| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DirObjectFactory.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/naming/spi/DirectoryManager.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/spi/DirStateFactory.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/spi/DirObjectFactory.html)    [**NO FRAMES**](http://docs.google.com/DirObjectFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

## **javax.naming.spi**

Interface DirObjectFactory

**All Superinterfaces:** [ObjectFactory](http://docs.google.com/javax/naming/spi/ObjectFactory.html)

public interface **DirObjectFactory**extends [ObjectFactory](http://docs.google.com/javax/naming/spi/ObjectFactory.html)

This interface represents a factory for creating an object given an object and attributes about the object.

The JNDI framework allows for object implementations to be loaded in dynamically via *object factories*. See ObjectFactory for details.

A DirObjectFactory extends ObjectFactory by allowing an Attributes instance to be supplied to the getObjectInstance() method. DirObjectFactory implementations are intended to be used by DirContext service providers. The service provider, in addition reading an object from the directory, might already have attributes that are useful for the object factory to check to see whether the factory is supposed to process the object. For instance, an LDAP-style service provider might have read the "objectclass" of the object. A CORBA object factory might be interested only in LDAP entries with "objectclass=corbaObject". By using the attributes supplied by the LDAP service provider, the CORBA object factory can quickly eliminate objects that it need not worry about, and non-CORBA object factories can quickly eliminate CORBA-related LDAP entries.

**Since:** 1.3 **See Also:**[NamingManager.getObjectInstance(java.lang.Object, javax.naming.Name, javax.naming.Context, java.util.Hashtable)](http://docs.google.com/javax/naming/spi/NamingManager.html#getObjectInstance(java.lang.Object,%20javax.naming.Name,%20javax.naming.Context,%20java.util.Hashtable)), [DirectoryManager.getObjectInstance(java.lang.Object, javax.naming.Name, javax.naming.Context, java.util.Hashtable, javax.naming.directory.Attributes)](http://docs.google.com/javax/naming/spi/DirectoryManager.html#getObjectInstance(java.lang.Object,%20javax.naming.Name,%20javax.naming.Context,%20java.util.Hashtable,%20javax.naming.directory.Attributes)), [ObjectFactory](http://docs.google.com/javax/naming/spi/ObjectFactory.html)

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getObjectInstance**](http://docs.google.com/javax/naming/spi/DirObjectFactory.html#getObjectInstance(java.lang.Object,%20javax.naming.Name,%20javax.naming.Context,%20java.util.Hashtable,%20javax.naming.directory.Attributes))([Object](http://docs.google.com/java/lang/Object.html) obj, [Name](http://docs.google.com/javax/naming/Name.html) name, [Context](http://docs.google.com/javax/naming/Context.html) nameCtx, [Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> environment, [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) attrs)            Creates an object using the location or reference information, and attributes specified. |

| **Methods inherited from interface javax.naming.spi.**[**ObjectFactory**](http://docs.google.com/javax/naming/spi/ObjectFactory.html) |
| --- |
| [getObjectInstance](http://docs.google.com/javax/naming/spi/ObjectFactory.html#getObjectInstance(java.lang.Object,%20javax.naming.Name,%20javax.naming.Context,%20java.util.Hashtable)) |

| **Method Detail** |
| --- |

### getObjectInstance

[Object](http://docs.google.com/java/lang/Object.html) **getObjectInstance**([Object](http://docs.google.com/java/lang/Object.html) obj,  
 [Name](http://docs.google.com/javax/naming/Name.html) name,  
 [Context](http://docs.google.com/javax/naming/Context.html) nameCtx,  
 [Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> environment,  
 [Attributes](http://docs.google.com/javax/naming/directory/Attributes.html) attrs)  
 throws [Exception](http://docs.google.com/java/lang/Exception.html)

Creates an object using the location or reference information, and attributes specified.

Special requirements of this object are supplied using environment. An example of such an environment property is user identity information.

DirectoryManager.getObjectInstance() successively loads in object factories. If it encounters a DirObjectFactory, it will invoke DirObjectFactory.getObjectInstance(); otherwise, it invokes ObjectFactory.getObjectInstance(). It does this until a factory produces a non-null answer.

When an exception is thrown by an object factory, the exception is passed on to the caller of DirectoryManager.getObjectInstance(). The search for other factories that may produce a non-null answer is halted. An object factory should only throw an exception if it is sure that it is the only intended factory and that no other object factories should be tried. If this factory cannot create an object using the arguments supplied, it should return null.

Since DirObjectFactory extends ObjectFactory, it effectively has two getObjectInstance() methods, where one differs from the other by the attributes argument. Given a factory that implements DirObjectFactory, DirectoryManager.getObjectInstance() will only use the method that accepts the attributes argument, while NamingManager.getObjectInstance() will only use the one that does not accept the attributes argument.

See ObjectFactory for a description URL context factories and other properties of object factories that apply equally to DirObjectFactory.

The name, attrs, and environment parameters are owned by the caller. The implementation will not modify these objects or keep references to them, although it may keep references to clones or copies.

**Parameters:**obj - The possibly null object containing location or reference information that can be used in creating an object.name - The name of this object relative to nameCtx, or null if no name is specified.nameCtx - The context relative to which the name parameter is specified, or null if name is relative to the default initial context.environment - The possibly null environment that is used in creating the object.attrs - The possibly null attributes containing some of obj's attributes. attrs might not necessarily have all of obj's attributes. If the object factory requires more attributes, it needs to get it, either using obj, or name and nameCtx. The factory must not modify attrs. **Returns:**The object created; null if an object cannot be created. **Throws:** [Exception](http://docs.google.com/java/lang/Exception.html) - If this object factory encountered an exception while attempting to create an object, and no other object factories are to be tried.**See Also:**[DirectoryManager.getObjectInstance(java.lang.Object, javax.naming.Name, javax.naming.Context, java.util.Hashtable, javax.naming.directory.Attributes)](http://docs.google.com/javax/naming/spi/DirectoryManager.html#getObjectInstance(java.lang.Object,%20javax.naming.Name,%20javax.naming.Context,%20java.util.Hashtable,%20javax.naming.directory.Attributes)), [NamingManager.getURLContext(java.lang.String, java.util.Hashtable)](http://docs.google.com/javax/naming/spi/NamingManager.html#getURLContext(java.lang.String,%20java.util.Hashtable))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DirObjectFactory.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/naming/spi/DirectoryManager.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/spi/DirStateFactory.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/spi/DirObjectFactory.html)    [**NO FRAMES**](http://docs.google.com/DirObjectFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).