| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ClassLoaderRepository.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   [**NEXT CLASS**](http://docs.google.com/javax/management/loading/DefaultLoaderRepository.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/loading/ClassLoaderRepository.html)    [**NO FRAMES**](http://docs.google.com/ClassLoaderRepository.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **javax.management.loading**

Interface ClassLoaderRepository

public interface **ClassLoaderRepository**

Instances of this interface are used to keep the list of ClassLoaders registered in an MBean Server. They provide the necessary methods to load classes using the registered ClassLoaders.

The first ClassLoader in a ClassLoaderRepository is always the MBean Server's own ClassLoader.

When an MBean is registered in an MBean Server, if it is of a subclass of [ClassLoader](http://docs.google.com/java/lang/ClassLoader.html) and if it does not implement the interface [PrivateClassLoader](http://docs.google.com/javax/management/loading/PrivateClassLoader.html), it is added to the end of the MBean Server's ClassLoaderRepository. If it is subsequently unregistered from the MBean Server, it is removed from the ClassLoaderRepository.

The order of MBeans in the ClassLoaderRepository is significant. For any two MBeans *X* and *Y* in the ClassLoaderRepository, *X* must appear before *Y* if the registration of *X* was completed before the registration of *Y* started. If *X* and *Y* were registered concurrently, their order is indeterminate. The registration of an MBean corresponds to the call to [MBeanServer.registerMBean(java.lang.Object, javax.management.ObjectName)](http://docs.google.com/javax/management/MBeanServer.html#registerMBean(java.lang.Object,%20javax.management.ObjectName)) or one of the [MBeanServer](http://docs.google.com/javax/management/MBeanServer.html).createMBean methods.

**Since:** 1.5 **See Also:**[MBeanServerFactory](http://docs.google.com/javax/management/MBeanServerFactory.html)

| **Method Summary** | |
| --- | --- |
| [Class](http://docs.google.com/java/lang/Class.html)<?> | [**loadClass**](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html#loadClass(java.lang.String))([String](http://docs.google.com/java/lang/String.html) className)            Load the given class name through the list of class loaders. |
| [Class](http://docs.google.com/java/lang/Class.html)<?> | [**loadClassBefore**](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html#loadClassBefore(java.lang.ClassLoader,%20java.lang.String))([ClassLoader](http://docs.google.com/java/lang/ClassLoader.html) stop, [String](http://docs.google.com/java/lang/String.html) className)            Load the given class name through the list of class loaders, stopping at the given one. |
| [Class](http://docs.google.com/java/lang/Class.html)<?> | [**loadClassWithout**](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html#loadClassWithout(java.lang.ClassLoader,%20java.lang.String))([ClassLoader](http://docs.google.com/java/lang/ClassLoader.html) exclude, [String](http://docs.google.com/java/lang/String.html) className)            Load the given class name through the list of class loaders, excluding the given one. |

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

### loadClass

[Class](http://docs.google.com/java/lang/Class.html)<?> **loadClass**([String](http://docs.google.com/java/lang/String.html) className)  
 throws [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html)

Load the given class name through the list of class loaders. Each ClassLoader in turn from the ClassLoaderRepository is asked to load the class via its [ClassLoader.loadClass(String)](http://docs.google.com/java/lang/ClassLoader.html#loadClass(java.lang.String)) method. If it successfully returns a [Class](http://docs.google.com/java/lang/Class.html) object, that is the result of this method. If it throws a [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html), the search continues with the next ClassLoader. If it throws another exception, the exception is propagated from this method. If the end of the list is reached, a [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html) is thrown.

**Parameters:**className - The name of the class to be loaded. **Returns:**the loaded class. **Throws:** [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html) - The specified class could not be found.

### loadClassWithout

[Class](http://docs.google.com/java/lang/Class.html)<?> **loadClassWithout**([ClassLoader](http://docs.google.com/java/lang/ClassLoader.html) exclude,  
 [String](http://docs.google.com/java/lang/String.html) className)  
 throws [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html)

Load the given class name through the list of class loaders, excluding the given one. Each ClassLoader in turn from the ClassLoaderRepository, except exclude, is asked to load the class via its [ClassLoader.loadClass(String)](http://docs.google.com/java/lang/ClassLoader.html#loadClass(java.lang.String)) method. If it successfully returns a [Class](http://docs.google.com/java/lang/Class.html) object, that is the result of this method. If it throws a [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html), the search continues with the next ClassLoader. If it throws another exception, the exception is propagated from this method. If the end of the list is reached, a [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html) is thrown.

Be aware that if a ClassLoader in the ClassLoaderRepository calls this method from its [loadClass](http://docs.google.com/java/lang/ClassLoader.html#loadClass(java.lang.String)) method, it exposes itself to a deadlock if another ClassLoader in the ClassLoaderRepository does the same thing at the same time. The [loadClassBefore(java.lang.ClassLoader, java.lang.String)](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html#loadClassBefore(java.lang.ClassLoader,%20java.lang.String)) method is recommended to avoid the risk of deadlock.

**Parameters:**className - The name of the class to be loaded.exclude - The class loader to be excluded. May be null, in which case this method is equivalent to [loadClass(className)](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html#loadClass(java.lang.String)). **Returns:**the loaded class. **Throws:** [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html) - The specified class could not be found.

### loadClassBefore

[Class](http://docs.google.com/java/lang/Class.html)<?> **loadClassBefore**([ClassLoader](http://docs.google.com/java/lang/ClassLoader.html) stop,  
 [String](http://docs.google.com/java/lang/String.html) className)  
 throws [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html)

Load the given class name through the list of class loaders, stopping at the given one. Each ClassLoader in turn from the ClassLoaderRepository is asked to load the class via its [ClassLoader.loadClass(String)](http://docs.google.com/java/lang/ClassLoader.html#loadClass(java.lang.String)) method. If it successfully returns a [Class](http://docs.google.com/java/lang/Class.html) object, that is the result of this method. If it throws a [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html), the search continues with the next ClassLoader. If it throws another exception, the exception is propagated from this method. If the search reaches stop or the end of the list, a [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html) is thrown.

Typically this method is called from the [loadClass](http://docs.google.com/java/lang/ClassLoader.html#loadClass(java.lang.String)) method of stop, to consult loaders that appear before it in the ClassLoaderRepository. By stopping the search as soon as stop is reached, a potential deadlock with concurrent class loading is avoided.

**Parameters:**className - The name of the class to be loaded.stop - The class loader at which to stop. May be null, in which case this method is equivalent to [loadClass(className)](http://docs.google.com/javax/management/loading/ClassLoaderRepository.html#loadClass(java.lang.String)). **Returns:**the loaded class. **Throws:** [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html) - The specified class could not be found.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ClassLoaderRepository.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   [**NEXT CLASS**](http://docs.google.com/javax/management/loading/DefaultLoaderRepository.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/loading/ClassLoaderRepository.html)    [**NO FRAMES**](http://docs.google.com/ClassLoaderRepository.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[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).