| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ControlFactory.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/ldap/Control.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/ldap/ExtendedRequest.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/ldap/ControlFactory.html)    [**NO FRAMES**](http://docs.google.com/ControlFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **javax.naming.ldap**

Class ControlFactory

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.naming.ldap.ControlFactory**

public abstract class **ControlFactory**extends [Object](http://docs.google.com/java/lang/Object.html)

This abstract class represents a factory for creating LDAPv3 controls. LDAPv3 controls are defined in [RFC 2251](ftp://ftp.isi.edu/in-notes/rfc2251.txt).

When a service provider receives a response control, it uses control factories to return the specific/appropriate control class implementation.

**Since:** 1.3 **See Also:**[Control](http://docs.google.com/javax/naming/ldap/Control.html)

| **Constructor Summary** | |
| --- | --- |
| protected | [**ControlFactory**](http://docs.google.com/javax/naming/ldap/ControlFactory.html#ControlFactory())() |

| **Method Summary** | |
| --- | --- |
| abstract  [Control](http://docs.google.com/javax/naming/ldap/Control.html) | [**getControlInstance**](http://docs.google.com/javax/naming/ldap/ControlFactory.html#getControlInstance(javax.naming.ldap.Control))([Control](http://docs.google.com/javax/naming/ldap/Control.html) ctl)            Creates a control using this control factory. |
| static [Control](http://docs.google.com/javax/naming/ldap/Control.html) | [**getControlInstance**](http://docs.google.com/javax/naming/ldap/ControlFactory.html#getControlInstance(javax.naming.ldap.Control,%20javax.naming.Context,%20java.util.Hashtable))([Control](http://docs.google.com/javax/naming/ldap/Control.html) ctl, [Context](http://docs.google.com/javax/naming/Context.html) ctx, [Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> env)            Creates a control using known control factories. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### ControlFactory

protected **ControlFactory**()

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

### getControlInstance

public abstract [Control](http://docs.google.com/javax/naming/ldap/Control.html) **getControlInstance**([Control](http://docs.google.com/javax/naming/ldap/Control.html) ctl)  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Creates a control using this control factory.

The factory is used by the service provider to return controls that it reads from the LDAP protocol as specialized control classes. Without this mechanism, the provider would be returning controls that only contained data in BER encoded format.

Typically, ctl is a "basic" control containing BER encoded data. The factory is used to create a specialized control implementation, usually by decoding the BER encoded data, that provides methods to access that data in a type-safe and friendly manner.

For example, a factory might use the BER encoded data in basic control and return an instance of a VirtualListReplyControl.

If this factory cannot create a control using the argument supplied, it should return null. A factory should only throw an exception if it is sure that it is the only intended factory and that no other control factories should be tried. This might happen, for example, if the BER data in the control does not match what is expected of a control with the given OID. Since this method throws NamingException, any other internally generated exception that should be propagated must be wrapped inside a NamingException.

**Parameters:**ctl - A non-null control. **Returns:**A possibly null Control. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - If ctl contains invalid data that prevents it from being used to create a control. A factory should only throw an exception if it knows how to produce the control (identified by the OID) but is unable to because of, for example invalid BER data.

### getControlInstance

public static [Control](http://docs.google.com/javax/naming/ldap/Control.html) **getControlInstance**([Control](http://docs.google.com/javax/naming/ldap/Control.html) ctl,  
 [Context](http://docs.google.com/javax/naming/Context.html) ctx,  
 [Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> env)  
 throws [NamingException](http://docs.google.com/javax/naming/NamingException.html)

Creates a control using known control factories.

The following rule is used to create the control:

* Use the control factories specified in the LdapContext.CONTROL\_FACTORIES property of the environment, and of the provider resource file associated with ctx, in that order. The value of this property is a colon-separated list of factory class names that are tried in order, and the first one that succeeds in creating the control is the one used. If none of the factories can be loaded, return ctl. If an exception is encountered while creating the control, the exception is passed up to the caller.

Note that a control factory must be public and must have a public constructor that accepts no arguments.

**Parameters:**ctl - The non-null control object containing the OID and BER data.ctx - The possibly null context in which the control is being created. If null, no such information is available.env - The possibly null environment of the context. This is used to find the value of the LdapContext.CONTROL\_FACTORIES property. **Returns:**A control object created using ctl; or ctl if a control object cannot be created using the algorithm described above. **Throws:** [NamingException](http://docs.google.com/javax/naming/NamingException.html) - if a naming exception was encountered while attempting to create the control object. If one of the factories accessed throws an exception, it is propagated up to the caller. If an error was encountered while loading and instantiating the factory and object classes, the exception is wrapped inside a NamingException and then rethrown.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ControlFactory.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/ldap/Control.html)   [**NEXT CLASS**](http://docs.google.com/javax/naming/ldap/ExtendedRequest.html) | [**FRAMES**](http://docs.google.com/index.html?javax/naming/ldap/ControlFactory.html)    [**NO FRAMES**](http://docs.google.com/ControlFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).