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

## **javax.management.remote**

Interface JMXConnectorServerMBean

**All Known Implementing Classes:** [JMXConnectorServer](http://docs.google.com/javax/management/remote/JMXConnectorServer.html), [RMIConnectorServer](http://docs.google.com/javax/management/remote/rmi/RMIConnectorServer.html)

public interface **JMXConnectorServerMBean**

MBean interface for connector servers. A JMX API connector server is attached to an MBean server, and establishes connections to that MBean server for remote clients.

A newly-created connector server is *inactive*, and does not yet listen for connections. Only when its [start](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#start()) method has been called does it start listening for connections.

**Since:** 1.5

| **Method Summary** | |
| --- | --- |
| [JMXServiceURL](http://docs.google.com/javax/management/remote/JMXServiceURL.html) | [**getAddress**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getAddress())()            The address of this connector server. |
| [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),?> | [**getAttributes**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getAttributes())()            The attributes for this connector server. |
| [String](http://docs.google.com/java/lang/String.html)[] | [**getConnectionIds**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#getConnectionIds())()            The list of IDs for currently-open connections to this connector server. |
| boolean | [**isActive**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#isActive())()            Determines whether the connector server is active. |
| void | [**setMBeanServerForwarder**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.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. |
| void | [**start**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#start())()            Activates the connector server, that is, starts listening for client connections. |
| void | [**stop**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#stop())()            Deactivates the connector server, that is, stops listening for client connections. |
| [JMXConnector](http://docs.google.com/javax/management/remote/JMXConnector.html) | [**toJMXConnector**](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.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. |

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

### start

void **start**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Activates the connector server, that is, starts listening for client connections. Calling this method when the connector server is already active has no effect. Calling this method when the connector server has been stopped will generate an [IOException](http://docs.google.com/java/io/IOException.html).

**Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if it is not possible to start listening or if the connector server has been stopped. [IllegalStateException](http://docs.google.com/java/lang/IllegalStateException.html) - if the connector server has not been attached to an MBean server.

### stop

void **stop**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Deactivates the connector server, that is, stops listening for client connections. Calling this method will also close all client connections that were made by this server. After this method returns, whether normally or with an exception, the connector server will not create any new client connections.

Once a connector server has been stopped, it cannot be started again.

Calling this method when the connector server has already been stopped has no effect. Calling this method when the connector server has not yet been started will disable the connector server object permanently.

If closing a client connection produces an exception, that exception is not thrown from this method. 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) is emitted from this MBean with the connection ID of the connection that could not be closed.

Closing a connector server is a potentially slow operation. For example, if a client machine with an open connection has crashed, the close operation might have to wait for a network protocol timeout. Callers that do not want to block in a close operation should do it in a separate thread.

**Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if the server cannot be closed cleanly. When this exception is thrown, the server has already attempted to close all client connections. All client connections are closed except possibly those that generated exceptions when the server attempted to close them.

### isActive

boolean **isActive**()

Determines whether the connector server is active. A connector server starts being active when its [start](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#start()) method returns successfully and remains active until either its [stop](http://docs.google.com/javax/management/remote/JMXConnectorServerMBean.html#stop()) method is called or the connector server fails.

**Returns:**true if the connector server is active.

### setMBeanServerForwarder

void **setMBeanServerForwarder**([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. 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.

**Parameters:**mbsf - the new MBeanServerForwarder. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the call to [mbsf.setMBeanServer](http://docs.google.com/javax/management/remote/MBeanServerForwarder.html#setMBeanServer(javax.management.MBeanServer)) fails with IllegalArgumentException. This includes the case where mbsf is null.

### getConnectionIds

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

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

**Returns:**a new string array containing the list of IDs. If there are no currently-open connections, this array will be empty.

### getAddress

[JMXServiceURL](http://docs.google.com/javax/management/remote/JMXServiceURL.html) **getAddress**()

The address of this connector server.

**Returns:**the address of this connector server, or null if it does not have one.

### getAttributes

[Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),?> **getAttributes**()

The attributes for this connector server.

**Returns:**a read-only map containing the attributes for this connector server. Attributes whose values are not serializable are omitted from this map. If there are no serializable attributes, the returned map is empty.

### toJMXConnector

[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).

**Parameters:**env - client connection parameters of the same sort that can 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 [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.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JMXConnectorServerMBean.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/JMXConnectorServerFactory.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/remote/JMXConnectorServerProvider.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/remote/JMXConnectorServerMBean.html)    [**NO FRAMES**](http://docs.google.com/JMXConnectorServerMBean.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).