
Package
ossj.common.tck

ossj.common.tck Class CommonApiTckTest

```
java.lang.Object
|
+-ossj.common.tck.CommonApiTckTest
```

```
public class CommonApiTckTest
extends Object
```

Test Suite for the OSS Common API (JSR144) TCK

Constructor Summary

public	CommonApiTckTest()
--------	------------------------------------

Method Summary

static void	main (String[] args) DOCUMENT ME!
static junit.framework.Test	suite () Buits the tests suites

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

CommonApiTckTest

```
public CommonApiTckTest()
```

Methods

main

```
public static void main(String[] args)
    throws Exception
```

DOCUMENT ME!

Parameters:

args - DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

suite

```
public static junit.framework.Test suite()  
    throws Exception
```

Buits the tests suites

Returns:

the Test

Throws:

Exception - any kind of

ossj.common.tck Class CommonAPIXMLTests

```

java.lang.Object
|
+- junit.framework.Assert
   |
   +- junit.framework.TestCase
      |
      +- ossj.common.tck.JVTTestBase
         |
         +- ossj.common.tck.CommonAPIXMLTests

```

All Implemented Interfaces:

junit.framework.Test

public class **CommonAPIXMLTests**
extends [JVTTestBase](#)

CommonAPIXMLTests Class Implement the XML Mesaging Mandatory Test of the JSR144 specification.

Author:

Copyright 2002 Sun Microsystems Inc., Inc. All rights reserved.

Version:

1.12

Field Summary

public	done
public	result

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Constructor Summary

public	CommonAPIXMLTests (String name) Creates a new CommonAPIXMLTests object.
--------	--

Method Summary

String	addXmlRootElement (String rootElementTag, String xmlDoc) DOCUMENT ME!
org.w3c.dom.Document	getDocument (String tag, String xmlString) return the Document corresponding to the given tag and XML String
org.w3c.dom.Element	getElement (org.w3c.dom.Document doc, String tagName) DOCUMENT ME!
junit.framework.TestResult	run () Start the test

assertTrue, assertTrue, assertTrueMsg, fail, fail, getEnvironment, getInstanceOf,
init, log, log, run, run

```
countTestCases, createResult, getName, run, run, runBare, runTest, setName, setUp,
tearDown, toString
```

```
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertFalse, assertFalse, assertNotNull, assertNotNull,
assertNotSame, assertNotSame, assertNull, assertNull, assertEquals, assertEquals,
assertTrue, assertTrue, fail, fail
```

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

countTestCases, run

```
public boolean done
```

result

```
public junit.framework.TestResult result
```

Constructors

CommonAPIXMLTests

```
public CommonAPIXMLTests(String name)
```

Creates a new CommonAPIXMLTests object.

Parameters:

name - the test name

Throws:

Exception - any

Methods

run

```
public junit.framework.TestResult run()
```

Start the test

Returns:

the test result

run

```
public void run(junit.framework.TestResult result)
```

Start the test

Parameters:

result - where to put test result

setUp

```
public void setUp()
```

Initialize the test

tearDown

```
public void tearDown()
```

test_JVTSessionFound

```
public void test_JVTSessionFound()
```

(continued from last page)

TEST test_JVTSessionFound

PRECONDITION:

Testing Strategy:

Success Conditions:

test_XmlManagedEntityKeyAndValue

```
public void test_XmlManagedEntityKeyAndValue()
```

TEST test_XmlManagedEntityKeyAndValue

PRECONDITION:

Testing Strategy:

Success Conditions:

test_XmlQueryValue

```
public void test_XmlQueryValue()
```

TEST test_XmlQueryValue

PRECONDITION:

Testing Strategy:

Success Conditions:

test_XmlEvent

```
public void test_XmlEvent()
```

TEST test_XmlEvent

PRECONDITION:

Testing Strategy:

Success Conditions:

getDocument

```
public org.w3c.dom.Document getDocument(String tag,  
                                         String xmlString)
```

return the Document corresponding to the given tag and XML String

Parameters:

tag - xml tag

xmlString - xml string to include

Returns:

(continued from last page)

the Document

getElement

```
public org.w3c.dom.Element getElement(org.w3c.dom.Document doc,  
    String tagName)
```

DOCUMENT ME!

Parameters:

doc - DOCUMENT ME!

tagName - DOCUMENT ME!

Returns:

DOCUMENT ME!

addXmlRootElement

```
public String addXmlRootElement(String rootElementTag,  
    String xmlDoc)
```

DOCUMENT ME!

Parameters:

rootElementTag - DOCUMENT ME!

xmlDoc - DOCUMENT ME!

Returns:

DOCUMENT ME!

ossj.common.tck Class FindRemote

java.lang.Object

└-ossj.common.tck.FindRemote

public class **FindRemote**
extends Object

FindRemote Class Implement the XML Mesaging Mandatory Test of the JSR144 specification.

Author:

Copyright 2002 Sun Microsystems Inc., Inc. All rights reserved.

Version:

1.5

Field Summary

public static	EVENT_TIMEOUT
public static	JNDI_FACT_CLASS_NAME
public static	JNDI_URL
public static	jvtSessionList
public static	LOG_FILE
public static	namingContext
public static	STDOUT_ENABLED
public static	TESTFILES_HOME
public static	util

Constructor Summary

public	FindRemote() Creates a new FindRemote object.
--------	--

Method Summary

void	addJvtSession (ArrayList jvtSessionList) DOCUMENT ME!
ArrayList	getRemote() DOCUMENT ME!
static void	main (String[] args) DOCUMENT ME!

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields**namingContext**

```
public static javax.naming.InitialContext namingContext
```

JNDI_FACT_CLASS_NAME

```
public static java.lang.String JNDI_FACT_CLASS_NAME
```

JNDI_URL

```
public static java.lang.String JNDI_URL
```

EVENT_TIMEOUT

```
public static int EVENT_TIMEOUT
```

TESTFILES_HOME

```
public static java.lang.String TESTFILES_HOME
```

LOG_FILE

```
public static java.lang.String LOG_FILE
```

STDOUT_ENABLED

```
public static boolean STDOUT_ENABLED
```

util

```
public static ossj.common.tck.Util util
```

jvtSessionList

```
public static java.util.ArrayList jvtSessionList
```

Constructors

FindRemote

```
public FindRemote()
```

Creates a new FindRemote object.

Throws:

Exception - DOCUMENT ME!

Methods

getRemote

```
public ArrayList getRemote()  
    throws Exception
```

DOCUMENT ME!

Returns:

DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

addJvtSession

```
public void addJvtSession(ArrayList jvtSessionList)  
    throws IOException
```

DOCUMENT ME!

Parameters:

jvtSessionList - DOCUMENT ME!

Throws:

IOException - DOCUMENT ME!

main

```
public static void main(String[] args)  
    throws Exception
```

DOCUMENT ME!

Parameters:

args - DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

ossj.common.tck

Class JVTTestBase

```

java.lang.Object
  |
  +- junit.framework.Assert
      |
      +- junit.framework.TestCase
          |
          +- ossj.common.tck.JVTTestBase
  
```

All Implemented Interfaces:

junit.framework.Test

Direct Known Subclasses:

[XmlTests](#), [TestMev](#), [TestManagedEntityKey](#), [TestJVTSessionSimple](#), [TestBasics](#), [CommonAPIXMLTests](#)

```

public class JVTTestBase
extends junit.framework.TestCase
  
```

JVTTestBase Class A Base Class for TTAPIJVTTTests This class is both a TT JVT Proxy and a Test Case. The TTAPIJVTTTests extends and use the services of this class. The value class implementations used in this class are the one retrieved by calling the make methods on the provider JVT Session Bean

Since:

December 2001

Author:

Copyright 2001-2002 Nortel Networks Limited. All rights reserved.

Version:

1.0

Field Summary

public static	debug
public	done
public static	EVENT_TIMEOUT
public	howmany_key
public	howmany_value
public static	JMS_FACTORY
public static	JNDI_FACT_CLASS_NAME
public static	JNDI_URL
public static	jvtSessionList
public static	LOG_FILE

public static	namingContext
public static	QUOTECHAR
public	result
public	return_failed_only
public static	runonce
public static	TESTFILES_HOME
public static	TOPIC_NAME
public static	util

Constructor Summary

public	JVTTestBase (String name)
--------	---

Method Summary

static void	assertTrue (String type, boolean c) overload to make life easier
static void	assertTrue (String attr, String type, boolean c) overload to make life easier
static void	assertTrueMsg (String msg, boolean c) overload to make life easier
static void	fail (String type) overload to make life easier
static void	fail (String attr, String type) overload to make life easier
void	getEnvironment () Set up the environment
javax.oss.JVTSession	getInstanceOf (String myClass) DOCUMENT ME!
void	init () Init the env, to be call by all inheritance
void	log (String s) Print the string to the log file
void	log (Throwable e) print the exception to the log file
junit.framework.TestResult	run () Start the test

runonce

```
public static boolean runonce
```

debug

```
public static boolean debug
```

namingContext

```
public static javax.naming.InitialContext namingContext
```

JNDI_FACT_CLASS_NAME

```
public static java.lang.String JNDI_FACT_CLASS_NAME
```

JNDI_URL

```
public static java.lang.String JNDI_URL
```

(continued from last page)

EVENT_TIMEOUT

```
public static int EVENT_TIMEOUT
```

TESTFILES_HOME

```
public static java.lang.String TESTFILES_HOME
```

LOG_FILE

```
public static java.lang.String LOG_FILE
```

JMS_FACTORY

```
public static java.lang.String JMS_FACTORY
```

TOPIC_NAME

```
public static java.lang.String TOPIC_NAME
```

jvtSessionList

```
public static java.util.ArrayList jvtSessionList
```

util

```
public static ossj.common.tck.Util util
```

QUOTECHAR

```
public static int QUOTECHAR
```

done

```
public boolean done
```

(continued from last page)

howmany_key

```
public int howmany_key
```

howmany_value

```
public int howmany_value
```

return_failed_only

```
public boolean return_failed_only
```

result

```
public junit.framework.TestResult result
```

Constructors

JVTTestBase

```
public JVTTestBase(String name)
```

Methods

run

```
public junit.framework.TestResult run()
```

Start the test

Returns:

the test result

run

```
public void run(junit.framework.TestResult result)
```

Start the test

Parameters:

result - where to put test result

log

```
public void log(String s)
```

Print the string to the log file

(continued from last page)

Parameters:s - text to add

log

```
public void log(Throwable e)
```

print the exception to the log file

Parameters:e - exception to trace

getEnvironment

```
public void getEnvironment()
```

Set up the environment

init

```
public void init()
```

Init the env, to be call by all inheritance

getInstanceOf

```
public javax.oss.JVTSession getInstanceOf(String myClass)  
throws TCKException
```

DOCUMENT ME!

Parameters:

myClass - DOCUMENT ME!

Returns:

DOCUMENT ME!

Throws:[TCKException](#) - DOCUMENT ME!

assertTrue

```
public static void assertTrue(String attr,  
                               String type,  
                               boolean c)
```

overload to make life easier

assertTrue

```
public static void assertTrue(String type,  
                               boolean c)
```

overload to make life easier

assertTrueMsg

```
public static void assertTrueMsg(String msg,  
                                   boolean c)
```

(continued from last page)

overload to make life easier

fail

```
public static void fail(String attr,  
                        String type)
```

overload to make life easier

fail

```
public static void fail(String type)
```

overload to make life easier

ossj.common.tck Class SafeProperties

```

java.lang.Object
  |
  +- java.util.Dictionary
        |
        +- java.util.Hashtable
              |
              +- java.util.Properties
                    |
                    +- ossj.common.tck.SafeProperties
  
```

All Implemented Interfaces:

Serializable, Cloneable, Map

Direct Known Subclasses:

[TestProps](#), [SysProps](#)

public class **SafeProperties**
extends Properties

Properties subclass which ensures that required properties are available

Author:

BanuPrasad

Version:

Fields inherited from class java.util.Properties

defaults

Constructor Summary

public	SafeProperties (String propFilename, String[] keys) creates a new SafeProperties instance
--------	--

Method Summary

boolean	containsAllKeys (String[] keys) checks if the property file contains all specified keys
---------	--

Methods inherited from class java.util.Properties

getProperty, getProperty, list, list, load, loadFromXML, propertyNames, save, setProperty, store, storeToXML, storeToXML

Methods inherited from class java.util.Hashtable

clear, clone, contains, containsKey, containsValue, elements, entrySet, equals, get, hashCode, isEmpty, keys, keySet, put, putAll, rehash, remove, size, toString, values

Methods inherited from class java.util.Dictionary

elements, get, isEmpty, keys, put, remove, size

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface `java.util.Map`

`clear`, `containsKey`, `containsValue`, `entrySet`, `equals`, `get`, `hashCode`, `isEmpty`, `keySet`, `put`, `putAll`, `remove`, `size`, `values`

Constructors

SafeProperties

```
public SafeProperties(String propFilename,  
                     String[] keys)
```

creates a new `SafeProperties` instance

Parameters:

`propFilename` - The file to read the properties from
`keys` - The list of keys which must be present in the property file

Throws:

`IOException` - If the file cannot be read
`IllegalArgumentException` - If a required property is missing

Methods

containsAllKeys

```
public boolean containsAllKeys(String[] keys)
```

checks if the property file contains all specified keys

Parameters:

`keys` - The list of keys

Returns:

If the object contains all specified keys

ossj.common.tck Class Starter

java.lang.Object

└-ossj.common.tck.Starter

public class **Starter**
extends Object

Author:
aebbert

Version:

Nested Class Summary

class	Starter.StreamRedirector Starter.StreamRedirector
class	Starter.SuiteStarter Starter.SuiteStarter

Field Summary

public static	stopAtFirstFailed
---------------	-----------------------------------

Constructor Summary

public	Starter (String[] args) Creates new Starter
--------	--

Method Summary

static void	main (String[] args)
void	startTck () Start the test suite

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

stopAtFirstFailed

public static boolean **stopAtFirstFailed**

Constructors

Starter

```
public Starter(String[] args)
```

Creates new Starter

Methods

startTck

```
public void startTck()  
    throws Exception
```

Start the test suite

Throws:

Exception - any

main

```
public static void main(String[] args)
```

Parameters:

args - the command line arguments

ossj.common.tck Class Starter.SuiteStarter

```

java.lang.Object
  |
  +- java.lang.Thread
        |
        +- ossj.common.tck.Starter.SuiteStarter
  
```

All Implemented Interfaces:
Runnable

public class **Starter.SuiteStarter**
extends Thread

Fields inherited from class java.lang.Thread

MAX_PRIORITY, MIN_PRIORITY, NORM_PRIORITY

Constructor Summary

public	Starter.SuiteStarter (String[] command)
--------	---

Method Summary

void	destroyProcess ()
------	-----------------------------------

boolean	didRan ()
---------	---------------------------

InputStream	getErrorStream ()
-------------	-----------------------------------

OutputStream	getInputStream () This is the processe' OutputStream!!!
--------------	--

InputStream	getOutputStream () This is the process' InputStream!!!
-------------	---

void	run ()
------	------------------------

Methods inherited from class java.lang.Thread

activeCount, checkAccess, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, run, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

Methods inherited from interface `java.lang.Runnable`

```
run
```

Constructors

Starter.SuiteStarter

```
public Starter.SuiteStarter(String[] command)
```

Methods

run

```
public void run()
```

didRan

```
public boolean didRan()
```

destroyProcess

```
public void destroyProcess()
```

getInputStream

```
public OutputStream getInputStream()
```

This is the processe' OutputStream!!!

getErrorStream

```
public InputStream getErrorStream()
```

getOutputStream

```
public InputStream getOutputStream()
```

This is the process' InputStream!!!

ossj.common.tck

Class Starter.StreamRedirector

```

java.lang.Object
  |
  +- java.lang.Thread
        |
        +- ossj.common.tck.Starter.StreamRedirector

```

All Implemented Interfaces:
Runnable

public class **Starter.StreamRedirector**
extends Thread

Fields inherited from class java.lang.Thread

MAX_PRIORITY, MIN_PRIORITY, NORM_PRIORITY

Constructor Summary

public	Starter.StreamRedirector (InputStream source, OutputStream dest, String prefix, junit.framework.TestSuite suite)
--------	--

Method Summary

void	run ()
------	------------------------

void	stopGentle ()
------	-------------------------------

Methods inherited from class java.lang.Thread

activeCount, checkAccess, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, run, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface java.lang.Runnable

run

(continued from last page)

Constructors

Starter.StreamRedirector

```
public Starter.StreamRedirector(InputStream source,  
                                OutputStream dest,  
                                String prefix,  
                                junit.framework.TestSuite suite)
```

Methods

run

```
public void run()
```

stopGentle

```
public void stopGentle()
```

ossj.common.tck Interface StringTable

public interface **StringTable**
extends

Author:
BanuPrasad Dhanakoti
Version:

Field Summary

public static final	ATTRIBUTE Value: attribute:
public static final	INVALIDATION_PREFIX Value: _#INVALIDATION_PREFIX#_
public static final	NEGATIVE_TEST_PREFIX Value: test_Negative_
public static final	POSITIVE_TEST_PREFIX Value: test_Positive_
public static final	REMOVE_EVENT_DEFAULT Value: null
public static final	SYSPROPERTYFILE Value: tck_properties.txt
public static final	TEST_PREFIX Value: test_
public static final	TEXTILLEGALARGUMENT Value: IllegalArgument
public static final	TYPE Value: type:
public static final	UNKNOWN Value: unknown
public static final	WORDS Value: Words:

(continued from last page)

Fields

SYSPROPERTYFILE

```
public static final java.lang.String SYSPROPERTYFILE
```

Constant value: **tck_properties.txt**

TEXTILLEGALARGUMENT

```
public static final java.lang.String TEXTILLEGALARGUMENT
```

Constant value: **IllegalArgument**

UNKNOWN

```
public static final java.lang.String UNKNOWN
```

Constant value: **unknown**

ATTRIBUTE

```
public static final java.lang.String ATTRIBUTE
```

Constant value: **attribute:**

TYPE

```
public static final java.lang.String TYPE
```

Constant value: **type:**

TEST_PREFIX

```
public static final java.lang.String TEST_PREFIX
```

Constant value: **test_**

POSITIVE_TEST_PREFIX

```
public static final java.lang.String POSITIVE_TEST_PREFIX
```

Constant value: **test_Positive_**

NEGATIVE_TEST_PREFIX

```
public static final java.lang.String NEGATIVE_TEST_PREFIX
```

Constant value: **test_Negative_**

(continued from last page)

INVALIDATION_PREFIX

```
public static final java.lang.String INVALIDATION_PREFIX
```

Constant value: `_#INVALIDATION_PREFIX#_`

WORDS

```
public static final java.lang.String WORDS
```

Constant value: `Words:`

REMOVE_EVENT_DEFAULT

```
public static final java.lang.String REMOVE_EVENT_DEFAULT
```

Constant value: `null`

ossj.common.tck Class SysProps

```

java.lang.Object
├-- java.util.Dictionary
│   ├── java.util.Hashtable
│   │   ├── java.util.Properties
│   │   │   ├── ossj.common.tck.SafeProperties
│   │   │   └── ossj.common.tck.SysProps

```

All Implemented Interfaces:

Serializable, Cloneable, Map

public class **SysProps**
extends [SafeProperties](#)

System property names

Author:

BanuPrasad

Version:

Field Summary

public static final	EVENTTIMEOUT (required property) property name of the event timeout Value: event_timeout
public static final	JARNAME (required property) property name for the Jar name to be introspect Value: jar_name
public static final	JNDIFACTORY (required property) property name of the jndi factory name Value: jndi_factory_class
public static final	JNDIURL (required property) property name of the url of the jndi service Value: jndi_url
public static final	JVTTOPICJNDINAME (required property) property name of the queue name where respnses are expected Value: jvt_topic_jndi_name
public static final	LOGFILE (required property) property name for logging trace Value: log_file
public static final	LOGGING_ENABLED (required property) property name for tracing Value: logging_enabled

public static final	<u>MAXSTRINGLENGTH</u> (required property) property name of the maximum string length Value: max_string_length
public static final	<u>QUEUEFACTORYJNDINAME</u> (required property) property name of the queue name where respnses are expected Value: queue_factory_jndi_name
public static final	<u>REQUESTQUEUEJNDINAME</u> (required property) property name of the queue name where respnses are expected Value: request_queue_jndi_name
public static final	<u>RESPONSEQUEUEJNDINAME</u> (required property) property name of the queue name where respnses are expected Value: response_queue_jndi_name
public static final	<u>TOPICFACTORYJNDINAME</u> (required property) property name of the queue name where respnses are expected Value: topic_factory_jndi_name
public static final	<u>TOTALTIMEOUT</u> (required property) property name of the total timeout for a event Value: total_timeout
public static final	<u>XMLSCHEMANAME</u> (required property) property name for the Jar name to be introspect Value: xml_schema_name
public static final	<u>XVTTOPICJNDINAME</u> (required property) property name of the queue name where respnses are expected Value: xvt_topic_jndi_name

Fields inherited from class java.util.Properties

defaults

Constructor Summary

public	<u>SysProps</u> (String propFilename) creates a new SysProps instance
--------	--

Methods inherited from class [ossj.common.tck.SafeProperties](#)[containsAllKeys](#)**Methods inherited from class** java.util.Properties

getProperty, getProperty, list, list, load, loadFromXML, propertyNames, save, setProperty, store, storeToXML, storeToXML

Methods inherited from class java.util.Hashtable

clear, clone, contains, containsKey, containsValue, elements, entrySet, equals, get, hashCode, isEmpty, keys, keySet, put, putAll, rehash, remove, size, toString, values

Methods inherited from class java.util.Dictionary

elements, get, isEmpty, keys, put, remove, size

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.util.Map`

clear, containsKey, containsValue, entrySet, equals, get, hashCode, isEmpty, keySet, put, putAll, remove, size, values

Fields

LOGGING_ENABLED

```
public static final java.lang.String LOGGING_ENABLED
```

(required property) property name for tracing
Constant value: **logging_enabled**

LOGFILE

```
public static final java.lang.String LOGFILE
```

(required property) property name for logging trace
Constant value: **log_file**

JARNAME

```
public static final java.lang.String JARNAME
```

(required property) property name for the Jar name to be introspect
Constant value: **jar_name**

XMLSCHEMANAME

```
public static final java.lang.String XMLSCHEMANAME
```

(required property) property name for the Jar name to be introspect
Constant value: **xml_schema_name**

JNDIFACTORY

```
public static final java.lang.String JNDIFACTORY
```

(required property) property name of the jndi factory name
Constant value: **jndi_factory_class**

JNDIURL

```
public static final java.lang.String JNDIURL
```

(required property) property name of the url of the jndi service
Constant value: **jndi_url**

TOPICFACTORYJNDINAME

public static final java.lang.String **TOPICFACTORYJNDINAME**

(required property) property name of the queue name where respnses are expected
Constant value: **topic_factory_jndi_name**

JVTTOPICJNDINAME

public static final java.lang.String **JVTTOPICJNDINAME**

(required property) property name of the queue name where respnses are expected
Constant value: **jvt_topic_jndi_name**

XVTTOPICJNDINAME

public static final java.lang.String **XVTTOPICJNDINAME**

(required property) property name of the queue name where respnses are expected
Constant value: **xvt_topic_jndi_name**

QUEUEFACTORYJNDINAME

public static final java.lang.String **QUEUEFACTORYJNDINAME**

(required property) property name of the queue name where respnses are expected
Constant value: **queue_factory_jndi_name**

REQUESTQUEUEJNDINAME

public static final java.lang.String **REQUESTQUEUEJNDINAME**

(required property) property name of the queue name where respnses are expected
Constant value: **request_queue_jndi_name**

RESPONSEQUEUEJNDINAME

public static final java.lang.String **RESPONSEQUEUEJNDINAME**

(required property) property name of the queue name where respnses are expected
Constant value: **response_queue_jndi_name**

MAXSTRINGLENGTH

public static final java.lang.String **MAXSTRINGLENGTH**

(required property) property name of the maximum string length
Constant value: **max_string_length**

EVENTTIMEOUT

public static final java.lang.String **EVENTTIMEOUT**

(required property) property name of the event timeout
Constant value: **event_timeout**

(continued from last page)

TOTALTIMEOUT

```
public static final java.lang.String TOTALTIMEOUT
```

(required property) property name of the total timeout for a event
Constant value: `total_timeout`

Constructors

SysProps

```
public SysProps(String propFilename)
```

creates a new SysProps instance

Parameters:

`propFilename` - The file containing the properties

Throws:

`IOException` - If the file cannot be read

`IllegalArgumentException` - If not all required properties are present

ossj.common.tck Class TCKException

```

java.lang.Object
  |
  +- java.lang.Throwable
        |
        +- java.lang.Exception
              |
              +- ossj.common.tck.TCKException
  
```

All Implemented Interfaces:
Serializable

```

public class TCKException
extends Exception
  
```

Simple class to provide more detailed information for unexpected implementation exceptions The implementation exception is wrapped, a description added an the new TCKException thrown

Author:
BanuPrasad

Version:
0.1

Constructor Summary

public	TCKException (String msg) Creates new TCKException
public	TCKException (String msg, Throwable nested) Creates new TCKException
public	TCKException (String name, String type, Exception nested) Creates new TCKException

Method Summary

void	printStackTrace () DOCUMENT ME!
void	printStackTrace (PrintStream s) DOCUMENT ME!
void	printStackTrace (PrintWriter s) DOCUMENT ME!

Methods inherited from class java.lang.Throwable

fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

TCKException

```
public TCKException(String msg)
```

Creates new TCKException

Parameters:

msg - the exception msg

TCKException

```
public TCKException(String msg,  
                    Throwable nested)
```

Creates new TCKException

Parameters:

msg - the exception msg

nested - The Exception thrown by the implementation

TCKException

```
public TCKException(String name,  
                    String type,  
                    Exception nested)
```

Creates new TCKException

Parameters:

name - The name of the current attribute

type - The type of the current ManagedEntityValue

nested - The Exception thrown by the implementation

Methods

printStackTrace

```
public void printStackTrace()
```

DOCUMENT ME!

printStackTrace

```
public void printStackTrace(PrintStream s)
```

DOCUMENT ME!

Parameters:

s - DOCUMENT ME!

printStackTrace

```
public void printStackTrace(PrintWriter s)
```

(continued from last page)

DOCUMENT ME!

Parameters:

s - DOCUMENT ME!

ossj.common.tck Class TestProps

```

java.lang.Object
├-- java.util.Dictionary
│   ├── java.util.Hashtable
│   │   ├── java.util.Properties
│   │   │   ├── ossj.common.tck.SafeProperties
│   │   │   └-- ossj.common.tck.TestProps

```

All Implemented Interfaces:

Serializable, Cloneable, Map

```

public class TestProps
extends SafeProperties

```

Test property names

Author:

BanuPrasad

Version:

Field Summary

public static final	MAX_BULK_ELEMENTS (required property) property name of the max no of elements per bulk operation Value: max_bulk_elements
public static final	NOOFRUNS (required property) property name of the number of runs per value object Value: no_of_runs
public static final	SEQUENCES (required property) property name of the no of complete sequence runs Value: sequences

Fields inherited from class java.util.Properties

defaults

Constructor Summary

public	TestProps (String propFilename) creates a new TestProps instance
--------	---

Methods inherited from class [ossj.common.tck.SafeProperties](#)

[containsAllKeys](#)

Methods inherited from class java.util.Properties

```
getProperty, getProperty, list, list, load, loadFromXML, propertyNames, save,
setProperty, store, storeToXML, storeToXML
```

Methods inherited from class `java.util.Hashtable`

```
clear, clone, contains, containsKey, containsValue, elements, entrySet, equals, get,
hashCode, isEmpty, keys, keySet, put, putAll, rehash, remove, size, toString, values
```

Methods inherited from class `java.util.Dictionary`

```
elements, get, isEmpty, keys, put, remove, size
```

Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

Methods inherited from interface `java.util.Map`

```
clear, containsKey, containsValue, entrySet, equals, get, hashCode, isEmpty, keySet,
put, putAll, remove, size, values
```

Fields

NOOFRUNS

```
public static final java.lang.String NOOFRUNS
```

(required property) property name of the number of runs per value object
Constant value: **no_of_runs**

SEQUENCES

```
public static final java.lang.String SEQUENCES
```

(required property) property name of the no of complete sequence runs
Constant value: **sequences**

MAX_BULK_ELEMENTS

```
public static final java.lang.String MAX_BULK_ELEMENTS
```

(required property) property name of the max no of elements per bulk operation
Constant value: **max_bulk_elements**

Constructors

TestProps

```
public TestProps(String propFilename)
```

creates a new TestProps instance

Parameters:

propFilename - The file containing the properties

(continued from last page)

Throws:

`IOException` - If the file cannot be read

`IllegalArgumentException` - If not all required properties are present

ossj.common.tck Class Util

java.lang.Object

└-ossj.common.tck.Util

public class **Util**
extends Object

Collection of useful methods & metadata

Author:

BanuPrasad Dhanakoti

Version:

Field Summary

public static	outputFile for log
public final	RANDOM used for random generator

Constructor Summary

public	Util() Creates new util
public	Util(boolean generic) Creates a new Util object.

Method Summary

static boolean	contains (int[] a, int key)
static boolean	contains (Object[] a, Object key)
static boolean	containsDuplicates (Vector v, javax.oss.ManagedEntityKey key)
javax.jms.QueueReceiver	createQueueReceiver (String msgSelector) DOCUMENT ME!
javax.jms.QueueSender	createQueueSender () DOCUMENT ME!
javax.jms.QueueSession	createQueueSession () DOCUMENT ME!
String[]	deepCopy (String[] src) performs a deep copy on src
static String[]	diff (String[] first, String[] second) gets first minus second i.e.

static int	find (ArrayList list, javax.oss.ManagedEntityKey key)
static int	find (javax.oss.ManagedEntityKey[] keys, javax.oss.ManagedEntityKey key)
static int	find (Vector v, javax.oss.ManagedEntityKey key)
ArrayList	findImplementationOf (Class myInterface) DOCUMENT ME!
ArrayList	findInterfaceStartsWith (String s)
Class	getAttributeType (javax.oss.AttributeAccess mev, String name)
long	getEventTimeout ()
Vector	getIllegalKeys (javax.oss.ManagedEntityValue mev)
static int[]	getIllegalValue (int[] vals, int count) gets a value which is different from all input values and not null input array members must not be null
static String[]	getIllegalValue (String[] vals, int count) gets a value which is different from all input values and not null input array members must not be null
javax.naming.InitialContext	getInitialContext ()
Hashtable	getManagedEntityKeys (javax.oss.JVTSession jas)
static Object	getMaxAsObject (Class cls)
static Object	getMinAsObject (Class cls)
String	getProperty (String prop) DOCUMENT ME!
String	getQueueFactoryName ()
String	getRequestQueueName ()
String	getTopicFactoryName () return the name of the topic factory from the properties file
String	getTopicName () return the name of the topic from the properties file
javax.jms.TopicSubscriber	getTopicSubscriber ()
javax.jms.TopicSubscriber	getTopicSubscriber (String msgSelector)
long	getTotalTimeout () The timeout to wait for all events

static String[]	<u>intersect</u> (String[] first, String[] second)
static boolean	<u>isSameType</u> (javax.oss.ManagedEntityValue first, javax.oss.ManagedEntityValue second)
static boolean	<u>isSubset</u> (String[] subSet, String[] superSet)
String	<u>isValidJndiName</u> (String jndiName) Test if the given JNDI name follows: System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Comp/name or System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Resource/name
void	<u>log</u> (String s) DOCUMENT ME!
void	<u>log</u> (Throwable e) DOCUMENT ME!
static String	<u>makeCSList</u> (Object[] items) makes a comma separated list of the items String representation
Date	<u>randomDate</u> ()
Object	<u>randomElement</u> (Object[] a)
javax.oss.AttributeAccess	<u>randomManagedEntityValueAttributes</u> (javax.oss.AttributeAccess mev) Creates a random value for the given ManagedEntityValue
javax.oss.MultiValueList	<u>randomMultiValue</u> (Class cls) DOCUMENT ME!
String	<u>randomString</u> ()
String[]	<u>randomSubset</u> (String[] set)
ArrayList	<u>randomSubsetNotEmpty</u> (Object[] set)
String[]	<u>randomSubsetNotEmpty</u> (String[] set)
Object	<u>randomValue</u> (String name, javax.oss.AttributeAccess mev) Creates a random value for the given attribute name and ManagedEntityValue So far, Date and String are the only non primitive types supported.
Object	<u>randomValueInstance</u> (Class cls) DOCUMENT ME!
void	<u>shuffle</u> (Object[] a) shuffles the array

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

outputFile

```
public static java.io.PrintStream outputFile
```

for log

RANDOM

```
public final java.util.Random RANDOM
```

used for random generator

Constructors

Util

```
public Util()
```

Creates new util

Util

```
public Util(boolean generic)
```

Creates a new Util object.

Parameters:

generic - DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

Methods

log

```
public void log(String s)
```

DOCUMENT ME!

Parameters:

s - DOCUMENT ME!

log

```
public void log(Throwable e)
```

DOCUMENT ME!

Parameters:

e - DOCUMENT ME!

(continued from last page)

getEventTimeout

```
public long getEventTimeout()
```

Returns:

The timeout to wait for an event

getTotalTimeout

```
public long getTotalTimeout()
```

The timeout to wait for all events

getAttributeType

```
public Class getAttributeType(javax.oss.AttributeAccess mev,  
    String name)  
    throws Exception
```

Returns:

The type of the attribute. For indexed attributes, the element type is returned. If an attribute of that name doesn't exist, null is returned

getInitialContext

```
public javax.naming.InitialContext getInitialContext()  
    throws TCKException
```

Returns:

A new InitialContext for the server

getTopicFactoryName

```
public String getTopicFactoryName()
```

return the name of the topic factory from the properties file

getQueueFactoryName

```
public String getQueueFactoryName()
```

getRequestQueueName

```
public String getRequestQueueName()
```

getTopicName

```
public String getTopicName()
```

(continued from last page)

return the name of the topic from the properties file

Returns:

DOCUMENT ME!

isValidJndiName

```
public String isValidJndiName(String jndiName)
```

Test if the given JNDI name follows: System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Comp/name or System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Resource/name

Parameters:

name - the jndi name to test

Returns:

null if the name is valid or the text error found

getTopicSubscriber

```
public javax.jms.TopicSubscriber getTopicSubscriber()  
throws Exception
```

Returns:

A new TopicSubscriber on a new TopicConnection to the topic SysProps.TOPICNAME Messages sent by this connection are received

getTopicSubscriber

```
public javax.jms.TopicSubscriber getTopicSubscriber(String msgSelector)  
throws TCKException
```

Parameters:

msgSelector - The selector for incoming messages

Returns:

A new TopicSubscriber on a new TopicConnection to the topic SysProps.TOPICNAME Messages sent by this connection are received

createQueueSession

```
public javax.jms.QueueSession createQueueSession()  
throws TCKException
```

DOCUMENT ME!

Returns:

DOCUMENT ME!

Throws:

[TCKException](#) - DOCUMENT ME!

(continued from last page)

createQueueSender

```
public javax.jms.QueueSender createQueueSender( )  
    throws TCKException
```

DOCUMENT ME!

Returns:

DOCUMENT ME!

Throws:

[TCKException](#) - DOCUMENT ME!

createQueueReceiver

```
public javax.jms.QueueReceiver createQueueReceiver(String msgSelector)  
    throws TCKException
```

DOCUMENT ME!

Parameters:

msgSelector - DOCUMENT ME!

Returns:

DOCUMENT ME!

Throws:

[TCKException](#) - DOCUMENT ME!

intersect

```
public static String[] intersect(String[] first,  
    String[] second)
```

Returns:

The intersection of first and second

isSubset

```
public static boolean isSubset(String[] subSet,  
    String[] superSet)
```

Parameters:

subSet - The subset

superSet - The superset

Returns:

true if both parameters are not null and subSet is subset of superSet

diff

```
public static String[] diff(String[] first,  
    String[] second)
```

gets first minus second i.e. all Strings of first which are not in second

(continued from last page)

Parameters:

first
second

Returns:

The difference of first and second

contains

```
public static boolean contains(Object[] a,  
    Object key)  
    throws IllegalArgumentException
```

Returns:

if a contains key
nota bene: all buckets of a must not be null

Throws:

java.lang.IllegalArgumentException - If an argument is null

contains

```
public static boolean contains(int[] a,  
    int key)  
    throws IllegalArgumentException
```

Parameters:

a
key

Returns:

if a contains key

Throws:

java.lang.IllegalArgumentException - If an argument is null

find

```
public static int find(ArrayList list,  
    javax.oss.ManagedEntityKey key)
```

Parameters:

list - List of Keys
key - The Key to be looked for

Returns:

Index where key was found, -1 otherwise

find

```
public static int find(Vector v,  
    javax.oss.ManagedEntityKey key)
```


(continued from last page)

Parameters:

v - Vector of Keys
key - The Key to be looked for

Returns:

Index where key was found, -1 otherwise

find

```
public static int find(javax.oss.ManagedEntityKey[] keys,  
    javax.oss.ManagedEntityKey key)
```

Parameters:

list - List of Keys
key - The Key to be looked for

Returns:

Index where key was found, -1 otherwise

containsDuplicates

```
public static boolean containsDuplicates(Vector v,  
    javax.oss.ManagedEntityKey key)
```

Parameters:

keys
key

Returns:

If the keys contains duplicates of key

makeCSList

```
public static String makeCSList(Object[] items)
```

makes a comma separated list of the items String representation

Parameters:

items - The items

Returns:

The comma separated list

deepCopy

```
public String[] deepCopy(String[] src)
```

performs a deep copy on src

Parameters:

src - The source to be copied

Returns:

The deep copy

shuffle

```
public void shuffle(Object[] a)  
    throws IllegalArgumentException
```

shuffles the array

Throws:

IllegalArgumentException - If a is null

randomSubset

```
public String[] randomSubset(String[] set)
```

Parameters:

set - The set of strings

Returns:

A random subset of set

randomSubsetNotEmpty

```
public String[] randomSubsetNotEmpty(String[] set)
```

Parameters:

set - The set of String

Returns:

A random subset of set

randomSubsetNotEmpty

```
public ArrayList randomSubsetNotEmpty(Object[] set)
```

Parameters:

set - The set

Returns:

A random subset of set

getMinAsObject

```
public static Object getMinAsObject(Class cls)  
    throws IllegalArgumentException
```

Parameters:

The - class to get the minimum from

Returns:

The minimum wrapped in the related java class

(continued from last page)

Throws:`IllegalArgumentException` - If `!cls.IsPrimitive()`

getMaxAsObject

```
public static Object getMaxAsObject(Class cls)
    throws IllegalArgumentException
```

Parameters:

The - class to get the maximum from

Returns:

The maximum wrapped in the related java class

Throws:

`java.lang.IllegalArgumentException` - If `!cls.IsPrimitive()`

getIllegalValue

```
public static String[] getIllegalValue(String[] vals,
    int count)
    throws IllegalArgumentException
```

gets a value which is different from all input values and not null input array members must not be null

Parameters:

`vals` - The values to be different from

`count` - The number of values to return

Returns:

The different values

getIllegalValue

```
public static int[] getIllegalValue(int[] vals,
    int count)
    throws IllegalArgumentException
```

gets a value which is different from all input values and not null input array members must not be null

Parameters:

`vals` - The values to be different from

`count` - The number of values to return

Returns:

The different values

randomDate

```
public Date randomDate()
```

Returns:

A random Date

randomString

```
public String randomString()
```

Returns:

A random String which is not null

randomValue

```
public Object randomValue(String name,  
    javax.oss.AttributeAccess mev)  
    throws Exception
```

Creates a random value for the given attribute name and ManagedEntityValue So far, Date and String are the only non primitive types supported. Other non primitive type must be creatable via newInstance

Parameters:

name - The name of the attribute

mev - The ManagedEntityValue

Returns:

The randomly created value

randomManagedEntityValueAttributes

```
public javax.oss.AttributeAccess  
randomManagedEntityValueAttributes(javax.oss.AttributeAccess mev)  
    throws Exception
```

Creates a random value for the given ManagedEntityValue

Parameters:

mev - The ManagedEntityValue

Returns:

The randomly created ManagedEntityValue

randomMultiValue

```
public javax.oss.MultiValueList randomMultiValue(Class cls)  
    throws Exception
```

DOCUMENT ME!

Parameters:

cls - DOCUMENT ME!

Returns:

DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

(continued from last page)

randomValueInstance

```
public Object randomValueInstance(Class cls)
    throws Exception
```

DOCUMENT ME!

Parameters:

cls - DOCUMENT ME!

Returns:

DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

isSameType

```
public static boolean isSameType(javax.oss.ManagedEntityValue first,
    javax.oss.ManagedEntityValue second)
```

Parameters:

first
second

Returns:

If both MEVs are of the same type

getManagedEntityKeys

```
public Hashtable getManagedEntityKeys(javax.oss.JVTSession jas)
    throws Exception
```

Returns:

A Hashtable containing a legal (but removed) Key per Type

getIllegalKeys

```
public Vector getIllegalKeys(javax.oss.ManagedEntityValue mev)
    throws Exception
```

Returns:

a Hashtable of Vector of illegal Keys per Type

randomElement

```
public Object randomElement(Object[] a)
```

Parameters:

a - Array to select from

(continued from last page)

Returns:Random element of the array. null if array is empty or null

getProperty

```
public String getProperty(String prop)
```

DOCUMENT ME!

Parameters:

prop - DOCUMENT ME!

Returns:DOCUMENT ME!

findInterfaceStartsWith

```
public ArrayList findInterfaceStartsWith(String s)
```

findImplementationOf

```
public ArrayList findImplementationOf(Class myInterface)
```

DOCUMENT ME!

Parameters:

myInterface - DOCUMENT ME!

Returns:DOCUMENT ME!

Package
ossj.common.tck.tests

ossj.common.tck.tests

Class TestBasics

```

java.lang.Object
  |
  +- junit.framework.Assert
      |
      +- junit.framework.TestCase
          |
          +- ossj.common.tck.JVTTestBase
              |
              +- ossj.common.tck.tests.TestBasics
  
```

All Implemented Interfaces:

junit.framework.Test

public class **TestBasics**
 extends [JVTTestBase](#)

This test checks basic functionality of the JVTSession and the JVTActivationSessionHome as it is needed by all other test suites

Author:

Vincent Perrot, Sun Microsystems Inc.

Version:

1.1

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Constructor Summary

public	TestBasics (String name) Creates new testJVTActivationSession
--------	--

Method Summary

void	setUp ()
static junit.framework.TestS uite	suite () Basic tests:
void	tearDown ()
void	test_GetInitialContext () verify if the InitialContext can be created.
void	test_GetJVTSession () tries to get an JVTActivationSession implementation
void	test_GetLegalKeys () Verify the key factory using the given supported managed entity types.

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#), [init](#), [log](#), [log](#), [run](#), [run](#)

```
countTestCases, createResult, getName, run, run, runBare, runTest, setName, setUp,
tearDown, toString
```

```
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertFalse, assertFalse, assertNotNull, assertNotNull,
assertNotSame, assertNotSame, assertNull, assertNull, assertEquals, assertEquals,
assertTrue, assertTrue, fail, fail
```

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

```
countTestCases, run
```

TestBasics

Creates new testJVTActivationSession

suite

Page 57 of 139

(continued from last page)

Basic tests:

- all the test from this class.

And tests from TestJVTSessionSimple

- test_GetManagedEntityTypes_ContainsACTIVATE: verify that the JVT support at least one managed Entity type using getManagedEntityTypes
- test_GetQueryTypes_IsNotNull: verify that JVT support at least one query type using getQueryTypes
- test_GetSupportedOperations_IsSubsetOfJVTSessionOption: just call getSupportedOptionalOperations
- test_JVT_JNDI_NAME: verify the JNDI name construction

Returns:

DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

setUp

```
protected void setUp()  
    throws Exception
```

tearDown

```
protected void tearDown()
```

test_GetInitialContext

```
public void test_GetInitialContext()  
    throws Exception
```

verify if the InitialContext can be created.
Validate the connection with the AS.

Throws:

Exception - DOCUMENT ME!

test_GetJVTSession

```
public void test_GetJVTSession()  
    throws Exception
```

tries to get an JVTActivationSession implementation

test_GetTopicSubscriber

```
public void test_GetTopicSubscriber()  
    throws Exception
```

Test if the declared Topic is reachable and validate the JNDI name used for the factory and topic.

Throws:

Exception - DOCUMENT ME!

test_GetLegalKeys

```
public void test_GetLegalKeys()  
    throws Exception
```

Verify the key factory using the given supported managed entity types.

Throws:

Exception - DOCUMENT ME!

ossj.common.tck.tests

Class TestCbeBasics

```

java.lang.Object
  |
  +- junit.framework.Assert
      |
      +- junit.framework.TestCase
          |
          +- ossj.common.tck.JVTTestBase
              |
              +- ossj.common.tck.tests.TestMev
                  |
                  +- ossj.common.tck.tests.TestCbeBasics
  
```

All Implemented Interfaces:

junit.framework.Test

public class **TestCbeBasics**
 extends [TestMev](#)

Author:

Vincent Perrot, Sun Microsystems Inc.

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Constructor Summary

public	TestCbeBasics (String name) Creates new testJVTActivationSession
--------	---

Method Summary

void	doGeneric (String testName) genericTest
void	setUp ()
static junit.framework.TestS uite	suite () Generate the test cases
void	tearDown ()
void	test_cbe_ClassNameAndTypeAndValueKeyPair () Test Value and Value type and key pattern
void	test_cbe_GetAttributeNames_ContainsKEY () checks weather the getattributes return key as attribute or not
void	test_cbe_GetAttributeNames_NamesStartWithLowerCase () tests that attributes

void	<u>test_cbe_GetOptionalAttributeNames_InAllAttributes()</u> tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter
void	<u>test_cbe_GetSettableAttributeNames_NotContainsKEY()</u> Checks if the KEY attribute is not settable
void	<u>test_cbe_GetSettableAttributeNames_SubsetOfGetAttributeNames()</u> tests that getSettableAttributesNames is a subset of getAttributeNames
void	<u>test_cbe_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()</u> tests that setAttributeValue(a) => a in getPopulatedAttributes()
void	<u>test_cbe_SetAttributeValue_IsPopulated()</u> tests that setAttributeValue(a) => isPopulated(a)
void	<u>test_cbe_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY()</u> tests that succesful setManagedEntityKey(val) implies KEY in getPopulatedAttributeNames()
void	<u>test_cbe_SetManagedEntityKey_IsPopulatedKEY()</u> tests that succesful setManagedEntityKey(val) implies isPopulated(KEY)
void	<u>test_cbe_UnpopulateAllAttributes_GetAttributeValueFailsForAll()</u> tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames
void	<u>test_cbe_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()</u> tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty
void	<u>test_cbe_UnpopulateAllAttributes_IsPopulatedFalseForAll()</u> tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()
void	<u>test_Negative_cbe_GetAttributeValue_AttributeNameIllegalValue()</u> tests failure of getAttributeValue(a) for a not in getAttributeNames();
void	<u>test_Negative_cbe_GetAttributeValue_AttributeNameNullValue()</u> tests failure of getAttributeValue(null)
void	<u>test_Negative_cbe_IsPopulated_AttributeNameIllegalValue()</u> tests failure of isPopulated(a) for String a not in getAttributeNames()
void	<u>test_Negative_cbe_IsPopulated_AttributeNameNullValue()</u> tests failure of isPopulated(null)
void	<u>test_Negative_cbe_SetAttributeValue_AttributeNameIllegalValue()</u> tests failure of setAttributeValue(a) for a not in getAttributeNames()
void	<u>test_Negative_cbe_SetAttributeValue_AttributeNameNullValue()</u> tests failure of setAttributeValue(a) for null
void	<u>test_Negative_cbe_SetAttributeValue_IllegalObject()</u> tests failure of setAttributeValue(a) for an illegal object()
void	<u>test_Negative_cbe_SetAttributeValue_PrimitiveNullValue()</u> tests Illegal null primitive attribute value
void	<u>test_Negative_cbe_UnpopulateAttribute_AttributeNameIllegalValue()</u> tests failure of unpopulateAttribute(a) for a not in getAttributeNames

void	<u>test_Negative_cbe_UnpopulateAttribute_AttributeNameNullValue()</u> tests failure of unpopulateAttribute(null)
void	<u>test_Positive_cbe_IsPopulated_AttributeNameAllDomainValues()</u> tests isPopulated for all attributes in getAttributeNames()
void	<u>test_Positive_cbe_ManagedEntityKey_IllegalKey()</u> checks for invalid managedentitykey
void	<u>test_Positive_cbe_SetAttributeValue_DomainValue()</u> this is a generic method to test the setattribute value function based on the type of the field
void	<u>test_Positive_cbe_SetAttributeValue_PrimitiveMINMAXValues()</u> tests setAttributeValue(min/max) for primitive attributes that are not enums
void	<u>test_Positive_cbe_SetManagedEntityKey_DomainValue()</u> tests a domain value
void	<u>test_Positive_cbe_testCloneEqualsImplementation()</u> Test clone and equals methods CBE implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true a.equals(b) == true b.equals(a) == true
void	<u>test_Positive_cbe_UnpopulateAttribute_AttributeNameAllDomainValues()</u> tests succesful invokation of unpopulateAttribute(a) for all a in getAttributeNames()
void	<u>test_Positive_getCbeImplementation()</u> Check presence of CBE interfaces implementations Search only in directoryies and jar file name starting with "oss" Expected Result: a hashtable completed with all CBE interfaces from test_Positive_getCbeInterfaces as key and ArrayList that contains all implementation Classes as value
void	<u>test_Positive_getCbeInterfaces()</u> Check presence of CBE interfaces Expected Result: a ArrayList completed with all CBE interfaces

Methods inherited from class [ossj.common.tck.tests.TestMev](#)

[setUp](#), [test_ClassNameAndTypeAndValueKeyPair](#), [test_GetAttributeNames_ContainsKEY](#),
[test_GetAttributeNames_NamesStartWithLowerCase](#),
[test_GetOptionalAttributeNames_InAllAttributes](#),
[test_GetSettableAttributeNames_NotContainsKEY](#),
[test_GetSettableAttributeNames_SubsetOfGetAttributeNames](#),
[test_Negative_GetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_GetAttributeValue_AttributeNameNullValue](#),
[test_Negative_IsPopulated_AttributeNameIllegalValue](#),
[test_Negative_IsPopulated_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_SetAttributeValue_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_IllegalObject](#),
[test_Negative_SetAttributeValue_PrimitiveNullValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameIllegalValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameNullValue](#),
[test_Positive_IsPopulated_AttributeNameAllDomainValues](#),
[test_Positive_ManagedEntityKey_IllegalKey](#),
[test_Positive_SetAttributeValue_DomainValue](#),
[test_Positive_SetAttributeValue_PrimitiveMINMAXValues](#),
[test_Positive_SetAttributeValueKEY_IllegalKey](#),
[test_Positive_SetManagedEntityKey_DomainValue](#),
[test_Positive_SetManagedEntityKey_IllegalKey](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues](#), [test_Query_Type](#),
[test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName](#),
[test_SetAttributeValue_IsPopulated](#),
[test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY](#),
[test_SetManagedEntityKey_IsPopulatedKEY](#),
[test_UnpopulateAllAttributes_GetAttributeValueFailsForAll](#),
[test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty](#),
[test_UnpopulateAllAttributes_IsPopulatedFalseForAll](#), [test_upv_Type](#)

Methods inherited from class [ossj.common.tck.JVTTestBase](#)

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#),
[init](#), [log](#), [log](#), [run](#), [run](#)

Methods inherited from class [junit.framework.TestCase](#)

[countTestCases](#), [createResult](#), [getName](#), [run](#), [run](#), [runBare](#), [runTest](#), [setName](#), [setUp](#),
[tearDown](#), [toString](#)

Methods inherited from class [junit.framework.Assert](#)

[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertFalse](#), [assertFalse](#), [assertNotNull](#), [assertNotNull](#),
[assertNotSame](#), [assertNotSame](#), [assertNull](#), [assertNull](#), [assertSame](#), [assertSame](#),
[assertTrue](#), [assertTrue](#), [fail](#), [fail](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#),
[wait](#)

Methods inherited from interface junit.framework.Test`countTestCases, run`

Constructors

TestCbeBasics

```
public TestCbeBasics(String name)
```

Creates new testJVTActivationSession

Methods

suite

```
public static junit.framework.TestSuite suite()  
    throws Exception
```

Generate the test cases

setUp

```
protected void setUp()  
    throws Exception
```

sets up the TestCase

tearDown

```
protected void tearDown()
```

test_Positive_getCbeInterfaces

```
public void test_Positive_getCbeInterfaces()  
    throws Exception
```

Check presence of CBE interfaces

Expected Result:

a ArrayList completed with all CBE interfaces

test_Positive_getCbeImplementation

```
public void test_Positive_getCbeImplementation()  
    throws Exception
```

Check presence of CBE interfaces implementations

Search only in directories and jar file name starting with "oss"

Expected Result:

a hashtable completed with all CBE interfaces from test_Positive_getCbeInterfaces as key and ArrayList that contains all implementation Classes as value

(continued from last page)

doGeneric

```
protected void doGeneric(String testName)
    throws Exception
```

genericTest

test_cbe_ClassNameAndTypeAndValueKeyPair

```
public void test_cbe_ClassNameAndTypeAndValueKeyPair()
    throws Exception
```

Test Value and Value type and key pattern

- test if VALUE_TYPE field is declared
- test if VALUE_TYPE field is public final static String
- test if the declared type exist (call forName)
- test if the class is of the declared type (decareType.isAssignableFrom)
- test if the VALUE_TYPE ends with "Value"
- test if the VALUE_TYPE corresponding interface exists

test_cbe_GetAttributeNames_NamesStartWithLowerCase

```
public void test_cbe_GetAttributeNames_NamesStartWithLowerCase()
    throws Exception
```

tests that attributes

- names have length > 0
- names start with lowercase letter
- get/set methods exist

test_cbe_GetOptionalAttributeNames_InAllAttributes

```
public void test_cbe_GetOptionalAttributeNames_InAllAttributes()
    throws Exception
```

tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter

test_Positive_cbe_IsPopulated_AttributeNameAllDomainValues

```
public void test_Positive_cbe_IsPopulated_AttributeNameAllDomainValues()
    throws Exception
```

tests isPopulated for all attributes in getAttributeNames()

test_Negative_cbe_IsPopulated_AttributeNameNullValue

```
public void test_Negative_cbe_IsPopulated_AttributeNameNullValue()
    throws Exception
```

tests failure of isPopulated(null)

(continued from last page)

test_Negative_cbe_IsPopulated_AttributeNameIllegalValue

```
public void test_Negative_cbe_IsPopulated_AttributeNameIllegalValue()  
    throws Exception
```

tests failure of isPopulated(a) for String a not in getAttributeNames()

test_cbe_GetSettableAttributeNames_SubsetOfGetAttributeNames

```
public void test_cbe_GetSettableAttributeNames_SubsetOfGetAttributeNames()  
    throws Exception
```

tests that getSettableAttributesNames is a subset of getAttributeNames

test_Positive_cbe_UnpopulateAttribute_AttributeNameAllDomainValues

```
public void test_Positive_cbe_UnpopulateAttribute_AttributeNameAllDomainValues()  
    throws Exception
```

tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()

test_Negative_cbe_UnpopulateAttribute_AttributeNameNullValue

```
public void test_Negative_cbe_UnpopulateAttribute_AttributeNameNullValue()  
    throws Exception
```

tests failure of unpopulatedAttribute(null)

test_Negative_cbe_UnpopulateAttribute_AttributeNameIllegalValue

```
public void test_Negative_cbe_UnpopulateAttribute_AttributeNameIllegalValue()  
    throws Exception
```

tests failure of unpopulatedAttribute(a) for a not in getAttributeNames

test_cbe_UnpopulateAllAttributes_IsPopulatedFalseForAll

```
public void test_cbe_UnpopulateAllAttributes_IsPopulatedFalseForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

test_cbe_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty

```
public void test_cbe_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()  
    throws Exception
```

tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty

test_cbe_UnpopulateAllAttributes_GetAttributeValueFailsForAll

```
public void test_cbe_UnpopulateAllAttributes_GetAttributeValueFailsForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames

(continued from last page)

test_Negative_cbe_SetAttributeValue_AttributeNameNullValue

```
public void test_Negative_cbe_SetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

tests failure of setAttributeValue(a) for null

test_Negative_cbe_SetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_cbe_SetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

tests failure of setAttributeValue(a) for a not in getAttributeNames()

test_Negative_cbe_GetAttributeValue_AttributeNameNullValue

```
public void test_Negative_cbe_GetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

tests failure of getAttributeValue(null)

test_Negative_cbe_GetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_cbe_GetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

tests failure of getAttributeValue(a) for a not in getAttributeValues();

test_Positive_cbe_SetManagedEntityKey_DomainValue

```
public void test_Positive_cbe_SetManagedEntityKey_DomainValue()  
    throws Exception
```

tests a domain value

test_cbe_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY

```
public void test_cbe_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY()  
    throws Exception
```

tests that succesful setManagedEntityKey(val) implies KEY in getPopulatedAttributeNames()

test_cbe_SetManagedEntityKey_IsPopulatedKEY

```
public void test_cbe_SetManagedEntityKey_IsPopulatedKEY()  
    throws Exception
```

tests that succesful setManagedEntityKey(val) implies isPopulated(KEY)

test_cbe_GetAttributeNames_ContainsKEY

```
public void test_cbe_GetAttributeNames_ContainsKEY()  
    throws Exception
```

checks weather the getattributes return key as attribute or not

(continued from last page)

test_cbe_GetSettableAttributeNames_NotContainsKEY

```
public void test_cbe_GetSettableAttributeNames_NotContainsKEY()  
    throws Exception
```

Checks if the KEY attribute is not settable

test_Positive_cbe_ManagedEntityKey_IllegalKey

```
public void test_Positive_cbe_ManagedEntityKey_IllegalKey()  
    throws Exception
```

checks for invalid managedentitykey

test_Positive_cbe_SetAttributeValue_PrimitiveMINMAXValues

```
public void test_Positive_cbe_SetAttributeValue_PrimitiveMINMAXValues()  
    throws Exception
```

tests setAttributeValue(min/max) for primitive attributes that are not enums

test_Negative_cbe_SetAttributeValue_PrimitiveNullValue

```
public void test_Negative_cbe_SetAttributeValue_PrimitiveNullValue()  
    throws Exception
```

tests Illegal null primitive attribute value

test_Positive_cbe_SetAttributeValue_DomainValue

```
public void test_Positive_cbe_SetAttributeValue_DomainValue()  
    throws Exception
```

this is a generic method to test the setattribute value function based on the type of the field

test_cbe_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName

```
public void  
test_cbe_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()  
    throws Exception
```

tests that setAttributeValue(a) => a in getPopulatedAttributes()

test_cbe_SetAttributeValue_IsPopulated

```
public void test_cbe_SetAttributeValue_IsPopulated()  
    throws Exception
```

tests that setAttributeValue(a) => isPopulated(a)

test_Negative_cbe_SetAttributeValue_IllegalObject

```
public void test_Negative_cbe_SetAttributeValue_IllegalObject()  
    throws Exception
```

tests failure of setAttributeValue(a) for an illegal object()

(continued from last page)

test_Positive_cbe_testCloneEqualsImplementation

```
public void test_Positive_cbe_testCloneEqualsImplementation( )  
    throws Exception
```

Test clone and equals methods CBE implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true
a.equals(b) == true b.equals(a) == true

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface `junit.framework.Test`

`countTestCases`, `run`

Constructors

TestEvents

```
public TestEvents(String name)
```

Creates new testEvents

Methods

setUp

```
protected void setUp()  
    throws Exception
```

tearDown

```
protected void tearDown()  
    throws Exception
```

testAllEvents

```
public void testAllEvents()  
    throws Exception
```

Infinit Loop

For all the messages received via the topic:

- verify they are of the type `javax.oss.Event`
- contains time and applicationDN properties
- is cloneable

Throws:

Exception - DOCUMENT ME!

All Implemented Interfaces:

junit.framework.Test

```
public static class TestEvents.EndlessEventTest
    extends junit.extensions.TestDecorator
```

Fields inherited from class <code>junit.extensions.TestDecorator</code>
<code>fTest</code>

public	<u>TestEvents.EndlessEventTest()</u>
--------	--

int	<code>countTestCases()</code>
void	<code>run(junit.framework.TestResult result)</code>
static junit.framework.TestS uite	<code>suite()</code>
String	<code>toString()</code>

Methods inherited from class <code>junit.extensions.TestDecorator</code>
<code>basicRun</code> , <code>countTestCases</code> , <code>getTest</code> , <code>run</code> , <code>toString</code>

Methods inherited from class junit.framework.Assert
assertEquals, assertFalse, assertFalse, assertNotNull, assertNotNull, assertNotSame, assertNotSame, assertNull, assertNull, assertEquals, assertEquals, assertTrue, assertTrue, fail, fail

Methods inherited from class `java.lang.Object`


```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

Methods inherited from interface junit.framework.Test

```
countTestCases, run
```

Constructors

TestEvents.EndlessEventTest

```
public TestEvents.EndlessEventTest()
```

Methods

suite

```
public static junit.framework.TestSuite suite()
```

countTestCases

```
public int countTestCases()
```

run

```
public void run(junit.framework.TestResult result)
```

toString

```
public String toString()
```

ossj.common.tck.tests Class TestEventValue

```

java.lang.Object
  |
  +- junit.framework.Assert
      |
      +- junit.framework.TestCase
          |
          +- ossj.common.tck.JVTTestBase
              |
              +- ossj.common.tck.tests.TestMev
                  |
                  +- ossj.common.tck.tests.TestEventValue
  
```

All Implemented Interfaces:

junit.framework.Test

public class **TestEventValue**
extends [TestMev](#)

Author:

Vincent Perrot

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Constructor Summary

public	TestEventValue (String name) Creates new testJVTActivationSession
--------	--

Method Summary

void	doGeneric (String testName) genericTest
void	setUp ()
static junit.framework.TestSuite	suite () Generate the test cases
void	tearDown ()
void	test_event_GetAttributeNames_NamesStartWithLowerCase () tests that attributes - names have length > 0 - names start with lowercase letter - get/set methods exist
void	test_event_GetOptionalAttributeNames_InAllAttributes () tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter

void	<u>test_event_GetSettableAttributeNames_SubsetOfGetAttributeNames()</u> tests that getSettableAttributesNames is a subset of getAttributeNames
void	<u>test_event_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeNames()</u> tests that setAttributeValue(a) => a in getPopulatedAttributes()
void	<u>test_event_SetAttributeValue_IsPopulated()</u> tests that setAttributeValue(a) => isPopulated(a)
void	<u>test_event_UnpopulateAllAttributes_GetAttributeValueFailsForAll()</u> tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames
void	<u>test_event_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()</u> tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty
void	<u>test_event_UnpopulateAllAttributes_IsPopulatedFalseForAll()</u> tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()
void	<u>test_Negative_event_GetAttributeValue_AttributeNameIllegalValue()</u> tests failure of getAttributeValue(a) for a not in getAttributeNames();
void	<u>test_Negative_event_GetAttributeValue_AttributeNameNullValue()</u> tests failure of getAttributeValue(null)
void	<u>test_Negative_event_IsPopulated_AttributeNameIllegalValue()</u> tests failure of isPopulated(a) for String a not in getAttributeNames()
void	<u>test_Negative_event_IsPopulated_AttributeNameNullValue()</u> tests failure of isPopulated(null)
void	<u>test_Negative_event_SetAttributeValue_AttributeNameIllegalValue()</u> tests failure of setAttributeValue(a) for a not in getAttributeNames()
void	<u>test_Negative_event_SetAttributeValue_AttributeNameNullValue()</u> tests failure of setAttributeValue(a) for null
void	<u>test_Negative_event_SetAttributeValue_IllegalObject()</u> tests failure of setAttributeValue(a) for an illegal object()
void	<u>test_Negative_event_SetAttributeValue_PrimitiveNullValue()</u> tests Illegal null primitive attribute value
void	<u>test_Negative_event_UnpopulateAttribute_AttributeNameIllegalValue()</u> tests failure of unpopulateAttribute(a) for a not in getAttributeNames
void	<u>test_Negative_event_UnpopulateAttribute_AttributeNameNullValue()</u> tests failure of unpopulateAttribute(null)
void	<u>test_Positive_event_IsPopulated_AttributeNameAllDomainValues()</u> tests isPopulated for all attributes in getAttributeNames()
void	<u>test_Positive_event_SetAttributeValue_DomainValue()</u> this is a generic method to test the setattribute value function based on the type of the field
void	<u>test_Positive_event_SetAttributeValue_PrimitiveMINMAXValues()</u> tests setAttributeValue(min/max) for primitive attributes that are not enums

void	test_Positive_event_testCloneEqualsImplementation() Test clone and equals methods CBE implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true a.equals(b) == true b.equals(a) == true
void	test_Positive_event_UnpopulateAttribute_AttributeNameAllDomainValues() tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()
void	test_Positive_getEventImplementation() Check presence of CBE interfaces implementations Search only in directoryies and jar file name starting with "oss" Expected Result: a hashtable completed with all CBE interfaces from test_Positive_getCbeInterfaces as key and ArrayList that contains all implementation Classes as value

Methods inherited from class [ossj.common.tck.tests.TestMev](#)

[setUp](#), [test_ClassNameAndTypeAndValueKeyPair](#), [test_GetAttributeNames_ContainsKEY](#),
[test_GetAttributeNames_NamesStartWithLowerCase](#),
[test_GetOptionalAttributeNames_InAllAttributes](#),
[test_GetSettableAttributeNames_NotContainsKEY](#),
[test_GetSettableAttributeNames_SubsetOfGetAttributeNames](#),
[test_Negative_GetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_GetAttributeValue_AttributeNameNullValue](#),
[test_Negative_IsPopulated_AttributeNameIllegalValue](#),
[test_Negative_IsPopulated_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_SetAttributeValue_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_IllegalObject](#),
[test_Negative_SetAttributeValue_PrimitiveNullValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameIllegalValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameNullValue](#),
[test_Positive_IsPopulated_AttributeNameAllDomainValues](#),
[test_Positive_ManagedEntityKey_IllegalKey](#),
[test_Positive_SetAttributeValue_DomainValue](#),
[test_Positive_SetAttributeValue_PrimitiveMINMAXValues](#),
[test_Positive_SetAttributeValueKEY_IllegalKey](#),
[test_Positive_SetManagedEntityKey_DomainValue](#),
[test_Positive_SetManagedEntityKey_IllegalKey](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues](#), [test_Query_Type](#),
[test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName](#),
[test_SetAttributeValue_IsPopulated](#),
[test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY](#),
[test_SetManagedEntityKey_IsPopulatedKEY](#),
[test_UnpopulateAllAttributes_GetAttributeValueFailsForAll](#),
[test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty](#),
[test_UnpopulateAllAttributes_IsPopulatedFalseForAll](#), [test_upv_Type](#)

Methods inherited from class [ossj.common.tck.JVTTestBase](#)

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#),
[init](#), [log](#), [log](#), [run](#), [run](#)

Methods inherited from class [junit.framework.TestCase](#)

[countTestCases](#), [createResult](#), [getName](#), [run](#), [run](#), [runBare](#), [runTest](#), [setName](#), [setUp](#),
[tearDown](#), [toString](#)

doGeneric

```
protected void doGeneric(String testName)
    throws Exception
```

genericTest

test_event_GetAttributeNames_NamesStartWithLowerCase

```
public void test_event_GetAttributeNames_NamesStartWithLowerCase()
    throws Exception
```

tests that attributes - names have length > 0 - names start with lowercase letter - get/set methods exist

test_event_GetOptionalAttributeNames_InAllAttributes

```
public void test_event_GetOptionalAttributeNames_InAllAttributes()
    throws Exception
```

tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter

test_Positive_event_IsPopulated_AttributeNameAllDomainValues

```
public void test_Positive_event_IsPopulated_AttributeNameAllDomainValues()
    throws Exception
```

tests isPopulated for all attributes in getAttributeNames()

test_Negative_event_IsPopulated_AttributeNameNullValue

```
public void test_Negative_event_IsPopulated_AttributeNameNullValue()
    throws Exception
```

tests failure of isPopulated(null)

test_Negative_event_IsPopulated_AttributeNameIllegalValue

```
public void test_Negative_event_IsPopulated_AttributeNameIllegalValue()
    throws Exception
```

tests failure of isPopulated(a) for String a not in getAttributeNames()

test_event_GetSettableAttributeNames_SubsetOfGetAttributeNames

```
public void test_event_GetSettableAttributeNames_SubsetOfGetAttributeNames()
    throws Exception
```

tests that getSettableAttributesNames is a subset of getAttributeNames

test_Positive_event_UnpopulateAttribute_AttributeNameAllDomainValues

```
public void test_Positive_event_UnpopulateAttribute_AttributeNameAllDomainValues()
    throws Exception
```

tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()

(continued from last page)

test_Negative_event_UnpopulateAttribute_AttributeNameNullValue

```
public void test_Negative_event_UnpopulateAttribute_AttributeNameNullValue()  
    throws Exception
```

tests failure of unpopulateAttribute(null)

test_Negative_event_UnpopulateAttribute_AttributeNameIllegalValue

```
public void test_Negative_event_UnpopulateAttribute_AttributeNameIllegalValue()  
    throws Exception
```

tests failure of unpopulateAttribute(a) for a not in getAttributeNames

test_event_UnpopulateAllAttributes_IsPopulatedFalseForAll

```
public void test_event_UnpopulateAllAttributes_IsPopulatedFalseForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

test_event_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty

```
public void test_event_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()  
    throws Exception
```

tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty

test_event_UnpopulateAllAttributes_GetAttributeValueFailsForAll

```
public void test_event_UnpopulateAllAttributes_GetAttributeValueFailsForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames

test_Negative_event_SetAttributeValue_AttributeNameNullValue

```
public void test_Negative_event_SetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

tests failure of setAttributeValue(a) for null

test_Negative_event_SetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_event_SetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

tests failure of setAttributeValue(a) for a not in getAttributeNames()

test_Negative_event_GetAttributeValue_AttributeNameNullValue

```
public void test_Negative_event_GetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

tests failure of getAttributeValue(null)

(continued from last page)

test_Negative_event_GetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_event_GetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

tests failure of getAttributeValue(a) for a not in getAttributeValues();

test_Positive_event_SetAttributeValue_PrimitiveMINMAXValues

```
public void test_Positive_event_SetAttributeValue_PrimitiveMINMAXValues()  
    throws Exception
```

tests setAttributeValue(min/max) for primitive attributes that are not enums

test_Negative_event_SetAttributeValue_PrimitiveNullValue

```
public void test_Negative_event_SetAttributeValue_PrimitiveNullValue()  
    throws Exception
```

tests Illegal null primitive attribute value

test_Positive_event_SetAttributeValue_DomainValue

```
public void test_Positive_event_SetAttributeValue_DomainValue()  
    throws Exception
```

this is a generic method to test the setattribute value function based on the type of the field

test_event_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName

```
public void  
test_event_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()  
    throws Exception
```

tests that setAttributeValue(a) => a in getPopulatedAttributes()

test_event_SetAttributeValue_IsPopulated

```
public void test_event_SetAttributeValue_IsPopulated()  
    throws Exception
```

tests that setAttributeValue(a) => isPopulated(a)

test_Negative_event_SetAttributeValue_IllegalObject

```
public void test_Negative_event_SetAttributeValue_IllegalObject()  
    throws Exception
```

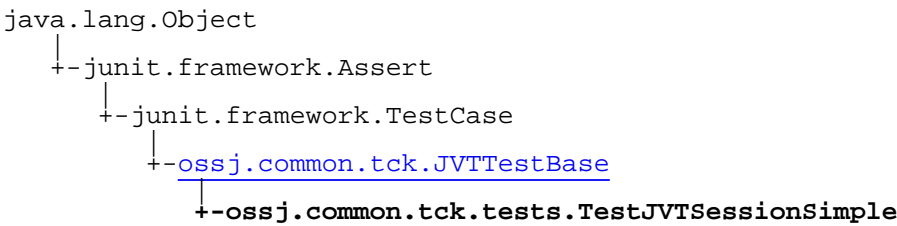
tests failure of setAttributeValue(a) for an illegal object()

test_Positive_event_testCloneEqualsImplementation

```
public void test_Positive_event_testCloneEqualsImplementation()  
    throws Exception
```

Test clone and equals methods CBE implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true
a.equals(b) == true b.equals(a) == true

ossj.common.tck.tests
Class TestJVTSessionSimple



All Implemented Interfaces:
junit.framework.Test

public class TestJVTSessionSimple
extends [JVTTestBase](#)

tests everything not related to ManagedEntity
Author:
Vincent Perrot Sun Microsystems
Version:
1.2

Fields inherited from class ossj.common.tck.JVTTestBase	
debug , done , EVENT_TIMEOUT , howmany_key , howmany_value , JMS_FACTORY , JNDI_FACT_CLASS_NAME , JNDI_URL , jvtSessionList , LOG_FILE , namingContext , QUOTECHAR , result , return_failed_only , runonce , TESTFILES_HOME , TOPIC_NAME , util	

Constructor Summary	
public	TestJVTSessionSimple (String name) Creates new TestJVTSessionSimple

Method Summary	
void	setUp ()
static junit.framework.TestS uite	suite () Creates the suite of tests
void	tearDown ()
void	test_GetManagedEntityTypes_ContainsACTIVATE () getManagedEntityTypes, instantiated from TestBasics verify that the JVT support at least one managed Entity type using getManagedEntityTypes
void	test_GetQueryTypes_IsNotNull () verify that JVT support at least one query type using getNamedQueryTypes. getNamedQueryTypes instantiated from TestBasics
void	test_GetSupportedOperations_IsSubsetOfJVTSessionOption () GetSupportedOperations instantiated from TestBasics. just call getSupportedOptionalOperations

void	<u>test_GetUpvTypes_IsNotNull()</u> verify that JVT support at least one query type using getUpdateProcedureTypes. getUpdateProcedureTypes instantiated from TestBasics
void	<u>test_JVT_JNDI_NAME()</u> Test if the given JNDI name follows: System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Comp/name or System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Resource/name
void	<u>test_Negative_NewManagedEntityValue_IllegalValue()</u> Test managed entity value using unsupported type Success if OssIllegalArgumentException is thrown when an invalid (not returned by getTypes) type is given
void	<u>test_Negative_NewManagedEntityValue_NullValue()</u> test managed entity value factory using null type Success is makeManagedEntityValue(null) throws OssIllegalArgumentException
void	<u>test_Negative_NewQueryValue_IllegalValue()</u> try to create new query value using the factory and invalid query type Success if OssIllegalArgumentException is thrown
void	<u>test_Negative_NewQueryValue_NullValue()</u> try to create new query value using the factory and null query type Success if OssIllegalArgumentException is thrown
void	<u>test_Negative_NewUpvValue_IllegalValue()</u> Test Upv value using unsupported type Success if OssIllegalArgumentException is thrown when an invalid (not returned by getTypes) type is given
void	<u>test_Negative_NewUpvValue_NullValue()</u> test Upv value factory using null type Success is makeUpdateProcedureValue(null) throws OssIllegalArgumentException
void	<u>test_Negative_query_NullQuery()</u> Query managed entities using null query type Success if query(null) throws OssIllegalArgumentException
void	<u>test_Negative_update_NullUpv()</u> Update procedure using null type Success if update(null) throws OssIllegalArgumentException
void	<u>test_Positive_callMethodKeyCreateWithValue()</u> Try to find and test methods with the format ManagedEntityKey methodName(ManagedEntityValue) Success if at least one method with the profile ManagedEntityKey (ManagedEntityValue) found (the method is called anyway (exception or valid retrn value are correct)
void	<u>test_Positive_callMethodKeysCreateWithValues()</u> Try to find and test methods with the format ManagedEntityKeyResults[] methodName(ManagedEntityValue[]) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults Success if at least one method with the profile MManagedEntityKeyResults[] methodName(ManagedEntityValue[]) found (the method is called anyway (exception or valid retrn value are correct)

void	test_Positive_callMethodValuesMethWithKeys() Try to find and test methods with the format ManagedEntityValue[] methodname(ManagedEntityKeys[]) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults Success if at least one method with the profile ManagedEntityValue[] methodname(ManagedEntityKey[]) found (the method is called anyway (exception or valid retrn value are correct)
void	test_Positive_callMethodValueWithKey() Try to find and test methods with the format ManagedEntityValue methodname(ManagedEntitykey) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults Success if at least one method with the profile ManagedEntityValue methodname(ManagedEntitykey) found (the method is called anyway (exception or valid retrn value are correct)
void	test_Positive_callMethodVoidWithKey() Try to find and test methods with the format void methodname(ManagedEntitykey) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults Success if at least one method with the profile void methodname(ManagedEntitykey) found (the method is called anyway (exception or valid retrn value are correct)
void	test_Positive_callMethodVoidWithValue() Try to find and test methods with the format void methodname(ManagedEntityValue) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults Success if at least one method with the profile void methodname(ManagedEntityValue) found (the method is called anyway (exception or valid retrn value are correct)
void	test_Positive_getEventTypes() Verify that the JVT support at least one event type. For each Event type, verify also the corresponding EventPropertyDescriptor using the factory.
void	test_Positive_NewManagedEntityValue_AllDomainValues() newManagedEntityValue Basically test for makeManagedEntityValue Success if a valid value instance is returned: no null and with Key field populated
void	test_Positive_NewQueryValue_AllDomainValues() Verify the query value factory to create new query values Success if makeNamedQueryValue return none null instance
void	test_Positive_NewUpvValue_AllDomainValues() new Upv Value Basically test for makeUpdateProcedureValue Success if a valid value instance is returned: no null

Methods inherited from class [ossj.common.tck.JVTTestBase](#)

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#),
[init](#), [log](#), [log](#), [run](#), [run](#)

Methods inherited from class [junit.framework.TestCase](#)

[countTestCases](#), [createResult](#), [getName](#), [run](#), [run](#), [runBare](#), [runTest](#), [setName](#), [setUp](#),
[tearDown](#), [toString](#)

Methods inherited from class [junit.framework.Assert](#)

[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertFalse](#), [assertFalse](#), [assertNotNull](#), [assertNotNull](#),
[assertNotSame](#), [assertNotSame](#), [assertNull](#), [assertNull](#), [assertSame](#), [assertSame](#),
[assertTrue](#), [assertTrue](#), [fail](#), [fail](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface `junit.framework.Test`

`countTestCases`, `run`

Constructors

TestJVTSessionSimple

```
public TestJVTSessionSimple(String name)
```

Creates new TestJVTSessionSimple

Methods

setUp

```
protected void setUp()  
    throws Exception
```

tearDown

```
protected void tearDown()
```

suite

```
public static junit.framework.TestSuite suite()  
    throws Exception
```

Creates the suite of tests

Returns:

the object encapsulating the suite of test

Throws:

Exception

test_JVT_JNDI_NAME

```
public void test_JVT_JNDI_NAME()  
    throws Exception
```

Test if the given JNDI name follows: `System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Comp/name` or `System/System1/ApplicationType/Common/Application/1-1;1-1;ReferenceImplementation/Resource/name`

Throws:

Exception

test_GetManagedEntityTypes_ContainsACTIVATE

```
public void test_GetManagedEntityTypes_ContainsACTIVATE()  
    throws Exception
```

getManagedEntityTypes, instantiated from TestBasics
verify that the JVT support at least one managed Entity type using getManagedEntityTypes

Success if the the return array is not null and not empty

test_GetQueryTypes_IsNotNull

```
public void test_GetQueryTypes_IsNotNull()  
    throws Exception
```

verify that JVT support at least one query type using getNamedQueryTypes.
getNamedQueryTypes instantiated from TestBasics

Success if the the return array is not null and not empty

test_GetUpvTypes_IsNotNull

```
public void test_GetUpvTypes_IsNotNull()  
    throws Exception
```

verify that JVT support at least one query type using getUpdateProcedureTypes.
getUpdateProcedureTypes instantiated from TestBasics

Success if the the return array is not null

test_GetSupportedOperations_IsSubsetOfJVTSessionOption

```
public void test_GetSupportedOperations_IsSubsetOfJVTSessionOption()  
    throws Exception
```

GetSupportedOperations instantiated from TestBasics.
just call getSupportedOptionalOperations

Success if no exceptioun is thrown.

test_Positive_NewManagedEntityValue_AllDomainValues

```
public void test_Positive_NewManagedEntityValue_AllDomainValues()  
    throws Exception
```

newManagedEntityValue Basically test for makeManagedEntityValue Success if a valid value instance is returned:
no null and with Key field populated

test_Negative_NewManagedEntityValue_NullValue

```
public void test_Negative_NewManagedEntityValue_NullValue()  
    throws Exception
```

test managed entity value factory using null type
Sucess is makeManagedEntityValue(null) throws OssIllegalArgumentException

Throws:

Exception - DOCUMENT ME!

(continued from last page)

test_Negative_NewManagedEntityValue_IllegalValue

```
public void test_Negative_NewManagedEntityValue_IllegalValue()  
    throws Exception
```

Test managed entity value using unsupported type

Success if OssIllegalArgumentException is thrown when an invalid (not returned by getTypes) type is given

Throws:

Exception - DOCUMENT ME!

test_Positive_NewUpvValue_AllDomainValues

```
public void test_Positive_NewUpvValue_AllDomainValues()  
    throws Exception
```

new Upv Value Basically test for makeUpdateProcedureValue Success if a valid value instance is returned:
no null

test_Negative_NewUpvValue_NullValue

```
public void test_Negative_NewUpvValue_NullValue()  
    throws Exception
```

test Upv value factory using null type

Success is makeUpdateProcedureValue(null) throws OssIllegalArgumentException

Throws:

Exception - DOCUMENT ME!

test_Negative_NewUpvValue_IllegalValue

```
public void test_Negative_NewUpvValue_IllegalValue()  
    throws Exception
```

Test Upv value using unsupported type

Success if OssIllegalArgumentException is thrown when an invalid (not returned by getTypes) type is given

Throws:

Exception - DOCUMENT ME!

test_Positive_NewQueryValue_AllDomainValues

```
public void test_Positive_NewQueryValue_AllDomainValues()  
    throws Exception
```

Verify the query value factory to create new query values

Success if makeNamedQueryValue return none null instance

test_Negative_NewQueryValue_NullValue

```
public void test_Negative_NewQueryValue_NullValue()  
    throws Exception
```

try to create new query value using the factory and null query type

Success if OssIllegalArgument is thrown

Throws:

Exception - DOCUMENT ME!

test_Negative_NewQueryValue_IllegalValue

```
public void test_Negative_NewQueryValue_IllegalValue()  
    throws Exception
```

try to create new query value using the factory and invalid query type
Success if OssIllegalArgument is thrown

Throws:

Exception - DOCUMENT ME!

test_Negative_query_NullQuery

```
public void test_Negative_query_NullQuery()  
    throws Exception
```

Query managed entities using null query type
Success if query(null) throws OssIllegalArgumentException

Throws:

Exception - DOCUMENT ME!

[TCKException](#) - DOCUMENT ME!

test_Negative_update_NullUpv

```
public void test_Negative_update_NullUpv()  
    throws Exception
```

Update procedure using null type
Success if update(null) throws OssIllegalArgumentException

Throws:

Exception - DOCUMENT ME!

[TCKException](#) - DOCUMENT ME!

test_Positive_callMethodKeyCreateWithValue

```
public void test_Positive_callMethodKeyCreateWithValue()  
    throws Exception
```

Try to find and test methods with the format ManagedEntityKey methodname(ManagedEntityValue)
Success if at least one method with the profile ManagedEntityKey (ManagedEntityValue) found (the method is called anyway (exception or valid return value are correct))

Throws:

Exception - DOCUMENT ME!

test_Positive_callMethodKeysCreateWithValues

```
public void test_Positive_callMethodKeysCreateWithValues()  
    throws Exception
```

Try to find and test methods with the format ManagedEntityKeyResults[] methodname(ManagedEntityValue[]) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults
Success if at least one method with the profile MManagedEntityKeyResults[] methodname(ManagedEntityValue[]) found (the method is called anyway (exception or valid return value are correct))

Throws:

Exception - DOCUMENT ME!

test_Positive_callMethodValuesMethWithKeys

```
public void test_Positive_callMethodValuesMethWithKeys()  
    throws Exception
```

Try to find and test methods with the format ManagedEntityValue[] methodname(ManagedEntityKeys[]) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults
Success if at least one method with the profile ManagedEntityValue[] methodname(ManagedEntityKey[]) found (the method is called anyway (exception or valid retrn value are correct)

Throws:

Exception - DOCUMENT ME!

test_Positive_callMethodVoidWithValue

```
public void test_Positive_callMethodVoidWithValue()  
    throws Exception
```

Try to find and test methods with the format void methodname(ManagedEntityValue) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults
Success if at least one method with the profile void methodname(ManagedEntityValue) found (the method is called anyway (exception or valid retrn value are correct)

Throws:

Exception - DOCUMENT ME!

test_Positive_callMethodValueWithKey

```
public void test_Positive_callMethodValueWithKey()  
    throws Exception
```

Try to find and test methods with the format ManagedEntityValue methodname(ManagedEntitykey) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults
Success if at least one method with the profile ManagedEntityValue methodname(ManagedEntitykey) found (the method is called anyway (exception or valid retrn value are correct)

Throws:

Exception - DOCUMENT ME!

test_Positive_callMethodVoidWithKey

```
public void test_Positive_callMethodVoidWithKey()  
    throws Exception
```

Try to find and test methods with the format void methodname(ManagedEntitykey) If call to the method is successful the isSuccess or getException method are called on all returned ManagedEntityKeyresults
Success if at least one method with the profile void methodname(ManagedEntitykey) found (the method is called anyway (exception or valid retrn value are correct)

Throws:

Exception - DOCUMENT ME!

test_Positive_getEventTypes

```
public void test_Positive_getEventTypes()  
    throws Exception
```


(continued from last page)

Verify that the JVT support at least one event type.

For each Event type, verify also the corresponding EventPropertyDescriptor using the factory.

Success

if getEventTypes return no null and at least 1 event type and if (getEventDescriptor) for each type return a valid event propertyDescriptor (correct event type value) and if makeEvent(type) do not throw any exception and returnan instance of the desired event type

Throws:

Exception - DOCUMENT ME!

ossj.common.tck.tests

Class TestManagedEntityKey

```

java.lang.Object
├─ junit.framework.Assert
│   └─ junit.framework.TestCase
│       └─ ossj.common.tck.JVTTestBase
│           └─ ossj.common.tck.tests.TestManagedEntityKey

```

All Implemented Interfaces:

junit.framework.Test

public class **TestManagedEntityKey**
 extends [JVTTestBase](#)

This test checks basic functionality of the TestManagedEntityKey

Author:

Vincent Perrot, Sun Microsystems Inc.

Version:

1.1

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Constructor Summary

public	TestManagedEntityKey (String name) Creates new testJVTActivationSession
--------	--

Method Summary

void	setUp ()
static junit.framework.TestS uite	suite () build the test Suite.
void	tearDown ()
void	test_GetApplicationContext () Verify the ApplicationContext attribute of the key implementation.
void	test_GetApplicationDN () Verify the ApplicationDN attribute of the key implementation.
void	test_GetPrimaryKey () Verify the primaryKey attribute of the key implementation.

void	test_GetType() Verify the type attribute of the key implementation.
void	test_KeyCloneEquals() Verify the Aimplementation of the clone and equals meethods. Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true a.equals(b) == true b.equals(a) == true
void	test_MakePrimaryKey() Verify the key factory of the key implementation.
void	test_SetMakeApplicationContext() Verify the ApplicationContext attribute of the key implementation.
void	test_SetPrimaryKeyIsNull() Verify the primaryKey attribute of the key implementation.
void	test_SetTypeIsNull() Verify the type attribute of the key implementation.

Methods inherited from class [ossj.common.tck.JVTTestBase](#)

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#), [init](#), [log](#), [log](#), [run](#), [run](#)

Methods inherited from class [junit.framework.TestCase](#)

[countTestCases](#), [createResult](#), [getName](#), [run](#), [run](#), [runBare](#), [runTest](#), [setName](#), [setUp](#), [tearDown](#), [toString](#)

Methods inherited from class [junit.framework.Assert](#)

[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertFalse](#), [assertFalse](#), [assertNotNull](#), [assertNotNull](#), [assertNotSame](#), [assertNotSame](#), [assertNull](#), [assertNull](#), [assertSame](#), [assertSame](#), [assertTrue](#), [assertTrue](#), [fail](#), [fail](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [junit.framework.Test](#)

[countTestCases](#), [run](#)

Constructors

TestManagedEntityKey

```
public TestManagedEntityKey(String name)
```

Creates new testJVTActivationSession

Methods

(continued from last page)

suite

```
public static junit.framework.TestSuite suite()
```

build the test Suite.

Returns:

DOCUMENT ME!

setUp

```
protected void setUp()  
    throws Exception
```

tearDown

```
protected void tearDown()
```

test_GetApplicationContext

```
public void test_GetApplicationContext()  
    throws Exception
```

Verify the ApplicationContext attribute of the key implementation. It shall be a ApplicationContext class or derived from it.

Success if getApplicationContext return a no null and the the object of ApplicationContext Type

Throws:

Exception

test_SetMakeApplicationContext

```
public void test_SetMakeApplicationContext()  
    throws Exception
```

Verify the ApplicationContext attribute of the key implementation.

Success if setApplicationContext(makeApplicationContext) doesn't throw any Exception ad getApplicationConetxt == value used to set ApplicationContext

Throws:

Exception

test_GetApplicationDN

```
public void test_GetApplicationDN()  
    throws Exception
```

Verify the ApplicationDN attribute of the key implementation. It shall not be null.

Throws:

Exception

(continued from last page)

test_GetPrimaryKey

```
public void test_GetPrimaryKey()  
    throws Exception
```

Verify the primaryKey attribute of the key implementation. It shall not be null.

Throws:

Exception

test_GetType

```
public void test_GetType()  
    throws Exception
```

Verify the type attribute of the key implementation. It shall not be null.

Throws:

Exception

test_MakePrimaryKey

```
public void test_MakePrimaryKey()  
    throws Exception
```

Verify the key factory of the key implementation. The generated key shall not be null. And the call to set(make) shall not throw any exception

Throws:

Exception

test_SetPrimaryKeyIsNull

```
public void test_SetPrimaryKeyIsNull()  
    throws Exception
```

Verify the primaryKey attribute of the key implementation. It shall not support the attribution of a null value.

Throws:

Exception

test_SetTypeIsNull

```
public void test_SetTypeIsNull()  
    throws Exception
```

Verify the type attribute of the key implementation. It shall not support the attribution of a null value.

Throws:

Exception

test_KeyCloneEquals

```
public void test_KeyCloneEquals()  
    throws Exception
```

(continued from last page)

Verify the Aimplementation of the clone and equals meethods.

Expected Result:

a = b.clone()

a.equals(a) == true

b.equals(b) == true

a.equals(b) == true

b.equals(a) == true

Throws:

Exception

ossj.common.tck.tests

Class TestManagedEntityValueSingle

```

java.lang.Object
├─ junit.framework.Assert
│   └─ junit.framework.TestCase
│       └─ ossj.common.tck.JVTTestBase
│           └─ ossj.common.tck.tests.TestMev
│               └─ ossj.common.tck.tests.TestManagedEntityValueSingle

```

All Implemented Interfaces:

junit.framework.Test

public class **TestManagedEntityValueSingle**
 extends [TestMev](#)

Test case for ManagedEntityValues

Author:

Version:

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Method Summary

void	doGeneric (String testName) genericTest
void	setUp () sets up the TestCase
static junit.framework.TestSuite	suite ()
void	test_ClassNameAndTypeAndValueKeyPair () Test Value and Value type and key pattern test if VALUE_TYPE field is declared test if VALUE_TYPE field is public final static String test if the declared type exist (call forName) test if the class is of the declared type (decalreType.isAssignableFrom) test if the VALUE_TYPE ends with "Value" test if the VALUE_TYPE corresponding interface exists
void	test_GetAttributeNames_ContainsKEY () setManagedEntityKey
void	test_GetAttributeNames_NamesStartWithLowerCase () tests that attributes names have length > 0 and start with lowercase letter

void	<u>test_GetOptionalAttributeNames_InAllAttributes()</u> tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter
void	<u>test_GetSettableAttributeNames_NotContainsKEY()</u>
void	<u>test_GetSettableAttributeNames_SubsetOfGetAttributeNames()</u> tests that getSettableAttributesNames is a subset of getAttributeNames
void	<u>test_Negative_GetAttributeValue_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of getAttributeValue(a) for a not in getAttributeValues();
void	<u>test_Negative_GetAttributeValue_AttributeNameNullValue()</u> DOMAINTEST tests failure of getAttributeValue(null)
void	<u>test_Negative_IsPopulated_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of isPopulated(a) for String a not in getAttributeNames()
void	<u>test_Negative_IsPopulated_AttributeNameNullValue()</u> DOMAINTEST tests failure of isPopulated(null)
void	<u>test_Negative_SetAttributeValue_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of setAttributeValue(a) for a not in getAttributeNames()
void	<u>test_Negative_SetAttributeValue_AttributeNameNullValue()</u> DOMAINTEST tests failure of setAttributeValue(a) for null
void	<u>test_Negative_SetAttributeValue_IllegalObject()</u> DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()
void	<u>test_Negative_SetAttributeValue_PrimitiveNullValue()</u> DOCUMENT ME!
void	<u>test_Negative_UnpopulateAttribute_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of unpopulatedAttribute(a) for a not in getAttributeNames
void	<u>test_Negative_UnpopulateAttribute_AttributeNameNullValue()</u> DOMAINTEST tests failure of unpopulatedAttribute(null)
void	<u>test_Positive_IsPopulated_AttributeNameAllDomainValues()</u> DOMAINTEST tests isPopulated for all attributes in getAttributeNames()
void	<u>test_Positive_ManagedEntityKey_IllegalKey()</u>
void	<u>test_Positive_SetAttributeValue_DomainValue()</u>
void	<u>test_Positive_SetAttributeValue_PrimitiveMINMAXValues()</u> tests setAttributeValue(min/max) for primitive attributes that are not enums
void	<u>test_Positive_SetAttributeValueKEY_IllegalKey()</u> tests a illegal key
void	<u>test_Positive_SetManagedEntityKey_DomainValue()</u> tests a domain value
void	<u>test_Positive_SetManagedEntityKey_IllegalKey()</u> tests a illegal key

void	test_Positive_testCloneEqualsImplementation() Test clone and equals method implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true a.equals(b) == true b.equals(a) == true
void	test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues() DOMAINTTEST tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()
void	test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName() tests that setAttributeValue(a) => a in getPopulatedAttributes()
void	test_SetAttributeValue_IsPopulated() tests that setAttributeValue(a) => isPopulated(a)
void	test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY() tests that succesful setManagedEntityKey(val) implies KEY in getPopulatedAttributeNames()
void	test_SetManagedEntityKey_IsPopulatedKEY() tests that succesful setManagedEntityKey(val) implies isPopulated(KEY)
void	test_UnpopulateAllAttributes_GetAttributeValueFailsForAll() tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames
void	test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty() tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty
void	test_UnpopulateAllAttributes_IsPopulatedFalseForAll() tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

Methods inherited from class [ossj.common.tck.tests.TestMev](#)

[setUp](#), [test_ClassNameAndTypeAndValueKeyPair](#), [test_GetAttributeNames_ContainsKEY](#),
[test_GetAttributeNames_NamesStartWithLowerCase](#),
[test_GetOptionalAttributeNames_InAllAttributes](#),
[test_GetSettableAttributeNames_NotContainsKEY](#),
[test_GetSettableAttributeNames_SubsetOfGetAttributeNames](#),
[test_Negative_GetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_GetAttributeValue_AttributeNameNullValue](#),
[test_Negative_IsPopulated_AttributeNameIllegalValue](#),
[test_Negative_IsPopulated_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_SetAttributeValue_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_IllegalObject](#),
[test_Negative_SetAttributeValue_PrimitiveNullValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameIllegalValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameNullValue](#),
[test_Positive_IsPopulated_AttributeNameAllDomainValues](#),
[test_Positive_ManagedEntityKey_IllegalKey](#),
[test_Positive_SetAttributeValue_DomainValue](#),
[test_Positive_SetAttributeValue_PrimitiveMINMAXValues](#),
[test_Positive_SetAttributeValueKEY_IllegalKey](#),
[test_Positive_SetManagedEntityKey_DomainValue](#),
[test_Positive_SetManagedEntityKey_IllegalKey](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues](#), [test_Query_Type](#),
[test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName](#),
[test_SetAttributeValue_IsPopulated](#),
[test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY](#),
[test_SetManagedEntityKey_IsPopulatedKEY](#),
[test_UnpopulateAllAttributes_GetAttributeValueFailsForAll](#),
[test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty](#),
[test_UnpopulateAllAttributes_IsPopulatedFalseForAll](#), [test_upv_Type](#)

Methods inherited from class [ossj.common.tck.JVTTestBase](#)

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#),
[init](#), [log](#), [log](#), [run](#), [run](#)

Methods inherited from class [junit.framework.TestCase](#)

[countTestCases](#), [createResult](#), [getName](#), [run](#), [run](#), [runBare](#), [runTest](#), [setName](#), [setUp](#),
[tearDown](#), [toString](#)

Methods inherited from class [junit.framework.Assert](#)

[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertFalse](#), [assertFalse](#), [assertNotNull](#), [assertNotNull](#),
[assertNotSame](#), [assertNotSame](#), [assertNull](#), [assertNull](#), [assertSame](#), [assertSame](#),
[assertTrue](#), [assertTrue](#), [fail](#), [fail](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#),
[wait](#)

Methods inherited from interface junit.framework.Test

countTestCases, run

Methods

setUp

```
protected void setUp()  
    throws Exception
```

sets up the TestCase

suite

```
public static junit.framework.TestSuite suite()  
    throws Exception
```

Returns:

All suites of this class

doGeneric

```
protected void doGeneric(String testName)  
    throws Exception
```

genericTest

test_ClassNameAndTypeAndValueKeyPair

```
public void test_ClassNameAndTypeAndValueKeyPair()  
    throws Exception
```

Test Value and Value type and key pattern test if VALUE_TYPE field is declared test if VALUE_TYPE field is public
final static String test if the declared type exist (call forName) test if the class is of the declared type
(decalreType.isAssignableFrom) test if the VALUE_TYPE ends with "Value" test if the VALUE_TYPE corresponding
interface exists

test_GetAttributeNames_NamesStartWithLowerCase

```
public void test_GetAttributeNames_NamesStartWithLowerCase()  
    throws Exception
```

tests that attributes names have length > 0 and start with lowercase letter

test_GetOptionalAttributeNames_InAllAttributes

```
public void test_GetOptionalAttributeNames_InAllAttributes()  
    throws Exception
```

tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with
lowercase letter

(continued from last page)

test_Positive_IsPopulated_AttributeNameAllDomainValues

```
public void test_Positive_IsPopulated_AttributeNameAllDomainValues()  
    throws Exception
```

DOMAINTEST tests isPopulated for all attributes in getAttributeNames()

test_Negative_IsPopulated_AttributeNameNullValue

```
public void test_Negative_IsPopulated_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of isPopulated(null)

test_Negative_IsPopulated_AttributeNameIllegalValue

```
public void test_Negative_IsPopulated_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of isPopulated(a) for String a not in getAttributeNames()

test_GetSettableAttributeNames_SubsetOfGetAttributeNames

```
public void test_GetSettableAttributeNames_SubsetOfGetAttributeNames()  
    throws Exception
```

tests that getSettableAttributesNames is a subset of getAttributeNames

test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues

```
public void test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues()  
    throws Exception
```

DOMAINTEST tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()

test_Negative_UnpopulateAttribute_AttributeNameNullValue

```
public void test_Negative_UnpopulateAttribute_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of unpopulatedAttribute(null)

test_Negative_UnpopulateAttribute_AttributeNameIllegalValue

```
public void test_Negative_UnpopulateAttribute_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of unpopulatedAttribute(a) for a not in getAttributeNames

test_UnpopulateAllAttributes_IsPopulatedFalseForAll

```
public void test_UnpopulateAllAttributes_IsPopulatedFalseForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

(continued from last page)

test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty

```
public void test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()  
    throws Exception
```

tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty

test_UnpopulateAllAttributes_GetAttributeValueFailsForAll

```
public void test_UnpopulateAllAttributes_GetAttributeValueFailsForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames

test_Negative_SetAttributeValue_AttributeNameNullValue

```
public void test_Negative_SetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

DOMAINTTEST tests failure of setAttributeValue(a) for null

test_Negative_SetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_SetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTTEST tests failure of setAttributeValue(a) for a not in getAttributeNames()

test_Negative_GetAttributeValue_AttributeNameNullValue

```
public void test_Negative_GetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

DOMAINTTEST tests failure of getAttributeValue(null)

test_Negative_GetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_GetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTTEST tests failure of getAttributeValue(a) for a not in getAttributeNames();

test_Positive_SetManagedEntityKey_DomainValue

```
public void test_Positive_SetManagedEntityKey_DomainValue()  
    throws Exception
```

tests a domain value

test_Positive_SetManagedEntityKey_IllegalKey

```
public void test_Positive_SetManagedEntityKey_IllegalKey()  
    throws Exception
```

tests a illegal key

(continued from last page)

test_Positive_SetAttributeValueKEY_IllegalKey

```
public void test_Positive_SetAttributeValueKEY_IllegalKey()  
    throws Exception
```

tests a illegal key

test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY

```
public void test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY()  
    throws Exception
```

tests that succesful setManagedEntityKey(val) implies KEY in getPopulatedAttributeNames()

test_SetManagedEntityKey_IsPopulatedKEY

```
public void test_SetManagedEntityKey_IsPopulatedKEY()  
    throws Exception
```

tests that succesful setManagedEntityKey(val) implies isPopulated(KEY)

test_GetAttributeNames_ContainsKEY

```
public void test_GetAttributeNames_ContainsKEY()  
    throws Exception
```

setManagedEntityKey

test_GetSettableAttributeNames_NotContainsKEY

```
public void test_GetSettableAttributeNames_NotContainsKEY()  
    throws Exception
```

test_Positive_ManagedEntityKey_IllegalKey

```
public void test_Positive_ManagedEntityKey_IllegalKey()  
    throws Exception
```

test_Positive_SetAttributeValue_PrimitiveMINMAXValues

```
public void test_Positive_SetAttributeValue_PrimitiveMINMAXValues()  
    throws Exception
```

tests setAttributeValue(min/max) for primitive attributes that are not enums

test_Negative_SetAttributeValue_PrimitiveNullValue

```
public void test_Negative_SetAttributeValue_PrimitiveNullValue()  
    throws Exception
```

DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

test_Positive_SetAttributeValue_DomainValue

```
public void test_Positive_SetAttributeValue_DomainValue()  
    throws Exception
```

test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName

```
public void test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()  
    throws Exception
```

tests that setAttributeValue(a) => a in getPopulatedAttributes()

test_SetAttributeValue_IsPopulated

```
public void test_SetAttributeValue_IsPopulated()  
    throws Exception
```

tests that setAttributeValue(a) => isPopulated(a)

test_Negative_SetAttributeValue_IllegalObject

```
public void test_Negative_SetAttributeValue_IllegalObject()  
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()

test_Positive_testCloneEqualsImplementation

```
public void test_Positive_testCloneEqualsImplementation()  
    throws Exception
```

Test clone and equals method implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true
a.equals(b) == true b.equals(a) == true

ossj.common.tck.tests

Class TestMev

```

java.lang.Object
  |
  +- junit.framework.Assert
      |
      +- junit.framework.TestCase
          |
          +- ossj.common.tck.JVTTestBase
              |
              +- ossj.common.tck.tests.TestMev
  
```

All Implemented Interfaces:

junit.framework.Test

Direct Known Subclasses:

[TestUpvSingle](#), [TestQueryValueSingle](#), [TestManagedEntityValueSingle](#), [TestEventValue](#), [TestCbeBasics](#)

public class **TestMev**
 extends [JVTTestBase](#)

Test case for ManagedEntityValues

Author:

Vincent Perrot, Sun Microsystems Inc.

Version:

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Method Summary

void	setUp() sets up the TestCase
void	test_ClassNameAndTypeAndValueKeyPair (javax.oss.AttributeAccess mev) Test Value and Value type and key pattern test if VALUE_TYPE field is declared test it VALUE_TYPE field is public final static String test if the declared type exist (call forName) test if the class is of the declared type (decalreType.isAssignableFrom) test if the VALUE_TYPE ends with "Value" test if the VALUE_TYPE corresponding key interface exists test if the get/set/make methods exist and use/return the correct key type test if set(make) don't throw exceptions
void	test_GetAttributeNames_ContainsKEY (javax.oss.AttributeAccess ac) this method tries to check weather the getattributes return KEY as attribute or not
void	test_GetAttributeNames_NamesStartWithLowerCase (javax.oss.AttributeAccess mev) tests that attributes names have length > 0 and start with lowercase letter - call getAttributeNames check if return array length>0 and != null - Fall all attribute names check name length >0 name following bean convention bean spec: However to support the occasional use of all upper-case names, we check if the first two characters of the name are both upper case and if so leave it alone.

void	test_GetOptionalAttributeNames_InAllAttributes (javax.oss.AttributeAccess mev) tests same as test_GetAttributeNames_NamesStartWithLowerCase for optional attributes
void	test_GetSettableAttributeNames_NotContainsKEY (javax.oss.AttributeAccess ac) The KEY attribute is usually set on the server side, so this test just looks at the absence of the "KEY" attribute from the settables attribute list
void	test_GetSettableAttributeNames_SubsetOfGetAttributeNames (javax.oss.AttributeAccess ac) tests that getSettableAttributesNames is a subset of getAttributeNames< Get the settable attribute list and check that all attribute names are also present in the list returned by getAttributeNames
void	test_Negative_GetAttributeValue_AttributeNameIllegalValue (javax.oss.AttributeAccess mev) tests failure of getAttributeValue(invalid_argument)
void	test_Negative_GetAttributeValue_AttributeNameNullValue (javax.oss.AttributeAccess mev) tests failure of getAttributeValue(null) Succeeded if java.lang.IllegalArgumentException is thrown
void	test_Negative_IsPopulated_AttributeNameIllegalValue (javax.oss.AttributeAccess mev) tests failure of isPopulated(a) for String a not in getAttributeNames() Succeeded if java.lang.IllegalArgumentException or javax.oss.OssUnsupportedAttributeException is thrown
void	test_Negative_IsPopulated_AttributeNameNullValue (javax.oss.AttributeAccess mev) tests failure of isPopulated(null) Succeeded if java.lang.IllegalArgumentException is thrown
void	test_Negative_SetAttributeValue_AttributeNameIllegalValue (javax.oss.AttributeAccess mev) tests failure of setAttributeValue(invalid_argument, value) Succeeded if java.lang.IllegalArgumentException is thrown
void	test_Negative_SetAttributeValue_AttributeNameNullValue (javax.oss.AttributeAccess mev) tests failure of setAttributeValue(null, value) Succeeded if java.lang.IllegalArgumentException is thrown
void	test_Negative_SetAttributeValue_IllegalObject (javax.oss.AttributeAccess mev) DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()
void	test_Negative_SetAttributeValue_PrimitiveNullValue (javax.oss.AttributeAccess mev) DOCUMENT ME!
void	test_Negative_UnpopulateAttribute_AttributeNameIllegalValue (javax.oss.AttributeAccess mev) tests failure of unpopulateAttribute(a) for an argument not in getAttributeNames Succeeded if java.lang.IllegalArgumentException is thrown
void	test_Negative_UnpopulateAttribute_AttributeNameNullValue (javax.oss.AttributeAccess mev) tests failure of unpopulateAttribute(null) Succeeded if java.lang.IllegalArgumentException is thrown

void	test_Positive_IsPopulated_AttributeNameAllDomainValues (javax.oss.AttributeAccess mev) tests isPopulated for all attributes in getAttributeNames() call ispopulated for all attributes fails if an exception is thrown
void	test_Positive_ManagedEntityKey_IllegalKey (javax.oss.AttributeAccess ac)
void	test_Positive_SetAttributeValue_DomainValue (javax.oss.AttributeAccess mev)
void	test_Positive_SetAttributeValue_PrimitiveMINMAXValues (javax.oss.AttributeAccess mev) tests setAttributeValue(min/max) for primitive attributes that are not enums
void	test_Positive_SetAttributeValueKEY_IllegalKey (javax.oss.AttributeAccess ac) tests a illegal key
void	test_Positive_SetManagedEntityKey_DomainValue (javax.oss.AttributeAccess ac) tests based key management Call setManagedEntityKey(mev.makeManagedEntityKey());
void	test_Positive_SetManagedEntityKey_IllegalKey (javax.oss.AttributeAccess ac) tests illegal keys Illegal keys are:
void	test_Positive_testCloneEqualsImplementation (javax.oss.AttributeAccess mev) Test clone and equals methods CBE implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true a.equals(b) == true b.equals(a) == true
void	test_Positive_testCloneEqualsImplementation (Object mev) Test clone and equals methods CBE implementations Expected Result:
void	test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues (javax.oss.AttributeAccess mev) tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()
void	test_Query_Type (javax.oss.AttributeAccess mev) Test QUERY_TYPE test if QUERY_TYPE field is declared test it QUERY_TYPE field is public final static String test if the declared type exist (call forName) test if the class is of the declared type (decalreType.isAssignableFrom) test if the QUERY_TYPE corresponding interface exists
void	test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName (javax.oss.AttributeAccess mev) tests that setAttributeValue(a) => a in getPopulatedAttributes()
void	test_SetAttributeValue_IsPopulated (javax.oss.AttributeAccess mev) tests that setAttributeValue(a) => isPopulated(a)
void	test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY (javax.oss.AttributeAccess ac) tests that succesful setManagedEntityKey(val) implies KEY in getPopulatedAttributeNames()
void	test_SetManagedEntityKey_IsPopulatedKEY (javax.oss.AttributeAccess ac) tests that succesful setManagedEntityKey(val) implies isPopulated(KEY)

assertTrue, assertTrue, assertTrueMsg, fail, fail, getEnvironment, getInstanceOf,
init, log, log, run, run

```
countTestCases, createResult, getName, run, run, runBare, runTest, setName, setUp,
tearDown, toString
```

```
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,
assertEquals, assertEquals, assertFalse, assertFalse, assertNotNull, assertNotNull,
assertNotSame, assertNotSame, assertNull, assertNull, assertEquals, assertEquals,
assertTrue, assertTrue, fail, fail
```

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

```
countTestCases, run
```

setUp

sets up the TestCase

(continued from last page)

test_ClassNameAndTypeAndValueKeyPair

```
public void test_ClassNameAndTypeAndValueKeyPair( javax.oss.AttributeAccess mev)
    throws Exception
```

Test Value and Value type and key pattern
test if VALUE_TYPE field is declared
test if VALUE_TYPE field is public final static String
test if the declared type exist (call forName)
test if the class is of the declared type (decalreType.isAssignableFrom)
test if the VALUE_TYPE ends with "Value"
test if the VALUE_TYPE corresponding key interface exists
test if the get/set/make methods exist and use/return the correct key type
test if set(make) don't throw exceptions

test_Query_Type

```
public void test_Query_Type( javax.oss.AttributeAccess mev)
    throws Exception
```

Test QUERY_TYPE

test if QUERY_TYPE field is declared
test if QUERY_TYPE field is public final static String
test if the declared type exist (call forName)
test if the class is of the declared type (decalreType.isAssignableFrom)
test if the QUERY_TYPE corresponding interface exists

test_upv_Type

```
public void test_upv_Type( javax.oss.AttributeAccess mev)
    throws Exception
```

Test UPDATE_TYPE

test if UPDATE_TYPE field is declared
test if UPDATE_TYPE field is public final static String
test if the declared type exist (call forName)
test if the class is of the declared type (decalreType.isAssignableFrom)
test if the UPDATE_TYPE corresponding interface exists

test_GetAttributeNames_NamesStartWithLowerCase

```
public void test_GetAttributeNames_NamesStartWithLowerCase( javax.oss.AttributeAccess mev)
    throws Exception
```

tests that attributes names have length > 0 and start with lowercase letter

- call getAttributeNames check if return array length>0 and != null
- Fall all attribute names check name length >0 name following bean convention
bean spec: However to support the occasional use of all upper-case names, we check if the first two characters of the name are both upper case and if so leave it alone.

test_GetOptionalAttributeNames_InAllAttributes

```
public void test_GetOptionalAttributeNames_InAllAttributes( javax.oss.AttributeAccess mev)
    throws Exception
```

tests same as test_GetAttributeNames_NamesStartWithLowerCase for optional attributes

(continued from last page)

See Also:[test_GetAttributeNames_NamesStartWithLowerCase\(AttributeAccess\)](#)

test_Positive_IsPopulated_AttributeNameAllDomainValues

```
public void
test_Positive_IsPopulated_AttributeNameAllDomainValues( javax.oss.AttributeAccess mev)
    throws Exception
```

tests isPopulated for all attributes in getAttributeNames()
 call isPopulated for all attributes fails if an exception is thrown

test_Negative_IsPopulated_AttributeNameNullValue

```
public void test_Negative_IsPopulated_AttributeNameNullValue( javax.oss.AttributeAccess
mev)
    throws Exception
```

tests failure of isPopulated(null)
 Succeeded if java.lang.IllegalArgumentException is thrown

test_Negative_IsPopulated_AttributeNameIllegalValue

```
public void
test_Negative_IsPopulated_AttributeNameIllegalValue( javax.oss.AttributeAccess mev)
    throws Exception
```

tests failure of isPopulated(a) for String a not in getAttributeNames()
 Succeeded if java.lang.IllegalArgumentException or javax.oss.OssUnsupportedAttributeException is thrown

test_GetSettableAttributeNames_SubsetOfGetAttributeNames

```
public void
test_GetSettableAttributeNames_SubsetOfGetAttributeNames( javax.oss.AttributeAccess ac)
    throws Exception
```

tests that getSettableAttributesNames is a subset of getAttributeNames<
 Get the settable attribute list and check that
 all attribute names are also present in the list returned by getAttributeNames

test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues

```
public void
test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues( javax.oss.AttributeAcce
ss mev)
    throws Exception
```

tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()

test_Negative_UnpopulateAttribute_AttributeNameNullValue

```
public void
test_Negative_UnpopulateAttribute_AttributeNameNullValue( javax.oss.AttributeAccess
mev)
    throws Exception
```

tests failure of unpopulatedAttribute(null)
 Succeeded if java.lang.IllegalArgumentException is thrown

(continued from last page)

test_Negative_UnpopulateAttribute_AttributeNameIllegalValue

```
public void
test_Negative_UnpopulateAttribute_AttributeNameIllegalValue( javax.oss.AttributeAccess
mev)
    throws Exception
```

tests failure of unpopulateAttribute(a) for an argument not in getAttributeNames
Succeeded if java.lang.IllegalArgumentException is thrown

test_UnpopulateAllAttributes_IsPopulatedFalseForAll

```
public void
test_UnpopulateAllAttributes_IsPopulatedFalseForAll( javax.oss.AttributeAccess mev)
    throws Exception
```

tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty

```
public void
test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty( javax.oss.AttributeAcc
ess mev)
    throws Exception
```

tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty

test_UnpopulateAllAttributes_GetAttributeValueFailsForAll

```
public void
test_UnpopulateAllAttributes_GetAttributeValueFailsForAll( javax.oss.AttributeAccess
mev)
    throws Exception
```

tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames

test_Negative_SetAttributeValue_AttributeNameNullValue

```
public void
test_Negative_SetAttributeValue_AttributeNameNullValue( javax.oss.AttributeAccess mev)
    throws Exception
```

tests failure of setAttributeValue(null, value)
Succeeded if java.lang.IllegalArgumentException is thrown

test_Negative_SetAttributeValue_AttributeNameIllegalValue

```
public void
test_Negative_SetAttributeValue_AttributeNameIllegalValue( javax.oss.AttributeAccess
mev)
    throws Exception
```

tests failure of setAttributeValue(invalid_argument, value) Succeeded if java.lang.IllegalArgumentException is thrown

test_Negative_GetAttributeValue_AttributeNameNullValue

```
public void
test_Negative_GetAttributeValue_AttributeNameNullValue( javax.oss.AttributeAccess mev)
    throws Exception
```

tests failure of getAttributeValue(null) Succeeded if java.lang.IllegalArgumentException is thrown

test_Negative_GetAttributeValue_AttributeNameIllegalValue

```
public void
test_Negative_GetAttributeValue_AttributeNameIllegalValue( javax.oss.AttributeAccess
mev)
    throws Exception

    tests failure of getAttributeValue(invalid_argument)
```

test_Positive_SetManagedEntityKey_DomainValue

```
public void test_Positive_SetManagedEntityKey_DomainValue( javax.oss.AttributeAccess
ac)
    throws Exception

    tests based key management
    Call setManagedEntityKey(mev.makeManagedEntityKey());
```

test_Positive_SetManagedEntityKey_IllegalKey

```
public void test_Positive_SetManagedEntityKey_IllegalKey( javax.oss.AttributeAccess ac)
    throws Exception

    tests illegal keys
    Illegal keys are:

    • type=""
    • type=invalid_prefix+getType()
    • promary key = new object()
    • primaryKey-newObject + invalid type
```

test_Positive_SetAttributeValueKEY_IllegalKey

```
public void test_Positive_SetAttributeValueKEY_IllegalKey( javax.oss.AttributeAccess
ac)
    throws Exception

    tests a illegal key
```

test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY

```
public void
test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY( javax.oss.AttributeAcce
ss ac)
    throws Exception

    tests that succesful setManagedEntityKey(val) implies KEY in getPopulatedAttributeNames()
```

test_SetManagedEntityKey_IsPopulatedKEY

```
public void test_SetManagedEntityKey_IsPopulatedKEY( javax.oss.AttributeAccess ac)
    throws Exception

    tests that succesful setManagedEntityKey(val) implies isPopulated(KEY)
```

(continued from last page)

test_GetAttributeNames_ContainsKEY

```
public void test_GetAttributeNames_ContainsKEY( javax.oss.AttributeAccess ac)
    throws Exception
```

this method tries to check weather the getattributes return KEY as attribute or not

test_GetSettableAttributeNames_NotContainsKEY

```
public void test_GetSettableAttributeNames_NotContainsKEY( javax.oss.AttributeAccess
ac)
    throws Exception
```

The KEY attribute is usually set on the server side, so this test just looks at the absence of the "KEY" attribute from the settables attribute list

test_Positive_ManagedEntityKey_IllegalKey

```
public void test_Positive_ManagedEntityKey_IllegalKey( javax.oss.AttributeAccess ac)
    throws Exception
```

test_Positive_SetAttributeValue_PrimitiveMINMAXValues

```
public void
test_Positive_SetAttributeValue_PrimitiveMINMAXValues( javax.oss.AttributeAccess mev)
    throws Exception
```

tests setAttributeValue(min/max) for primitive attributes that are not enums

test_Negative_SetAttributeValue_PrimitiveNullValue

```
public void
test_Negative_SetAttributeValue_PrimitiveNullValue( javax.oss.AttributeAccess mev)
    throws Exception
```

DOCUMENT ME!

Parameters:

mev - DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

[TCKException](#) - DOCUMENT ME!

test_Positive_SetAttributeValue_DomainValue

```
public void test_Positive_SetAttributeValue_DomainValue( javax.oss.AttributeAccess mev)
    throws Exception
```

test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName

```
public void
test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName( javax.oss.Attr
ibuteAccess mev)
    throws Exception
```

tests that setAttributeValue(a) => a in getPopulatedAttributes()

test_SetAttributeValue_IsPopulated

```
public void test_SetAttributeValue_IsPopulated( javax.oss.AttributeAccess mev)
    throws Exception
```

tests that setAttributeValue(a) => isPopulated(a)

test_Negative_SetAttributeValue_IllegalObject

```
public void test_Negative_SetAttributeValue_IllegalObject( javax.oss.AttributeAccess
mev)
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()

test_Positive_testCloneEqualsImplementation

```
public void test_Positive_testCloneEqualsImplementation( javax.oss.AttributeAccess mev)
    throws Exception
```

Test clone and equals methods CBE implementations

Expected Result:

```
a = b.clone()
a.equals(a) == true
b.equals(b) == true
a.equals(b) == true
b.equals(a) == true
```

test_Positive_testCloneEqualsImplementation

```
public void test_Positive_testCloneEqualsImplementation(Object mev)
    throws Exception
```

Test clone and equals methods CBE implementations

Expected Result:

```
new a;
a = b.clone()
a.equals(a) == true
b.equals(b) == true
a.equals(b) == true
b.equals(a) == true

set random attribute in a
a = b.clone()
a.equals(a) == true
b.equals(b) == true
a.equals(b) == true
b.equals(a) == true
```

Throws:

Exception - DOCUMENT ME!

(continued from last page)

[TCKException](#) - DOCUMENT ME!

ossj.common.tck.tests

Class TestQueryValueSingle

```
java.lang.Object
├-- junit.framework.Assert
│   └-- junit.framework.TestCase
│       ├── ossj.common.tck.JVTTestBase
│       │   └-- ossj.common.tck.tests.TestMev
│       │       └-- ossj.common.tck.tests.TestQueryValueSingle
```

All Implemented Interfaces:

junit.framework.Test

public class **TestQueryValueSingle**
extends [TestMev](#)

Test case for ManagedEntityValues

Author:

Version:

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Method Summary

void	doGeneric (String testName) genericTest
void	setUp () sets up the TestCase
static junit.framework.TestSuite	suite ()
void	test_Negative_query_GetAttributeValue_AttributeNameIllegalValue () DOMAINTTEST tests failure of getAttributeValue(a) for a not in getAttributeValues();
void	test_Negative_query_GetAttributeValue_AttributeNameNullValue () DOMAINTTEST tests failure of getAttributeValue(null)
void	test_Negative_query_IsPopulated_AttributeNameIllegalValue () DOMAINTTEST tests failure of isPopulated(a) for String a not in getAttributeNames()
void	test_Negative_query_IsPopulated_AttributeNameNullValue () DOMAINTTEST tests failure of isPopulated(null)

void	<u>test_Negative_query_SetAttributeValue_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of setAttributeValue(a) for a not in getAttributeNames()
void	<u>test_Negative_query_SetAttributeValue_AttributeNameNullValue()</u> DOMAINTEST tests failure of setAttributeValue(a) for null
void	<u>test_Negative_query_SetAttributeValue_IllegalObject()</u> DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()
void	<u>test_Negative_query_SetAttributeValue_PrimitiveNullValue()</u> DOCUMENT ME!
void	<u>test_Negative_query_UnpopulateAttribute_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of unpopulatedAttribute(a) for a not in getAttributeNames()
void	<u>test_Negative_query_UnpopulateAttribute_AttributeNameNullValue()</u> DOMAINTEST tests failure of unpopulatedAttribute(null)
void	<u>test_Positive_query_IsPopulated_AttributeNameAllDomainValues()</u> DOMAINTEST tests isPopulated for all attributes in getAttributeNames()
void	<u>test_Positive_query_SetAttributeValue_DomainValue()</u>
void	<u>test_Positive_query_SetAttributeValue_PrimitiveMINMAXValues()</u> tests setAttributeValue(min/max) for primitive attributes that are not enums
void	<u>test_Positive_query_testCloneEqualsImplementation()</u> Test clone and equals method implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true a.equals(b) == true b.equals(a) == true
void	<u>test_Positive_query_UnpopulateAttribute_AttributeNameAllDomainValues()</u> DOMAINTEST tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()
void	<u>test_query_GetAttributeNames_NamesStartWithLowerCase()</u> tests that attributes names have length > 0 and start with lowercase letter
void	<u>test_query_GetOptionalAttributeNames_InAllAttributes()</u> tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter
void	<u>test_query_Query_Type()</u> tests the public final static declaration QUERY_TYPE
void	<u>test_query_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()</u> tests that setAttributeValue(a) => a in getPopulatedAttributes()
void	<u>test_query_SetAttributeValue_IsPopulated()</u> tests that setAttributeValue(a) => isPopulated(a)
void	<u>test_query_UnpopulateAllAttributes_GetAttributeValueFailsForAll()</u> tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames
void	<u>test_query_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()</u> tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty
void	<u>test_query_UnpopulateAllAttributes_IsPopulatedFalseForAll()</u> tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

Methods inherited from class [ossj.common.tck.tests.TestMev](#)

[setUp](#), [test_ClassNameAndTypeAndValueKeyPair](#), [test_GetAttributeNames_ContainsKEY](#),
[test_GetAttributeNames_NamesStartWithLowerCase](#),
[test_GetOptionalAttributeNames_InAllAttributes](#),
[test_GetSettableAttributeNames_NotContainsKEY](#),
[test_GetSettableAttributeNames_SubsetOfGetAttributeNames](#),
[test_Negative_GetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_GetAttributeValue_AttributeNameNullValue](#),
[test_Negative_IsPopulated_AttributeNameIllegalValue](#),
[test_Negative_IsPopulated_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_SetAttributeValue_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_IllegalObject](#),
[test_Negative_SetAttributeValue_PrimitiveNullValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameIllegalValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameNullValue](#),
[test_Positive_IsPopulated_AttributeNameAllDomainValues](#),
[test_Positive_ManagedEntityKey_IllegalKey](#),
[test_Positive_SetAttributeValue_DomainValue](#),
[test_Positive_SetAttributeValue_PrimitiveMINMAXValues](#),
[test_Positive_SetAttributeValueKEY_IllegalKey](#),
[test_Positive_SetManagedEntityKey_DomainValue](#),
[test_Positive_SetManagedEntityKey_IllegalKey](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues](#), [test_Query_Type](#),
[test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName](#),
[test_SetAttributeValue_IsPopulated](#),
[test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY](#),
[test_SetManagedEntityKey_IsPopulatedKEY](#),
[test_UnpopulateAllAttributes_GetAttributeValueFailsForAll](#),
[test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty](#),
[test_UnpopulateAllAttributes_IsPopulatedFalseForAll](#), [test_upv_Type](#)

Methods inherited from class [ossj.common.tck.JVTTestBase](#)

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#),
[init](#), [log](#), [log](#), [run](#), [run](#)

Methods inherited from class [junit.framework.TestCase](#)

[countTestCases](#), [createResult](#), [getName](#), [run](#), [run](#), [runBare](#), [runTest](#), [setName](#), [setUp](#),
[tearDown](#), [toString](#)

Methods inherited from class [junit.framework.Assert](#)

[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertFalse](#), [assertFalse](#), [assertNotNull](#), [assertNotNull](#),
[assertNotSame](#), [assertNotSame](#), [assertNull](#), [assertNull](#), [assertSame](#), [assertSame](#),
[assertTrue](#), [assertTrue](#), [fail](#), [fail](#)

Methods inherited from class [java.lang.Object](#)

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface junit.framework.Test

countTestCases, run

Methods

setUp

protected void **setUp**()
throws Exception

sets up the TestCase

suite

public static junit.framework.TestSuite **suite**()
throws Exception

Returns:

All suites of this class

doGeneric

protected void **doGeneric**(String testName)
throws Exception

genericTest

test_query_Query_Type

public void **test_query_Query_Type**()
throws Exception

tests the public final static declaration QUERY_TYPE

test_query_GetAttributeNames_NamesStartWithLowerCase

public void **test_query_GetAttributeNames_NamesStartWithLowerCase**()
throws Exception

tests that attributes names have length > 0 and start with lowercase letter

test_query_GetOptionalAttributeNames_InAllAttributes

public void **test_query_GetOptionalAttributeNames_InAllAttributes**()
throws Exception

tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter

(continued from last page)

test_Positive_query_IsPopulated_AttributeNameAllDomainValues

```
public void test_Positive_query_IsPopulated_AttributeNameAllDomainValues()  
    throws Exception
```

DOMAINTEST tests isPopulated for all attributes in getAttributeNames()

test_Negative_query_IsPopulated_AttributeNameNullValue

```
public void test_Negative_query_IsPopulated_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of isPopulated(null)

test_Negative_query_IsPopulated_AttributeNameIllegalValue

```
public void test_Negative_query_IsPopulated_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of isPopulated(a) for String a not in getAttributeNames()

test_Positive_query_UnpopulateAttribute_AttributeNameAllDomainValues

```
public void test_Positive_query_UnpopulateAttribute_AttributeNameAllDomainValues()  
    throws Exception
```

DOMAINTEST tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()

test_Negative_query_UnpopulateAttribute_AttributeNameNullValue

```
public void test_Negative_query_UnpopulateAttribute_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of unpopulatedAttribute(null)

test_Negative_query_UnpopulateAttribute_AttributeNameIllegalValue

```
public void test_Negative_query_UnpopulateAttribute_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of unpopulatedAttribute(a) for a not in getAttributeNames

test_query_UnpopulateAllAttributes_IsPopulatedFalseForAll

```
public void test_query_UnpopulateAllAttributes_IsPopulatedFalseForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

test_query_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty

```
public void test_query_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()  
    throws Exception
```

tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty

(continued from last page)

test_query_UnpopulateAllAttributes_GetAttributeValueFailsForAll

```
public void test_query_UnpopulateAllAttributes_GetAttributeValueFailsForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames

test_Negative_query_SetAttributeValue_AttributeNameNullValue

```
public void test_Negative_query_SetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for null

test_Negative_query_SetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_query_SetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for a not in getAttributeNames()

test_Negative_query_GetAttributeValue_AttributeNameNullValue

```
public void test_Negative_query_GetAttributeValue_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of getAttributeValue(null)

test_Negative_query_GetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_query_GetAttributeValue_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of getAttributeValue(a) for a not in getAttributeNames();

test_Positive_query_SetAttributeValue_PrimitiveMINMAXValues

```
public void test_Positive_query_SetAttributeValue_PrimitiveMINMAXValues()  
    throws Exception
```

tests setAttributeValue(min/max) for primitive attributes that are not enums

test_Negative_query_SetAttributeValue_PrimitiveNullValue

```
public void test_Negative_query_SetAttributeValue_PrimitiveNullValue()  
    throws Exception
```

DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

test_Positive_query_SetAttributeValue_DomainValue

```
public void test_Positive_query_SetAttributeValue_DomainValue()  
    throws Exception
```

test_query_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName

```
public void  
test_query_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()  
    throws Exception
```

tests that setAttributeValue(a) => a in getPopulatedAttributes()

test_query_SetAttributeValue_IsPopulated

```
public void test_query_SetAttributeValue_IsPopulated()  
    throws Exception
```

tests that setAttributeValue(a) => isPopulated(a)

test_Negative_query_SetAttributeValue_IllegalObject

```
public void test_Negative_query_SetAttributeValue_IllegalObject()  
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()

test_Positive_query_testCloneEqualsImplementation

```
public void test_Positive_query_testCloneEqualsImplementation()  
    throws Exception
```

Test clone and equals method implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true
a.equals(b) == true b.equals(a) == true

ossj.common.tck.tests

Class TestUpvSingle

```

java.lang.Object
├── junit.framework.Assert
│   ├── junit.framework.TestCase
│   │   ├── ossj.common.tck.JVTTestBase
│   │   │   ├── ossj.common.tck.tests.TestMev
│   │   │   │   └── TestUpvSingle

```

All Implemented Interfaces:

junit.framework.Test

```

public class TestUpvSingle
extends TestMev

```

Test case for ManagedEntityValues

Author:

Version:

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Method Summary

void	doGeneric (String testName) genericTest
void	setUp () sets up the TestCase
static junit.framework.TestSuite	suite ()
void	test_Negative_upv_GetAttributeValue_AttributeNameIllegalValue () DOMAINTEST tests failure of getAttributeValue(a) for a not in getAttributeValues();
void	test_Negative_upv_GetAttributeValue_AttributeNameNullValue () DOMAINTEST tests failure of getAttributeValue(null)
void	test_Negative_upv_IsPopulated_AttributeNameIllegalValue () DOMAINTEST tests failure of isPopulated(a) for String a not in getAttributeNames()
void	test_Negative_upv_IsPopulated_AttributeNameNullValue () DOMAINTEST tests failure of isPopulated(null)

void	<u>test_Negative_upv_SetAttributeValue_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of setAttributeValue(a) for a not in getAttributeNames()
void	<u>test_Negative_upv_SetAttributeValue_AttributeNameNullValue()</u> DOMAINTEST tests failure of setAttributeValue(a) for null
void	<u>test_Negative_upv_SetAttributeValue_IllegalObject()</u> DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()
void	<u>test_Negative_upv_SetAttributeValue_PrimitiveNullValue()</u> DOCUMENT ME!
void	<u>test_Negative_upv_UnpopulateAttribute_AttributeNameIllegalValue()</u> DOMAINTEST tests failure of unpopulatedAttribute(a) for a not in getAttributeNames()
void	<u>test_Negative_upv_UnpopulateAttribute_AttributeNameNullValue()</u> DOMAINTEST tests failure of unpopulatedAttribute(null)
void	<u>test_Positive_upv_IsPopulated_AttributeNameAllDomainValues()</u> DOMAINTEST tests isPopulated for all attributes in getAttributeNames()
void	<u>test_Positive_upv_SetAttributeValue_DomainValue()</u>
void	<u>test_Positive_upv_SetAttributeValue_PrimitiveMINMAXValues()</u> tests setAttributeValue(min/max) for primitive attributes that are not enums
void	<u>test_Positive_upv_testCloneEqualsImplementation()</u> Test clone and equals method implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true a.equals(b) == true b.equals(a) == true
void	<u>test_Positive_upv_UnpopulateAttribute_AttributeNameAllDomainValues()</u> DOMAINTEST tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()
void	<u>test_upv_GetAttributeNames_NamesStartWithLowerCase()</u> tests that attributes names have length > 0 and start with lowercase letter
void	<u>test_upv_GetOptionalAttributeNames_InAllAttributes()</u> tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter
void	<u>test_upv_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()</u> tests that setAttributeValue(a) => a in getPopulatedAttributes()
void	<u>test_upv_SetAttributeValue_IsPopulated()</u> tests that setAttributeValue(a) => isPopulated(a)
void	<u>test_upv_Type()</u> tests the public final static declaration UPDATE_TYPE
void	<u>test_upv_UnpopulateAllAttributes_GetAttributeValueFailsForAll()</u> tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames()
void	<u>test_upv_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()</u> tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty
void	<u>test_upv_UnpopulateAllAttributes_IsPopulatedFalseForAll()</u> tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

Methods inherited from class [ossj.common.tck.tests.TestMev](#)

[setUp](#), [test_ClassNameAndTypeAndValueKeyPair](#), [test_GetAttributeNames_ContainsKEY](#),
[test_GetAttributeNames_NamesStartWithLowerCase](#),
[test_GetOptionalAttributeNames_InAllAttributes](#),
[test_GetSettableAttributeNames_NotContainsKEY](#),
[test_GetSettableAttributeNames_SubsetOfGetAttributeNames](#),
[test_Negative_GetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_GetAttributeValue_AttributeNameNullValue](#),
[test_Negative_IsPopulated_AttributeNameIllegalValue](#),
[test_Negative_IsPopulated_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_AttributeNameIllegalValue](#),
[test_Negative_SetAttributeValue_AttributeNameNullValue](#),
[test_Negative_SetAttributeValue_IllegalObject](#),
[test_Negative_SetAttributeValue_PrimitiveNullValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameIllegalValue](#),
[test_Negative_UnpopulateAttribute_AttributeNameNullValue](#),
[test_Positive_IsPopulated_AttributeNameAllDomainValues](#),
[test_Positive_ManagedEntityKey_IllegalKey](#),
[test_Positive_SetAttributeValue_DomainValue](#),
[test_Positive_SetAttributeValue_PrimitiveMINMAXValues](#),
[test_Positive_SetAttributeValueKEY_IllegalKey](#),
[test_Positive_SetManagedEntityKey_DomainValue](#),
[test_Positive_SetManagedEntityKey_IllegalKey](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_testCloneEqualsImplementation](#),
[test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues](#), [test_Query_Type](#),
[test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName](#),
[test_SetAttributeValue_IsPopulated](#),
[test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY](#),
[test_SetManagedEntityKey_IsPopulatedKEY](#),
[test_UnpopulateAllAttributes_GetAttributeValueFailsForAll](#),
[test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty](#),
[test_UnpopulateAllAttributes_IsPopulatedFalseForAll](#), [test_upv_Type](#)

Methods inherited from class [ossj.common.tck.JVTTestBase](#)

[assertTrue](#), [assertTrue](#), [assertTrueMsg](#), [fail](#), [fail](#), [getEnvironment](#), [getInstanceOf](#),
[init](#), [log](#), [log](#), [run](#), [run](#)

Methods inherited from class [junit.framework.TestCase](#)

[countTestCases](#), [createResult](#), [getName](#), [run](#), [run](#), [runBare](#), [runTest](#), [setName](#), [setUp](#),
[tearDown](#), [toString](#)

Methods inherited from class [junit.framework.Assert](#)

[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#), [assertEquals](#),
[assertEquals](#), [assertEquals](#), [assertFalse](#), [assertFalse](#), [assertNotNull](#), [assertNotNull](#),
[assertNotSame](#), [assertNotSame](#), [assertNull](#), [assertNull](#), [assertSame](#), [assertSame](#),
[assertTrue](#), [assertTrue](#), [fail](#), [fail](#)

Methods inherited from class [java.lang.Object](#)

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

Methods inherited from interface junit.framework.Test

```
countTestCases, run
```

Methods

setUp

```
protected void setUp()  
    throws Exception
```

sets up the TestCase

suite

```
public static junit.framework.TestSuite suite()  
    throws Exception
```

Returns:

All suites of this class

doGeneric

```
protected void doGeneric(String testName)  
    throws Exception
```

genericTest

test_upv_Type

```
public void test_upv_Type()  
    throws Exception
```

tests the public final static declaration UPDATE_TYPE

test_upv_GetAttributeNames_NamesStartWithLowerCase

```
public void test_upv_GetAttributeNames_NamesStartWithLowerCase()  
    throws Exception
```

tests that attributes names have length > 0 and start with lowercase letter

test_upv_GetOptionalAttributeNames_InAllAttributes

```
public void test_upv_GetOptionalAttributeNames_InAllAttributes()  
    throws Exception
```

tests that all supported optional attributes are also listed in all attributes and attributes names have length > 0 and start with lowercase letter

(continued from last page)

test_Positive_upv_IsPopulated_AttributeNameAllDomainValues

```
public void test_Positive_upv_IsPopulated_AttributeNameAllDomainValues()  
    throws Exception
```

DOMAINTEST tests isPopulated for all attributes in getAttributeNames()

test_Negative_upv_IsPopulated_AttributeNameNullValue

```
public void test_Negative_upv_IsPopulated_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of isPopulated(null)

test_Negative_upv_IsPopulated_AttributeNameIllegalValue

```
public void test_Negative_upv_IsPopulated_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of isPopulated(a) for String a not in getAttributeNames()

test_Positive_upv_UnpopulateAttribute_AttributeNameAllDomainValues

```
public void test_Positive_upv_UnpopulateAttribute_AttributeNameAllDomainValues()  
    throws Exception
```

DOMAINTEST tests succesful invocation of unpopulateAttribute(a) for all a in getAttributeNames()

test_Negative_upv_UnpopulateAttribute_AttributeNameNullValue

```
public void test_Negative_upv_UnpopulateAttribute_AttributeNameNullValue()  
    throws Exception
```

DOMAINTEST tests failure of unpopulatedAttribute(null)

test_Negative_upv_UnpopulateAttribute_AttributeNameIllegalValue

```
public void test_Negative_upv_UnpopulateAttribute_AttributeNameIllegalValue()  
    throws Exception
```

DOMAINTEST tests failure of unpopulatedAttribute(a) for a not in getAttributeNames

test_upv_UnpopulateAllAttributes_IsPopulatedFalseForAll

```
public void test_upv_UnpopulateAllAttributes_IsPopulatedFalseForAll()  
    throws Exception
```

tests that unpopulateAllAttributes implies !isPopulated(a) for all a in getAttributeNames()

test_upv_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty

```
public void test_upv_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty()  
    throws Exception
```

tests that unpopulateAllAttributes implies getPopulatedAttributesNamesIsEmpty() is empty

(continued from last page)

test_upv_UnpopulateAllAttributes_GetAttributeValueFailsForAll

```
public void test_upv_UnpopulateAllAttributes_GetAttributeValueFailsForAll()
    throws Exception
```

tests that unpopulateAllAttributes implies failure of getAttributeValue(a) for all a in getAttributeNames

test_Negative_upv_SetAttributeValue_AttributeNameNullValue

```
public void test_Negative_upv_SetAttributeValue_AttributeNameNullValue()
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for null

test_Negative_upv_SetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_upv_SetAttributeValue_AttributeNameIllegalValue()
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for a not in getAttributeNames()

test_Negative_upv_GetAttributeValue_AttributeNameNullValue

```
public void test_Negative_upv_GetAttributeValue_AttributeNameNullValue()
    throws Exception
```

DOMAINTEST tests failure of getAttributeValue(null)

test_Negative_upv_GetAttributeValue_AttributeNameIllegalValue

```
public void test_Negative_upv_GetAttributeValue_AttributeNameIllegalValue()
    throws Exception
```

DOMAINTEST tests failure of getAttributeValue(a) for a not in getAttributeNames();

test_Positive_upv_SetAttributeValue_PrimitiveMINMAXValues

```
public void test_Positive_upv_SetAttributeValue_PrimitiveMINMAXValues()
    throws Exception
```

tests setAttributeValue(min/max) for primitive attributes that are not enums

test_Negative_upv_SetAttributeValue_PrimitiveNullValue

```
public void test_Negative_upv_SetAttributeValue_PrimitiveNullValue()
    throws Exception
```

DOCUMENT ME!

Throws:

Exception - DOCUMENT ME!

test_Positive_upv_SetAttributeValue_DomainValue

```
public void test_Positive_upv_SetAttributeValue_DomainValue()
    throws Exception
```

test_upv_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName

```
public void  
test_upv_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName()  
    throws Exception
```

tests that setAttributeValue(a) => a in getPopulatedAttributes()

test_upv_SetAttributeValue_IsPopulated

```
public void test_upv_SetAttributeValue_IsPopulated()  
    throws Exception
```

tests that setAttributeValue(a) => isPopulated(a)

test_Negative_upv_SetAttributeValue_IllegalObject

```
public void test_Negative_upv_SetAttributeValue_IllegalObject()  
    throws Exception
```

DOMAINTEST tests failure of setAttributeValue(a) for an illegal object()

test_Positive_upv_testCloneEqualsImplementation

```
public void test_Positive_upv_testCloneEqualsImplementation()  
    throws Exception
```

Test clone and equals method implementations Expected Result: a = b.clone() a.equals(a) == true b.equals(b) == true
a.equals(b) == true b.equals(a) == true

ossj.common.tck.tests

Class XmlTests

```

java.lang.Object
|
+- junit.framework.Assert
   |
   +- junit.framework.TestCase
      |
      +- ossj.common.tck.JVTTestBase
         |
         +- ossj.common.tck.tests.XmlTests

```

All Implemented Interfaces:

junit.framework.Test

public class **XmlTests**
 extends [JVTTestBase](#)

XmlTests Class Implement the XML Mesaging Mandatory Test of the JSR144 specification.

Author:

Copyright 2002 Sun Microsystems Inc., Inc. All rights reserved.

Version:

1.12

Field Summary

public	done
public	result

Fields inherited from class [ossj.common.tck.JVTTestBase](#)

[debug](#), [done](#), [EVENT_TIMEOUT](#), [howmany_key](#), [howmany_value](#), [JMS_FACTORY](#), [JNDI_FACT_CLASS_NAME](#), [JNDI_URL](#), [jvtSessionList](#), [LOG_FILE](#), [namingContext](#), [QUOTECHAR](#), [result](#), [return_failed_only](#), [runonce](#), [TESTFILES_HOME](#), [TOPIC_NAME](#), [util](#)

Constructor Summary

public	XmlTests (String name) Creates a new CommonAPIXMLTests object.
--------	---

Method Summary

void	setUp () Initialize every test
static junit.framework.TestS uite	suite () Build the suite of test
void	tearDown ()

```
assertTrue, assertTrue, assertTrueMsg, fail, fail, getEnvironment, getInstanceOf,  
init, log, log, run, run
```

```
countTestCases, createResult, getName, run, run, runBare, runTest, setName, setUp,
tearDown, toString
```

```
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,  
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,  
assertEquals, assertEquals, assertEquals, assertEquals, assertEquals, assertEquals,  
assertEquals, assertEquals, assertFalse, assertFalse, assertNotNull, assertNotNull,  
assertNotSame, assertNotSame, assertNull, assertNull, assertEquals, assertEquals,  
assertTrue, assertTrue, fail, fail
```

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

```
countTestCases, run
```

Fields

done

```
public boolean done
```

result

```
public junit.framework.TestResult result
```

Constructors

XmlTests

```
public XmlTests(String name)
```

Creates a new CommonAPIXMLTests object.

Parameters:

name - the test name

Throws:

Exception - any

Methods

setUp

```
public void setUp()
```

Initialize every test

tearDown

```
public void tearDown()
```

suite

```
public static junit.framework.TestSuite suite()  
    throws Exception
```

Build the suite of test

Returns:

the test suite

Throws:

(continued from last page)

Exception - ...

test_xml_Queue_JndiName

```
public void test_xml_Queue_JndiName()  
    throws Exception
```

Test if the given JNDI name follows: System/System1/ApplicationType/Common/Application/1-2;1-2;ReferenceImplementation/Comp/name or System/System1/ApplicationType/Common/Application/1-2;1-2;ReferenceImplementation/Resource/name

Throws:

Exception

test_xml_GetManagedEntityTypes_ContainsACTIVATE

```
public void test_xml_GetManagedEntityTypes_ContainsACTIVATE()  
    throws Exception
```

getManagedEntityTypes, instantiated from TestBasics
verify that the JVT support at least one managed Entity type using getManagedEntityTypes

Success if the the return array is not null and not empty

test_xml_GetQueryTypes_IsNotNull

```
public void test_xml_GetQueryTypes_IsNotNull()  
    throws Exception
```

verify that JVT support at least one query type using getNamedQueryTypes.
getNamedQueryTypes instantiated from TestBasics

Success if the the return array is not null and not empty

test_xml_GetUpvTypes_IsNotNull

```
public void test_xml_GetUpvTypes_IsNotNull()  
    throws Exception
```

verify that JVT support at least one query type using getUpdateProcedureTypes.
getUpdateProcedureTypes instantiated from TestBasics

Success if the the return array is not null

test_xml_GetSupportedOperations_IsSubsetOfJVTSessionOption

```
public void test_xml_GetSupportedOperations_IsSubsetOfJVTSessionOption()  
    throws Exception
```

GetSupportedOperations instantiated from TestBasics.
just call getSupportedOptionalOperations

Success if no exceptioun is thrown.

test_Negative_xml_query_NullQuery

```
public void test_Negative_xml_query_NullQuery()  
    throws Exception
```

Query managed entities using null query type
Success if query(null) throws OssIllegalArgumentException

(continued from last page)

Throws:

Exception - DOCUMENT ME!

[TCKException](#) - DOCUMENT ME!

test_Negative_xml_update_NullUpv

```
public void test_Negative_xml_update_NullUpv()  
    throws Exception
```

Update procedure using null type

Success if update(null) throws OssIllegalArgumentException

Throws:

Exception - DOCUMENT ME!

[TCKException](#) - DOCUMENT ME!

test_Positive_xml_getEventTypes

```
public void test_Positive_xml_getEventTypes()  
    throws Exception
```

Verify that the JVT support at least one event type.

For each Event type, verify also the corresponding EventPropertyDescriptor using the factory.

Success

if getEventTypes return no null and at least 1 event type and if (getEventDescriptor) for each type return a valid event propertyDescriptor (correct event type value) and if makeEvent(type) do not throw any exception and returnan instance of the desired event type

Throws:

Exception - DOCUMENT ME!

Index

A

addJvtSession 11
addXmlRootElement 8
assertTrue 17
assertTrueMsg 17
ATTRIBUTE 28

C

CommonApiTckTest 2
CommonAPIXMLTests 6
contains 48
containsAllKeys 20
containsDuplicates 49
countTestCases 73
createQueueReceiver 47
createQueueSender 46
createQueueSession 46

D

debug 14
deepCopy 49
destroyProcess 24
didRan 24
diff 47
doGeneric 64, 78, 99, 118, 125
done 5, 15, 131

E

EndlessEventTest 73
EVENT_TIMEOUT 10, 15
EVENTTIMEOUT 33

F

fail 18
find 48, 49
findImplementationOf 54
findInterfaceStartsWith 54
FindRemote 11

G

getAttributeType 45
getDocument 7
getElement 8
getEnvironment 17
getErrorStream 24
getEventTimeout 44
getIllegalKeys 53
getIllegalValue 51
getInitialContext 45
getInputStream 24
getInstanceOf 17
getManagedEntityKeys 53
getMaxAsObject 51
getMinAsObject 50
getOutputStream 24
getProperty 54
getQueueFactoryName 45
getRemote 11
getRequestQueueName 45
getTopicFactoryName 45
getTopicName 45
getTopicSubscriber 46
getTotalTimeout 45

H

howmany_key 15
howmany_value 16

I

init 17
intersect 47
INVALIDATION_PREFIX 28
isSameType 53
isSubset 47
isValidJndiName 46

J

JARNAME 32
JMS_FACTORY 15

JNDI_FACT_CLASS_NAME 10, 14
JNDI_URL 10, 14
JNDIFACTORY 32
JNDIURL 32
jvtSessionList 11, 15
JVTTestBase 16
JVTTOPICJNDINAME 33

L

log 16, 17, 44
LOG_FILE 10, 15
LOGFILE 32
LOGGING_ENABLED 32

M

main 2, 11, 22
makeCSList 49
MAX_BULK_ELEMENTS 39
MAXSTRINGLENGTH 33

N

namingContext 10, 14
NEGATIVE_TEST_PREFIX 28
NOOFRUNS 39

O

outputFile 44

P

POSITIVE_TEST_PREFIX 28
printStackTrace 36

Q

QUEUEFACTORYJNDINAME 33
QUOTECHAR 15

R

RANDOM 44

randomDate 51
randomElement 53
randomManagedEntityValueAttributes 52
randomMultiValue 52
randomString 52
randomSubset 50
randomSubsetNotEmpty 50
randomValue 52
randomValueInstance 52
REMOVE_EVENT_DEFAULT 29
REQUESTQUEUEJNDINAME 33
RESPONSEQUEUEJNDINAME 33
result 6, 16, 131
return_failed_only 16
run 6, 16, 24, 26, 73
runonce 14

S

SafeProperties 20
SEQUENCES 39
setUp 6, 58, 64, 71, 77, 84, 92, 99, 107, 118, 125, 131
shuffle 50
Starter 21
startTck 22
STDOUT_ENABLED 10
stopAtFirstFailed 21
stopGentle 26
StreamRedirector 25
suite 3, 57, 64, 73, 77, 84, 91, 99, 118, 125, 131
SuiteStarter 24
SYSPROPERTYFILE 27
SysProps 34

T

TCKException 36
tearDown 6, 58, 64, 71, 77, 84, 92, 131
test_cbe_ClassNameAndTypeAndValueKeyValuePair 65
test_cbe_GetAttributeNames_ContainsKEY 67
test_cbe_GetAttributeNames_NamesStartWithLowerCase 65
test_cbe_GetOptionalAttributeNames_InAllAttributes 65
test_cbe_GetSettableAttributeNames_NotContainsKEY 67
test_cbe_GetSettableAttributeNames_SubsetOfGetAttributeNames 66

- test_cbe_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName 68
- test_cbe_SetAttributeValue_IsPopulated 68
- test_cbe_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY 67
- test_cbe_SetManagedEntityKey_IsPopulatedKEY 67
- test_cbe_UnpopulateAllAttributes_GetAttributeValueFailsForAll 66
- test_cbe_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty 66
- test_cbe_UnpopulateAllAttributes_IsPopulatedFalseForAll 66
- test_ClassNameAndTypeAndValueKeyPair 99, 107
- test_event_GetAttributeNames_NamesStartWithLowerCase 78
- test_event_GetOptionalAttributeNames_InAllAttributes 78
- test_event_GetSettableAttributeNames_SubsetOfGetAttributeNames 78
- test_event_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName 80
- test_event_SetAttributeValue_IsPopulated 80
- test_event_UnpopulateAllAttributes_GetAttributeValueFailsForAll 79
- test_event_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty 79
- test_event_UnpopulateAllAttributes_IsPopulatedFalseForAll 79
- test_GetApplicationContext 92
- test_GetApplicationDN 92
- test_GetAttributeNames_ContainsKEY 102, 111
- test_GetAttributeNames_NamesStartWithLowerCase 99, 108
- test_GetInitialContext 58
- test_GetJVTSession 58
- test_GetLegalKeys 59
- test_GetManagedEntityTypes_ContainsACTIVATE 85
- test_GetOptionalAttributeNames_InAllAttributes 99, 108
- test_GetPrimaryKey 92
- test_GetQueryTypes_IsNotNull 85
- test_GetSettableAttributeNames_NotContainsKEY 102, 112
- test_GetSettableAttributeNames_SubsetOfGetAttributeNames 100, 109
- test_GetSupportedOperations_IsSubsetOfJVTSessionOption 85
- test_GetTopicSubscriber 58
- test_GetType 93
- test_GetUpvTypes_IsNotNull 85
- test_JVT_JNDI_NAME 84
- test_JVTSessionFound 6
- test_KeyCloneEquals 93
- test_MakePrimaryKey 93
- test_Negative_cbe_GetAttributeValue_AttributeNameIllegalValue 67
- test_Negative_cbe_GetAttributeValue_AttributeNameNullValue 67
- test_Negative_cbe_IsPopulated_AttributeNameIllegalValue 65
- test_Negative_cbe_IsPopulated_AttributeNameNullValue 65
- test_Negative_cbe_SetAttributeValue_AttributeNameIllegalValue 67
- test_Negative_cbe_SetAttributeValue_AttributeNameNullValue 66
- test_Negative_cbe_SetAttributeValue_IllegalObject 68
- test_Negative_cbe_SetAttributeValue_PrimitiveNullValue 68
- test_Negative_cbe_UnpopulateAttribute_AttributeNameIllegalValue 66
- test_Negative_cbe_UnpopulateAttribute_AttributeNameNullValue 66
- test_Negative_event_GetAttributeValue_AttributeNameIllegalValue 79
- test_Negative_event_GetAttributeValue_AttributeNameNullValue 79
- test_Negative_event_IsPopulated_AttributeNameIllegalValue 78
- test_Negative_event_IsPopulated_AttributeNameNullValue 78
- test_Negative_event_SetAttributeValue_AttributeNameIllegalValue 79
- test_Negative_event_SetAttributeValue_AttributeNameNullValue 79
- test_Negative_event_SetAttributeValue_IllegalObject 80
- test_Negative_event_SetAttributeValue_PrimitiveNullValue 80
- test_Negative_event_UnpopulateAttribute_AttributeNameIllegalValue 79
- test_Negative_event_UnpopulateAttribute_AttributeNameNullValue 78
- test_Negative_GetAttributeValue_AttributeNameIllegalValue 101, 111
- test_Negative_GetAttributeValue_AttributeNameNullValue 101, 110
- test_Negative_IsPopulated_AttributeNameIllegalValue 100, 109
- test_Negative_IsPopulated_AttributeNameNullValue 100, 109

- test_Negative_NewManagedEntityValue_IllegalValue 85
- test_Negative_NewManagedEntityValue_NullValue 85
- test_Negative_NewQueryValue_IllegalValue 87
- test_Negative_NewQueryValue_NullValue 86
- test_Negative_NewUpvValue_IllegalValue 86
- test_Negative_NewUpvValue_NullValue 86
- test_Negative_query_GetAttributeValue_AttributeNameIllegalValue 120
- test_Negative_query_GetAttributeValue_AttributeNameNullValue 120
- test_Negative_query_IsPopulated_AttributeNameIllegalValue 119
- test_Negative_query_IsPopulated_AttributeNameNullValue 119
- test_Negative_query_NullQuery 87
- test_Negative_query_SetAttributeValue_AttributeNameIllegalValue 120
- test_Negative_query_SetAttributeValue_AttributeNameNullValue 120
- test_Negative_query_SetAttributeValue_IllegalObject 121
- test_Negative_query_SetAttributeValue_PrimitiveNullValue 120
- test_Negative_query_UnpopulateAttribute_AttributeNameIllegalValue 119
- test_Negative_query_UnpopulateAttribute_AttributeNameNullValue 119
- test_Negative_SetAttributeValue_AttributeNameIllegalValue 101, 110
- test_Negative_SetAttributeValue_AttributeNameNullValue 101, 110
- test_Negative_SetAttributeValue_IllegalObject 103, 113
- test_Negative_SetAttributeValue_PrimitiveNullValue 102, 112
- test_Negative_UnpopulateAttribute_AttributeNameIllegalValue 100, 109
- test_Negative_UnpopulateAttribute_AttributeNameNullValue 100, 109
- test_Negative_update_NullUpv 87
- test_Negative_upv_GetAttributeValue_AttributeNameIllegalValue 127
- test_Negative_upv_GetAttributeValue_AttributeNameNullValue 127
- test_Negative_upv_IsPopulated_AttributeNameIllegalValue 126
- test_Negative_upv_IsPopulated_AttributeNameNullValue 126
- test_Negative_upv_SetAttributeValue_AttributeNameIllegalValue 127
- test_Negative_upv_SetAttributeValue_AttributeNameNullValue 127
- test_Negative_upv_SetAttributeValue_IllegalObject 128
- test_Negative_upv_SetAttributeValue_PrimitiveNullValue 127
- test_Negative_upv_UnpopulateAttribute_AttributeNameIllegalValue 126
- test_Negative_upv_UnpopulateAttribute_AttributeNameNullValue 126
- test_Negative_xml_query_NullQuery 132
- test_Negative_xml_update_NullUpv 133
- test_Positive_callMethodKeyCreateWithValue 87
- test_Positive_callMethodKeysCreateWithValues 87
- test_Positive_callMethodValuesMethWithKeys 88
- test_Positive_callMethodValueWithKey 88
- test_Positive_callMethodVoidWithKey 88
- test_Positive_callMethodVoidWithValue 88
- test_Positive_cbe_IsPopulated_AttributeNameAllDomainValues 65
- test_Positive_cbe_ManagedEntityKey_IllegalKey 68
- test_Positive_cbe_SetAttributeValue_DomainValue 68
- test_Positive_cbe_SetAttributeValue_PrimitiveMINMAXValues 68
- test_Positive_cbe_SetManagedEntityKey_DomainValue 67
- test_Positive_cbe_testCloneEqualsImplementation 68
- test_Positive_cbe_UnpopulateAttribute_AttributeNameAllDomainValues 66
- test_Positive_event_IsPopulated_AttributeNameAllDomainValues 78
- test_Positive_event_SetAttributeValue_DomainValue 80
- test_Positive_event_SetAttributeValue_PrimitiveMINMAXValues 80
- test_Positive_event_testCloneEqualsImplementation 80
- test_Positive_event_UnpopulateAttribute_AttributeNameAllDomainValues 78
- test_Positive_getCbeImplementation 64
- test_Positive_getCbeInterfaces 64
- test_Positive_getEventImplementation 77
- test_Positive_getEventTypes 88
- test_Positive_IsPopulated_AttributeNameAllDomainValues 99, 109
- test_Positive_ManagedEntityKey_IllegalKey 102, 112
- test_Positive_NewManagedEntityValue_AllDomainValues 85

-
- test_Positive_NewQueryValue_AllDomainValues 86
 - test_Positive_NewUpvValue_AllDomainValues 86
 - test_Positive_query_IsPopulated_AttributeNameAllDomainValues 118
 - test_Positive_query_SetAttributeValue_DomainValue 120
 - test_Positive_query_SetAttributeValue_PrimitiveMINMAXValues 120
 - test_Positive_query_testCloneEqualsImplementation 121
 - test_Positive_query_UnpopulateAttribute_AttributeNameAllDomainValues 119
 - test_Positive_SetAttributeValue_DomainValue 103, 112
 - test_Positive_SetAttributeValue_PrimitiveMINMAXValues 102, 112
 - test_Positive_SetAttributeValueKEY_IllegalKey 101, 111
 - test_Positive_SetManagedEntityKey_DomainValue 101, 111
 - test_Positive_SetManagedEntityKey_IllegalKey 101, 111
 - test_Positive_testCloneEqualsImplementation 103, 113
 - test_Positive_UnpopulateAttribute_AttributeNameAllDomainValues 100, 109
 - test_Positive_upv_IsPopulated_AttributeNameAllDomainValues 125
 - test_Positive_upv_SetAttributeValue_DomainValue 127
 - test_Positive_upv_SetAttributeValue_PrimitiveMINMAXValues 127
 - test_Positive_upv_testCloneEqualsImplementation 128
 - test_Positive_upv_UnpopulateAttribute_AttributeNameAllDomainValues 126
 - test_Positive_xml_getEventTypes 133
 - TEST_PREFIX 28
 - test_query_GetAttributeNames_NamesStartWithLowerCase 118
 - test_query_GetOptionalAttributeNames_InAllAttributes 118
 - test_query_Query_Type 118
 - test_query_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName 121
 - test_query_SetAttributeValue_IsPopulated 121
 - test_Query_Type 108
 - test_query_UnpopulateAllAttributes_GetAttributeValueFailsForAll 119
 - test_query_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty 119
 - test_query_UnpopulateAllAttributes_IsPopulatedFalseForAll 119
 - test_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName 103, 112
 - test_SetAttributeValue_IsPopulated 103, 113
 - test_SetMakeApplicationContext 92
 - test_SetManagedEntityKey_GetPopulatedAttributeNamesContainsKEY 102, 111
 - test_SetManagedEntityKey_IsPopulatedKEY 102, 111
 - test_SetPrimaryKeyIsNull 93
 - test_SetTypeIsNull 93
 - test_UnpopulateAllAttributes_GetAttributeValueFailsForAll 101, 110
 - test_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty 100, 110
 - test_UnpopulateAllAttributes_IsPopulatedFalseForAll 100, 110
 - test_upv_GetAttributeNames_NamesStartWithLowerCase 125
 - test_upv_GetOptionalAttributeNames_InAllAttributes 125
 - test_upv_SetAttributeValue_GetPopulatedAttributesNamesContainsAttributeName 128
 - test_upv_SetAttributeValue_IsPopulated 128
 - test_upv_Type 108, 125
 - test_upv_UnpopulateAllAttributes_GetAttributeValueFailsForAll 126
 - test_upv_UnpopulateAllAttributes_GetPopulatedAttributesNamesIsEmpty 126
 - test_upv_UnpopulateAllAttributes_IsPopulatedFalseForAll 126
 - test_xml_GetManagedEntityTypes_ContainsACTIVATE 132
 - test_xml_GetQueryTypes_IsNotNull 132
 - test_xml_GetSupportedOperations_IsSubsetOfJVTSessionOption 132
 - test_xml_GetUpvTypes_IsNotNull 132
 - test_xml_Queue_JndiName 132
 - test_XmlEvent 7
 - test_XmlManagedEntityKeyAndValue 7
 - test_XmlQueryValue 7
 - testAllEvents 71
 - TestBasics 57
 - TestCbeBasics 64
 - TestEvents 71
 - TestEventValue 77
 - TESTFILES_HOME 10, 15
 - TestJVTSessionSimple 84
 - TestManagedEntityKey 91
 - TestProps 39
 - TEXTILLEGALARGUMENT 28
 - TOPIC_NAME 15

TOPICFACTORYJNDINAME 33

toString 73

TOTALTIMEOUT 33

TYPE 28

U

UNKNOWN 28

Util 44

util 10, 15

W

WORDS 29

X

XMLSCHEMANAME 32

XmlTests 131

XVTTOPICJNDINAME 33