| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JMXConnectorServer.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/management/remote/JMXConnectorProvider.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/remote/JMXConnectorServerFactory.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/remote/JMXConnectorServer.html)    [**NO FRAMES**](http://docs.google.com/JMXConnectorServer.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#26in1rg) | [METHOD](#1ksv4uv) |

## **javax.management.remote**

Class JMXConnectorServer

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.management.NotificationBroadcasterSupport](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html)  
 **javax.management.remote.JMXConnectorServer**

**All Implemented Interfaces:** [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html), [NotificationBroadcaster](http://docs.google.com/javax/management/NotificationBroadcaster.html), [NotificationEmitter](http://docs.google.com/javax/management/NotificationEmitter.html), [JMXAddressable](http://docs.google.com/javax/management/remote/JMXAddressable.html), [JMXConnectorServerMBean](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html) **Direct Known Subclasses:** [RMIConnectorServer](http://docs.google.com/javax/management/remote/rmi/RMIConnectorServer.html)

public abstract class **JMXConnectorServer**extends [NotificationBroadcasterSupport](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html)implements [JMXConnectorServerMBean](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html), [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html), [JMXAddressable](http://docs.google.com/javax/management/remote/JMXAddressable.html)

Superclass of every connector server. A connector server is attached to an MBean server. It listens for client connection requests and creates a connection for each one.

A connector server is associated with an MBean server either by registering it in that MBean server, or by passing the MBean server to its constructor.

A connector server is inactive when created. It only starts listening for client connections when the [start](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#start()) method is called. A connector server stops listening for client connections when the [stop](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#stop()) method is called or when the connector server is unregistered from its MBean server.

Stopping a connector server does not unregister it from its MBean server. A connector server once stopped cannot be restarted.

Each time a client connection is made or broken, a notification of class [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html) is emitted.

**Since:** 1.5

| **Field Summary** | |
| --- | --- |
| static [String](http://docs.google.com/java/lang/String.html) | [**AUTHENTICATOR**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#AUTHENTICATOR)            Name of the attribute that specifies the authenticator for a connector server. |

| **Constructor Summary** | |
| --- | --- |
| [**JMXConnectorServer**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#JMXConnectorServer())()            Constructs a connector server that will be registered as an MBean in the MBean server it is attached to. |
| [**JMXConnectorServer**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#JMXConnectorServer(javax.management.MBeanServer))([MBeanServer](http://docs.google.com/javax/management/MBeanServer.html) mbeanServer)            Constructs a connector server that is attached to the given MBean server. |

| **Method Summary** | |
| --- | --- |
| protected  void | [**connectionClosed**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#connectionClosed(java.lang.String,%20java.lang.String,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) connectionId, [String](http://docs.google.com/java/lang/String.html) message, [Object](http://docs.google.com/java/lang/Object.html) userData)            Called by a subclass when a client connection is closed normally. |
| protected  void | [**connectionFailed**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#connectionFailed(java.lang.String,%20java.lang.String,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) connectionId, [String](http://docs.google.com/java/lang/String.html) message, [Object](http://docs.google.com/java/lang/Object.html) userData)            Called by a subclass when a client connection fails. |
| protected  void | [**connectionOpened**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#connectionOpened(java.lang.String,%20java.lang.String,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) connectionId, [String](http://docs.google.com/java/lang/String.html) message, [Object](http://docs.google.com/java/lang/Object.html) userData)            Called by a subclass when a new client connection is opened. |
| [String](http://docs.google.com/java/lang/String.html)[] | [**getConnectionIds**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#getConnectionIds())()            The list of IDs for currently-open connections to this connector server. |
| [MBeanServer](http://docs.google.com/javax/management/MBeanServer.html) | [**getMBeanServer**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#getMBeanServer())()            Returns the MBean server that this connector server is attached to. |
| [MBeanNotificationInfo](http://docs.google.com/javax/management/MBeanNotificationInfo.html)[] | [**getNotificationInfo**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#getNotificationInfo())()            Returns an array indicating the notifications that this MBean sends. |
| void | [**postDeregister**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#postDeregister())()            Allows the MBean to perform any operations needed after having been unregistered in the MBean server. |
| void | [**postRegister**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#postRegister(java.lang.Boolean))([Boolean](http://docs.google.com/java/lang/Boolean.html) registrationDone)            Allows the MBean to perform any operations needed after having been registered in the MBean server or after the registration has failed. |
| void | [**preDeregister**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#preDeregister())()            Called by an MBean server when this connector server is unregistered from that MBean server. |
| [ObjectName](http://docs.google.com/javax/management/ObjectName.html) | [**preRegister**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#preRegister(javax.management.MBeanServer,%20javax.management.ObjectName))([MBeanServer](http://docs.google.com/javax/management/MBeanServer.html) mbs, [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)            Called by an MBean server when this connector server is registered in that MBean server. |
| void | [**setMBeanServerForwarder**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#setMBeanServerForwarder(javax.management.remote.MBeanServerForwarder))([MBeanServerForwarder](http://docs.google.com/javax/management/remote/MBeanServerForwarder.html) mbsf)            Adds an object that intercepts requests for the MBean server that arrive through this connector server. |
| [JMXConnector](http://docs.google.com/javax/management/remote/JMXConnector.html) | [**toJMXConnector**](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#toJMXConnector(java.util.Map))([Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),?> env)            Returns a client stub for this connector server. |

| **Methods inherited from class javax.management.**[**NotificationBroadcasterSupport**](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html) |
| --- |
| [addNotificationListener](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html#addNotificationListener(javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object)), [handleNotification](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html#handleNotification(javax.management.NotificationListener,%20javax.management.Notification,%20java.lang.Object)), [removeNotificationListener](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html#removeNotificationListener(javax.management.NotificationListener)), [removeNotificationListener](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html#removeNotificationListener(javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object)), [sendNotification](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html#sendNotification(javax.management.Notification)) |

| **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)) |

| **Methods inherited from interface javax.management.remote.**[**JMXConnectorServerMBean**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html) |
| --- |
| [getAddress](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getAddress()), [getAttributes](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getAttributes()), [isActive](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#isActive()), [start](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#start()), [stop](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#stop()) |

| **Methods inherited from interface javax.management.remote.**[**JMXAddressable**](http://docs.google.com/javax/management/remote/JMXAddressable.html) |
| --- |
| [getAddress](http://docs.google.com/javax/management/remote/JMXAddressable.html#getAddress()) |

| **Field Detail** |
| --- |

### AUTHENTICATOR

public static final [String](http://docs.google.com/java/lang/String.html) **AUTHENTICATOR**

Name of the attribute that specifies the authenticator for a connector server. The value associated with this attribute, if any, must be an object that implements the interface [JMXAuthenticator](http://docs.google.com/javax/management/remote/JMXAuthenticator.html).

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.management.remote.JMXConnectorServer.AUTHENTICATOR)

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

### JMXConnectorServer

public **JMXConnectorServer**()

Constructs a connector server that will be registered as an MBean in the MBean server it is attached to. This constructor is typically called by one of the createMBean methods when creating, within an MBean server, a connector server that makes it available remotely.

### JMXConnectorServer

public **JMXConnectorServer**([MBeanServer](http://docs.google.com/javax/management/MBeanServer.html) mbeanServer)

Constructs a connector server that is attached to the given MBean server. A connector server that is created in this way can be registered in a different MBean server.

**Parameters:**mbeanServer - the MBean server that this connector server is attached to. Null if this connector server will be attached to an MBean server by being registered in it.

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

### getMBeanServer

public [MBeanServer](http://docs.google.com/javax/management/MBeanServer.html) **getMBeanServer**()

Returns the MBean server that this connector server is attached to.

**Returns:**the MBean server that this connector server is attached to, or null if it is not yet attached to an MBean server.

### setMBeanServerForwarder

public void **setMBeanServerForwarder**([MBeanServerForwarder](http://docs.google.com/javax/management/remote/MBeanServerForwarder.html) mbsf)

**Description copied from interface:** [**JMXConnectorServerMBean**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#setMBeanServerForwarder(javax.management.remote.MBeanServerForwarder))

Adds an object that intercepts requests for the MBean server that arrive through this connector server. This object will be supplied as the MBeanServer for any new connection created by this connector server. Existing connections are unaffected.

If this connector server is already associated with an MBeanServer object, then that object is given to [mbsf.setMBeanServer](http://docs.google.com/javax/management/remote/MBeanServerForwarder.html#setMBeanServer(javax.management.MBeanServer)). If doing so produces an exception, this method throws the same exception without any other effect.

If this connector is not already associated with an MBeanServer object, or if the mbsf.setMBeanServer call just mentioned succeeds, then mbsf becomes this connector server's MBeanServer.

**Specified by:**[setMBeanServerForwarder](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#setMBeanServerForwarder(javax.management.remote.MBeanServerForwarder)) in interface [JMXConnectorServerMBean](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html) **Parameters:**mbsf - the new MBeanServerForwarder.

### getConnectionIds

public [String](http://docs.google.com/java/lang/String.html)[] **getConnectionIds**()

**Description copied from interface:** [**JMXConnectorServerMBean**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getConnectionIds())

The list of IDs for currently-open connections to this connector server.

**Specified by:**[getConnectionIds](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getConnectionIds()) in interface [JMXConnectorServerMBean](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html) **Returns:**a new string array containing the list of IDs. If there are no currently-open connections, this array will be empty.

### toJMXConnector

public [JMXConnector](http://docs.google.com/javax/management/remote/JMXConnector.html) **toJMXConnector**([Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),?> env)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Returns a client stub for this connector server. A client stub is a serializable object whose [connect](http://docs.google.com/javax/management/remote/JMXConnector.html#connect(java.util.Map)) method can be used to make one new connection to this connector server.

A given connector need not support the generation of client stubs. However, the connectors specified by the JMX Remote API do (JMXMP Connector and RMI Connector).

The default implementation of this method uses [JMXConnectorServerMBean.getAddress()](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getAddress()) and [JMXConnectorFactory](http://docs.google.com/javax/management/remote/JMXConnectorFactory.html) to generate the stub, with code equivalent to the following:

JMXServiceURL addr = [getAddress()](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getAddress());  
 return [JMXConnectorFactory.newJMXConnector(addr, env)](http://docs.google.com/javax/management/remote/JMXConnectorFactory.html#newJMXConnector(javax.management.remote.JMXServiceURL,%20java.util.Map));

A connector server for which this is inappropriate must override this method so that it either implements the appropriate logic or throws [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html).

**Specified by:**[toJMXConnector](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#toJMXConnector(java.util.Map)) in interface [JMXConnectorServerMBean](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html) **Parameters:**env - client connection parameters of the same sort that could be provided to [JMXConnector.connect(Map)](http://docs.google.com/javax/management/remote/JMXConnector.html#connect(java.util.Map)). Can be null, which is equivalent to an empty map. **Returns:**a client stub that can be used to make a new connection to this connector server. **Throws:** [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if this connector server does not support the generation of client stubs. [IllegalStateException](http://docs.google.com/java/lang/IllegalStateException.html) - if the JMXConnectorServer is not started (see [JMXConnectorServerMBean.isActive()](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#isActive())). [IOException](http://docs.google.com/java/io/IOException.html) - if a communications problem means that a stub cannot be created.

### getNotificationInfo

public [MBeanNotificationInfo](http://docs.google.com/javax/management/MBeanNotificationInfo.html)[] **getNotificationInfo**()

Returns an array indicating the notifications that this MBean sends. The implementation in JMXConnectorServer returns an array with one element, indicating that it can emit notifications of class [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html) with the types defined in that class. A subclass that can emit other notifications should return an array that contains this element plus descriptions of the other notifications.

**Specified by:**[getNotificationInfo](http://docs.google.com/javax/management/NotificationBroadcaster.html#getNotificationInfo()) in interface [NotificationBroadcaster](http://docs.google.com/javax/management/NotificationBroadcaster.html)**Overrides:**[getNotificationInfo](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html#getNotificationInfo()) in class [NotificationBroadcasterSupport](http://docs.google.com/javax/management/NotificationBroadcasterSupport.html) **Returns:**the array of possible notifications.

### connectionOpened

protected void **connectionOpened**([String](http://docs.google.com/java/lang/String.html) connectionId,  
 [String](http://docs.google.com/java/lang/String.html) message,  
 [Object](http://docs.google.com/java/lang/Object.html) userData)

Called by a subclass when a new client connection is opened. Adds connectionId to the list returned by [getConnectionIds()](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#getConnectionIds()), then emits a [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html) with type [JMXConnectionNotification.OPENED](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html#OPENED).

**Parameters:**connectionId - the ID of the new connection. This must be different from the ID of any connection previously opened by this connector server.message - the message for the emitted [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html). Can be null. See [Notification.getMessage()](http://docs.google.com/javax/management/Notification.html#getMessage()).userData - the userData for the emitted [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html). Can be null. See [Notification.getUserData()](http://docs.google.com/javax/management/Notification.html#getUserData()). **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if connectionId is null.

### connectionClosed

protected void **connectionClosed**([String](http://docs.google.com/java/lang/String.html) connectionId,  
 [String](http://docs.google.com/java/lang/String.html) message,  
 [Object](http://docs.google.com/java/lang/Object.html) userData)

Called by a subclass when a client connection is closed normally. Removes connectionId from the list returned by [getConnectionIds()](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#getConnectionIds()), then emits a [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html) with type [JMXConnectionNotification.CLOSED](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html#CLOSED).

**Parameters:**connectionId - the ID of the closed connection.message - the message for the emitted [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html). Can be null. See [Notification.getMessage()](http://docs.google.com/javax/management/Notification.html#getMessage()).userData - the userData for the emitted [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html). Can be null. See [Notification.getUserData()](http://docs.google.com/javax/management/Notification.html#getUserData()). **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if connectionId is null.

### connectionFailed

protected void **connectionFailed**([String](http://docs.google.com/java/lang/String.html) connectionId,  
 [String](http://docs.google.com/java/lang/String.html) message,  
 [Object](http://docs.google.com/java/lang/Object.html) userData)

Called by a subclass when a client connection fails. Removes connectionId from the list returned by [getConnectionIds()](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#getConnectionIds()), then emits a [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html) with type [JMXConnectionNotification.FAILED](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html#FAILED).

**Parameters:**connectionId - the ID of the failed connection.message - the message for the emitted [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html). Can be null. See [Notification.getMessage()](http://docs.google.com/javax/management/Notification.html#getMessage()).userData - the userData for the emitted [JMXConnectionNotification](http://docs.google.com/javax/management/remote/JMXConnectionNotification.html). Can be null. See [Notification.getUserData()](http://docs.google.com/javax/management/Notification.html#getUserData()). **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if connectionId is null.

### preRegister

public [ObjectName](http://docs.google.com/javax/management/ObjectName.html) **preRegister**([MBeanServer](http://docs.google.com/javax/management/MBeanServer.html) mbs,  
 [ObjectName](http://docs.google.com/javax/management/ObjectName.html) name)

Called by an MBean server when this connector server is registered in that MBean server. This connector server becomes attached to the MBean server and its [getMBeanServer()](http://docs.google.com/javax/management/remote/JMXConnectorServer.html#getMBeanServer()) method will return mbs.

If this connector server is already attached to an MBean server, this method has no effect. The MBean server it is attached to is not necessarily the one it is being registered in.

**Specified by:**[preRegister](http://docs.google.com/javax/management/MBeanRegistration.html#preRegister(javax.management.MBeanServer,%20javax.management.ObjectName)) in interface [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html) **Parameters:**mbs - the MBean server in which this connection server is being registered.name - The object name of the MBean. **Returns:**The name under which the MBean is to be registered. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if mbs or name is null.

### postRegister

public void **postRegister**([Boolean](http://docs.google.com/java/lang/Boolean.html) registrationDone)

**Description copied from interface:** [**MBeanRegistration**](http://docs.google.com/javax/management/MBeanRegistration.html#postRegister(java.lang.Boolean)) Allows the MBean to perform any operations needed after having been registered in the MBean server or after the registration has failed.

**Specified by:**[postRegister](http://docs.google.com/javax/management/MBeanRegistration.html#postRegister(java.lang.Boolean)) in interface [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html) **Parameters:**registrationDone - Indicates whether or not the MBean has been successfully registered in the MBean server. The value false means that the registration phase has failed.

### preDeregister

public void **preDeregister**()  
 throws [Exception](http://docs.google.com/java/lang/Exception.html)

Called by an MBean server when this connector server is unregistered from that MBean server. If this connector server was attached to that MBean server by being registered in it, and if the connector server is still active, then unregistering it will call the [stop](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#stop()) method. If the stop method throws an exception, the unregistration attempt will fail. It is recommended to call the stop method explicitly before unregistering the MBean.

**Specified by:**[preDeregister](http://docs.google.com/javax/management/MBeanRegistration.html#preDeregister()) in interface [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html) **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if thrown by the [stop](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#stop()) method. [Exception](http://docs.google.com/java/lang/Exception.html) - This exception will be caught by the MBean server and re-thrown as an [MBeanRegistrationException](http://docs.google.com/javax/management/MBeanRegistrationException.html).

### postDeregister

public void **postDeregister**()

**Description copied from interface:** [**MBeanRegistration**](http://docs.google.com/javax/management/MBeanRegistration.html#postDeregister()) Allows the MBean to perform any operations needed after having been unregistered in the MBean server.

**Specified by:**[postDeregister](http://docs.google.com/javax/management/MBeanRegistration.html#postDeregister()) in interface [MBeanRegistration](http://docs.google.com/javax/management/MBeanRegistration.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JMXConnectorServer.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/management/remote/JMXConnectorProvider.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/remote/JMXConnectorServerFactory.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/remote/JMXConnectorServer.html)    [**NO FRAMES**](http://docs.google.com/JMXConnectorServer.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#26in1rg) | [METHOD](#1ksv4uv) |

[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).