stop start	OE_TC_188
OE0707		The create operation shall perform the comparison of allocation properties of the application to those of each candidate device, according to the allocation property's action element, for those application component properties whose kindtype is allocation and whose action element is not external.	3.1.3.2.2.5.1.3	Manual		OE_TC_121
OE0708		The create operation shall deallocate any capacity allocations on devices that do not satisfy the application components allocation requirements or that are not utilized due to an unsuccessful application creation.	3.1.3.2.2.5.1.3	Manual		OE_TC_162
OE0710		The create operation shall establish connections for an application which are specified in the SAD domainfinder element.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_164
OE0712		The registerDeviceManager operation shall return without exception and not register a new device manager when that device manager, indicated by the input deviceMgr parameter, has the same identifier attribute as a previously registered device manager and the reference to the registered device manager refers to an existing object.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager	OE_TC_213

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0713		The registerDeviceManager operation shall register the new device manager indicated by the input deviceMgr parameter, when the previously registered device manager has the same identifier attribute as the new device manager and the reference to the registered device manager refers to a nonexistent object.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager Non-Existent Object	OE_TC_213
OE0714		The registerDeviceManager operation shall write an ADMINISTRATIVE_EVENT log record when reference to the registered device manager refers to a nonexistent object.	3.1.3.2.3.6.1.3	Automated	DomainManager registerDeviceManager Non-Existent Object	OE_TC_213
OE0715		The registerDevice operation shall verify that the input parameters, registeringDevice and registeredDeviceMgr, are not nil CORBA object references.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice InvalidObjectReference	OE_TC_166
OE0716		The registerDevice operation shall return without exception and not register a new device when that device, indicated by the input registeringDevice parameter, has the same identifier attribute as a previously registered device and the reference to the registered device refers to an existing object.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_167
OE0717		The registerDevice operation shall register the new device indicated by the input registeringDevice parameter, when the previously registered device has the same identifier attribute as the new device and the reference to the registered device refers to a nonexistent object.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_168
OE0718		The registerDevice operation shall write an ADMINISTRATIVE_EVENT log record when reference to the registered device refers to a nonexistent object.	3.1.3.2.3.6.2.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_168

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0719		The registerDevice operation shall establish any pending connections from previously registered device managers when the registering device completes these connections.	3.1.3.2.3.6.2.3	Manual		OE_TC_113
OE0720		The domain manager shall create a naming context using "/DomainName" as the id attribute to the input name parameter, and "" (Null string) as the kind attribute.	3.1.3.2.3.5	Automated	DomainManager Naming Service Register	OE_TC_156
OE0722		The installApplication operation shall raise the ApplicationAlreadyInstalled exception when the softwareassembly element id attribute of the referenced application is the same as a previously registered application.	3.1.3.2.3.6.3.5	Automated	DomainManager installApplication ApplicationAlreadyInstalled	OE_TC_026
OE0723		Connections broken as a result of the unregisterDeviceManager operation shall be considered as "pending" for future connections when the component to which the device manager or its registered devices and services were connected still exists.	3.1.3.2.3.6.4.3	Manual		OE_TC_251
OE0724		Connections broken as a result of the unregisterDevice operation shall be considered as "pending" for future connections when the component to which the device was connected still exists.	3.1.3.2.3.6.5.3	Automated	DomainManager registerDevice unregisterDevice	OE_TC_253
OE0725		The registerService operation shall register the new service, indicated by the input registeringService parameter, when the previously registered service has the same name and the reference to the registered service refers to a nonexistent object.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_073
OE0726		The registerService operation shall write an ADMINISTRATIVE_EVENT log record when reference to the registered service refers to a nonexistent object.	3.1.3.2.3.6.7.3	Automated	DomainManager registerService unregisterService	OE_TC_256

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0727		The unregisterService operation shall disconnect the established connections (including those made to the CORBA Event Service event channels) of the unregistering service indicated by the input unregisteringService parameter.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258
OE0728		Connections broken as a result of the unregisterService operation shall be considered as "pending" for future connections when the component to which the service was connected still exists.	3.1.3.2.3.6.8.3	Automated	DomainManager registerService unregisterService	OE_TC_258
OE0729		The readonly compositeDevice attribute shall contain a nil CORBA object reference when this device is not a parent device.	3.1.3.3.1.4.6	Automated	Device compositeDevice Attribute	OE_TC_218
OE0730		The allocateCapacity operation shall only accept properties for the input capacities parameter which are simple properties whose kindtype is allocation and whose action element is external contained in the component's SPD.	3.1.3.3.1.5.1.3	Manual		OE_TC_118
OE0731		Multiple loads of the same file as indicated by the input fileName parameter shall not result in an exception.	3.1.3.3.2.5.1.3	semi-Automated	Device load execute terminate unload	OE_TC_153
OE0732		The registerDeviceManager operation shall raise the CF InvalidProfile exception when the device manager's DCD file and the DCD's referenced files do not exist.	3.1.3.2.3.6.1.5	semi-Automated	DomainManager registerDeviceManager InvalidProfile	OE_TC_024
OE0733		A valid pathname shall not exceed 1024 characters.	3.1.3.4.2.1	Automated	FileSystem mkdir rmdir	OE_TC_060
OE0733	C125	Valid pathnames are structured according to the POSIX specification whose valid characters include the "/" (forward slash) character in addition to the valid filename characters.	3.1.3.4.2.1	Manual		OE_TC_060



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0734		The copy operation shall overwrite the destination file, when the destination file already exists and is not identical to the source file.	3.1.3.4.2.5.2.3	Automated	FileManager copy, FileSystemcopy	OE_TC_106
OE0735		The copy operation shall raise the CF InvalidFileName exception when the destination pathname is identical to the source pathname.	3.1.3.4.2.5.2.5	Automated	FileManager copy, FileSystemcopy	OE_TC_144
OE0736		These wildcards shall only be applied following the right-most forward-slash character ("/") in the pathname contained in the input pattern parameter.	3.1.3.4.2.5.4.3	semi-Automated	FileManager list	OE_TC_031
OE0737		The open operation shall set the filePointer attribute of the returned file instance to the beginning of the file.	3.1.3.4.2.5.6.4	Automated	FileManager open, FileSystemopen	OE_TC_273
OE0738		The rmdir operation shall not remove the directory identified by the input directoryName parameter when the directory contains files.	3.1.3.4.2.5.8.3	Automated	FileManager mkdir rmdir, FileSystem mkdir rmdir	OE_TC_211
OE0739		The file manager shall propagate exceptions raised by a mounted file system.	3.1.3.4.3.5.4	Automated	FileManager copy FileException, FileManager copy InvalidFileName, FileManager create FileException, FileManager create InvalidFileName, FileManager exists InvalidFileName, FileManager list FileException, FileManager list InvalidFileName, FileManager mkdir FileException, FileManager mkdir InvalidFileName, FileManager open FileException, FileManager open InvalidFileName, FileManager query UnknownFileSystemProperties, FileManager remove FileException, FileManager remove invalidFileName, FileManager rmdir FileException, FileManager rmdir InvalidFileName	OE_TC_040

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0745		The domain manager shall create a name binding to the created naming context using "/DomainName" as the id attribute to the input name parameter, and "" (Null string) as the kind attribute, where DomainName is identical to the name attribute of the domain manager's DMD domainmanagerconfiguration element and the input object parameter is the domain manager object reference. [6]	3.1.3.2.3.5	Automated	DomainManger Naming Service Register	OE_TC_156
OE0746		The OE shall provide an implementation of a CORBA Naming Service which implements the CosNaming module NamingContext interface operations: bind, bind_new_context, unbind, destroy, and resolve as defined in the OMGInteroperable Naming Service Specification [6] using the IDL found in Appendix A of that reference.	3.1.2.1	Automated	DomainManager Naming Service Register, NamingService destroy, NamingService Functions	OE_TC_001
OE0747		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	semi-Automated	Application PRODUCER_LOG_LEVEL, DeviceManager PRODUCER_LOG_LEVEL, DomainManager PRODUCER_LOG_LEVEL	OE_TC_002
OE0749		The create operation shall load application modules onto devices that have been granted successful capacity allocations and that satisfy the application components allocation requirements.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform, ApplicationFactory create with deviceAssignments	OE_TC_284
OE0750		The execute parameter for the Naming Context IOR shall be a CF Properties type with an id element set to "NAMING_CONTEXT_IOR" and a value element set to the stringified IOR of the naming context to which the component will bind.	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_155

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0751		The Name Binding execute parameter shall be a CF Properties type with an id element set to "NAME_BINDING" and a value element set to a string in the format of "ComponentName_UniqueIdentifier".	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_160
OE0752		The Component Identifier execute parameter shall be a CF Properties type with an id element set to "COMPONENT_IDENTIFIER" and a value element set to a string in the format of "Component_Instantiation_Identifier: Application_Name".	3.1.3.2.2.5.1.3	Automated	ApplicationFactory create PseudoWaveform	OE_TC_161
OE0754		The device manager shall pass "execparam" parameters' IDs and values as string values.	3.1.3.2.4.5	Manual		OE_TC_112
OE0755		The list operation shall return a zero length sequence when no file is found which matches the search pattern.	3.1.3.4.2.5.4.4	semi-Automated	FileManager list	OE_TC_031
OE0756		The device manager shall pass the componentinstantiation element "execparam" properties that have values as parameters.	3.1.3.2.4.5	Manual		OE_TC_112
OE0760		Each mounted file systemname shall be unique within the device manager.	3.1.3.2.4.5	Automated	DeviceManager fileSys Attribute	OE_TC_283
OE0761		The open operation shall open the file with read-only access when the inputread_Only parameter is TRUE.	3.1.3.4.2.5.6.3	Automated	FileManager open, FileSystemopen	OE_TC_273
<b>Appendix B - Application Environment Profile (AEP)</b>						
OE0657		The functions in Table B-3 shall behave as described in the applicable clauses of the referenced POSIX referenced POSIX specifications contained in Table B-1.	B4.1.1	Manual		OE_TC_131
OE0658		The functions listed in Table B-4 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.2	Manual		OE_TC_131

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0659		The functions listed in Table B-5 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.3	Manual		OE_TC_131
OE0660		The functions listed in Table B-6 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.4	Manual		OE_TC_131
OE0663		The functions listed in Table B-8 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.6	Manual		OE_TC_131
OE0664		The functions listed in Table B-9 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.7	Manual		OE_TC_131
OE0665		The functions listed in Table B-10 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.8	Manual		OE_TC_131
OE0666		An application that conforms to the AEP shall be guaranteed that the file mode creation mask for any object created by any process is S-IRWXU; that is, the object shall be fully accessible to the creator.	B.4.1.8	Manual		OE_TC_044
OE0667		The functions listed in Table B-11 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.9	Manual		OE_TC_131
OE0668		The functions listed in Table B-12 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.10	Manual		OE_TC_131
OE0669		The functions listed in Table B-13 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.11	Manual		OE_TC_131
OE0670		The functions listed in Table B-14 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.12	Manual		OE_TC_131

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0671		The function listed in Table B-15 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.13	Manual		OE_TC_131
OE0672		The function listed in Table B-16 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.14	Manual		OE_TC_131
OE0673		The functions listed in Table B-17 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.15	Manual		OE_TC_131
OE0762		The functions listed in Table B-7 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.5	Manual		OE_TC_131
OE0763		The functions listed in Table B-18 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.16	Manual		OE_TC_131
OE0764		The functions listed in Table B-19 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.17	Manual		OE_TC_131
OE0765		The functions listed in Table B-20 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.18	Manual		OE_TC_131
OE0767		The functions listed in Table B-21 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.19	Manual		OE_TC_131
OE0768		The functions listed in Table B-22 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.20	Manual		OE_TC_131
OE0769		The options, limits, and any other constraints on POSIX.1 shall be provided as described in Table B-2.	B.4.1	Manual		OE_TC_131
OE0770		The function listed in Table B-23 shall behave as described in the referenced clause.	B.4.1.21	Manual		OE_TC_131

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
<b>SCA 2.2.2 Extensions 22 December 2006</b>						
OE0771		The registerService operation shall, upon successful service registration of a non-SCA service with an input name parameter in the "identifier\type" format, make the value provided in the "identifier" portion of the name accessible via the domainfinder servicename mechanism.	2.1 Extension	Manual		OE_TC_176
OE0772		The registerService operation shall, upon successful service registration of a non-SCA service with an input name parameter in the "identifier\type" format, make the value provided in the "type" portion of the name accessible via the domainfinder servicetype mechanism.	2.1 Extension	Manual		OE_TC_176
OE0773		The unregisterService operation shall remove non-SCA services (i.e. those with a name in the "identifier\type" format) by matching either a fully qualified name in the "identifier\type" format or a simple name with only the "identifier" portion.	2.2 Extension	Manual		OE_TC_177
OE0774		The create operation shall recognize application deployment channel preferences contained within an Application Deployment Descriptor file if the CF implementation provides enhanced deployment support via the use of both a Deployment Platform Descriptor and an Application Deployment Descriptor file.	2.3 Extension	Manual		OE_TC_178
OE0775		The create operation shall recognize a property which is a CF Properties type with an id of "DEPLOYMENT_CHANNEL" and a value that is a string sequence if the CF implementation provides enhanced deployment support via the use of a Deployment PlatformDescriptor file.	2.3 Extension	Manual		OE_TC_178

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0776		The create operation shall recognize channel preferences contained within a "DEPLOYMENT_CHANNEL" property contained within the initConfiguration parameter if the CF implementation provides enhanced deployment support via the use of a Deployment Platform Descriptor file.	2.3 Extension	Manual		OE_TC_178
OE0777		The create operation shall attempt to allocate an application to the Deployment PlatformDescriptor file channel alternatives provided within a "DEPLOYMENT_CHANNEL" property or an Application Deployment Descriptor file in a sequential manner.	2.3 Extension	Manual		OE_TC_178
OE0778		The create operation shall utilize channel preferences expressed within a "DEPLOYMENT_CHANNEL" property rather than those contained within an Application Deployment Descriptor file if both exist and the CF implementation provides enhanced deployment support via the use of a Deployment Platform Descriptor file.	2.3 Extension	Manual		OE_TC_178
OE0779		The create operation shall recognize a deployment option with a deployedname attribute value of "DEFAULT" which matches all application instance names that are not explicitly identified by a deployedname attribute value within the same descriptor file if the CF implementation provides enhanced deployment support via the use of an Application Deployment Descriptor file.	2.3 Extension	Manual		OE_TC_179

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0780		For domainfinder element "servicetype" connections to a non-SCA service whose service type is provided by a service contained within a channelement servicelist, the create operation shall only attempt to establish connections to services within the list if the CF implementation provides enhanced deployment support via the use of a Deployment Platform Descriptor file.	2.3 Extension	Manual		OE_TC_180
OE0781		The create operation shall raise the InvalidInitConfiguration exception when the input initConfiguration parameter "DEPLOYMENT_CHANNEL" property contains an invalid channel reference.	2.4 Extension	Manual		OE_TC_181
OE0782		The InvalidInitConfiguration invalidProperties parameter shall identify the invalid channels.	2.4 Extension	Manual		OE_TC_181
OE0783		The create operation shall raise the CreateApplicationError exception when the CF implementation provides enhanced deployment support via the use of a Deployment Platform Descriptor file but the CF is not able to allocate the application to any of the provided channel alternatives.	2.4 Extension	Manual		OE_TC_182
OE0784		The create operation shall raise the CreateApplicationError exception when the CF implementation provides enhanced deployment support via the use of a Deployment Platform Descriptor file and a domainfinder element "servicetype" connection to a non-SCA service whose service type is provided by a service contained within a channelement servicelist can not be established to a service identified within that list.	2.4 Extension	Manual		OE_TC_183



Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0785		If a non-SCA service is deployed by the device manager, the device manager shall supply execute operation parameters consisting of: 1. Device manager IOR - The ID is "DEVICE_MGR_IOR" and the value is a string that is the DeviceManager stringified IOR. 2. Service Name - The ID is "SERVICE_NAME" and the value is a string in an "identifier\type" format where the identifier corresponds to the DCD componentinstantiation usagename element and the type corresponds to a servicetype repository identifier from the SCD. 3. The execute ("execparam") properties as specified in the DCD for a componentinstantiation element.	2.5 Extension	Manual		OE_TC_184
OE0786		The device manager shall pass the componentinstantiation element "execparam" properties that have values as parameters.	2.5 Extension	Manual		OE_TC_184
OE0787		The device manager shall pass "execparam" parameters' IDs and values as string values.	2.5 Extension	Manual		OE_TC_184
OE0788		A Deployment PlatformDescriptor file shall have a ".pdd.xml" extension.	2.8 Extension	Manual		OE_TC_185
<b>SCA Application Environment Profile Amended 22 October 2008</b>						
OE0790		The function listed in Table B-24 shall behave as described in the referenced clause.	B.4.1.22	Manual		OE_TC_172
OE0791		The Standard C Library header files listed in Table B-25 shall be included within the AEP as described in the referenced clause.	B.5	Manual		OE_TC_175
OE0794		The functions in Table B-3 shall behave as described in the applicable clauses of the referenced POSIX referenced POSIX specifications contained in Table B-1.	B.4.1.1	Manual		OE_TC_172

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0795		The functions listed in Table B-4 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.2	Manual		OE_TC_172
OE0796		The functions listed in Table B-5 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.3	Manual		OE_TC_172
OE0797		The functions listed in Table B-6 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.4	Manual		OE_TC_172
OE0800		The functions listed in Table B-8 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.6	Manual		OE_TC_172
OE0801		The functions listed in Table B-9 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.7	Manual		OE_TC_172
OE0802		The functions listed in Table B-10 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.8	Manual		OE_TC_172
OE0804		The functions listed in Table B-11 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.9	Manual		OE_TC_172
OE0805		The functions listed in Table B-12 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.10	Manual		OE_TC_172
OE0806		The functions listed in Table B-13 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.11	Manual		OE_TC_172
OE0807		The functions listed in Table B-14 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.12	Manual		OE_TC_172

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
OE0808		The function listed in Table B-15 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.13	Manual		OE_TC_172
OE0809		The function listed in Table B-16 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.14	Manual		OE_TC_172
OE0810		The functions listed in Table B-17 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.15	Manual		OE_TC_172
OE0811		The functions listed in Table B-7 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.5	Manual		OE_TC_172
OE0812		The functions listed in Table B-18 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.16	Manual		OE_TC_172
OE0813		The functions listed in Table B-19 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.17	Manual		OE_TC_172
OE0814		The functions listed in Table B-20 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.18	Manual		OE_TC_172
OE0815		The functions listed in Table B-21 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.19	Manual		OE_TC_172
OE0816		The functions listed in Table B-22 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table B-1.	B.4.1.20	Manual		OE_TC_172
OE0817		The options, limits, and any other constraints on POSIX.1 shall be provided as described in Table B-2.	B.4.1	Manual		OE_TC_172
OE0818		The function listed in Table B-23 shall behave as described in the referenced clause.	B.4.1.21	Manual		OE_TC_172

Requirement Tag	Criterion Tag	Requirement/Criterion Text	Section Number	Test Method	JTAP Test Case Name	Manual Test Case Number
SCA Networking Application Environment Profile 19 March 2010						
OE0819		The functions listed in Table 1 2 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table 1 1.	1.4.1.1	Manual		OE_TC_290
OE0820		The functions listed in Table 1 3 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table 1 1.	1.4.1.2	Manual		OE_TC_290