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

## **javax.management.modelmbean**

Interface ModelMBeanNotificationBroadcaster

**All Superinterfaces:** [NotificationBroadcaster](http://docs.google.com/javax/management/NotificationBroadcaster.html) **All Known Subinterfaces:** [ModelMBean](http://docs.google.com/javax/management/modelmbean/ModelMBean.html) **All Known Implementing Classes:** [RequiredModelMBean](http://docs.google.com/javax/management/modelmbean/RequiredModelMBean.html)

public interface **ModelMBeanNotificationBroadcaster**extends [NotificationBroadcaster](http://docs.google.com/javax/management/NotificationBroadcaster.html)

This interface must be implemented by the ModelMBeans. An implementation of this interface must be shipped with every JMX Agent.

Java resources wishing to be manageable instantiate the ModelMBean using the MBeanServer's createMBean method. The resource then sets the ModelMBeanInfo (with Descriptors) for the ModelMBean instance. The attributes and operations exposed via the ModelMBeanInfo for the ModelMBean are accessible from MBeans, connectors/adaptors like other MBeans. Through the ModelMBeanInfo Descriptors, values and methods in the managed application can be defined and mapped to attributes and operations of the ModelMBean. This mapping can be defined during development in an XML formatted file or dynamically and programmatically at runtime.

Every ModelMBean which is instantiated in the MBeanServer becomes manageable: its attributes and operations become remotely accessible through the connectors/adaptors connected to that MBeanServer. A Java object cannot be registered in the MBeanServer unless it is a JMX compliant MBean. By instantiating a ModelMBean, resources are guaranteed that the MBean is valid.

MBeanException and RuntimeOperationsException must be thrown on every public method. This allows for wrapping exceptions from distributed communications (RMI, EJB, etc.). These exceptions do not have to be thrown by the implementation except in the scenarios described in the specification and javadoc.

**Since:** 1.5

| **Method Summary** | |
| --- | --- |
| void | [**addAttributeChangeNotificationListener**](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#addAttributeChangeNotificationListener(javax.management.NotificationListener,%20java.lang.String,%20java.lang.Object))([NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener, [String](http://docs.google.com/java/lang/String.html) attributeName, [Object](http://docs.google.com/java/lang/Object.html) handback)            Registers an object which implements the NotificationListener interface as a listener. |
| void | [**removeAttributeChangeNotificationListener**](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#removeAttributeChangeNotificationListener(javax.management.NotificationListener,%20java.lang.String))([NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener, [String](http://docs.google.com/java/lang/String.html) attributeName)            Removes a listener for attributeChangeNotifications from the RequiredModelMBean. |
| void | [**sendAttributeChangeNotification**](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#sendAttributeChangeNotification(javax.management.Attribute,%20javax.management.Attribute))([Attribute](http://docs.google.com/javax/management/Attribute.html) oldValue, [Attribute](http://docs.google.com/javax/management/Attribute.html) newValue)            Sends an attributeChangeNotification which contains the old value and new value for the attribute to the registered AttributeChangeNotification listeners on the ModelMBean. |
| void | [**sendAttributeChangeNotification**](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#sendAttributeChangeNotification(javax.management.AttributeChangeNotification))([AttributeChangeNotification](http://docs.google.com/javax/management/AttributeChangeNotification.html) notification)            Sends an attributeChangeNotification which is passed in to the registered attributeChangeNotification listeners on the ModelMBean. |
| void | [**sendNotification**](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#sendNotification(javax.management.Notification))([Notification](http://docs.google.com/javax/management/Notification.html) ntfyObj)            Sends a Notification which is passed in to the registered Notification listeners on the ModelMBean as a jmx.modelmbean.generic notification. |
| void | [**sendNotification**](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#sendNotification(java.lang.String))([String](http://docs.google.com/java/lang/String.html) ntfyText)            Sends a Notification which contains the text string that is passed in to the registered Notification listeners on the ModelMBean. |

| **Methods inherited from interface javax.management.**[**NotificationBroadcaster**](http://docs.google.com/javax/management/NotificationBroadcaster.html) |
| --- |
| [addNotificationListener](http://docs.google.com/javax/management/NotificationBroadcaster.html#addNotificationListener(javax.management.NotificationListener,%20javax.management.NotificationFilter,%20java.lang.Object)), [getNotificationInfo](http://docs.google.com/javax/management/NotificationBroadcaster.html#getNotificationInfo()), [removeNotificationListener](http://docs.google.com/javax/management/NotificationBroadcaster.html#removeNotificationListener(javax.management.NotificationListener)) |

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

### sendNotification

void **sendNotification**([Notification](http://docs.google.com/javax/management/Notification.html) ntfyObj)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Sends a Notification which is passed in to the registered Notification listeners on the ModelMBean as a jmx.modelmbean.generic notification.

**Parameters:**ntfyObj - The notification which is to be passed to the 'handleNotification' method of the listener object. **Throws:** [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps an IllegalArgumentException: The Notification object passed in parameter is null.

### sendNotification

void **sendNotification**([String](http://docs.google.com/java/lang/String.html) ntfyText)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Sends a Notification which contains the text string that is passed in to the registered Notification listeners on the ModelMBean.

**Parameters:**ntfyText - The text which is to be passed in the Notification to the 'handleNotification' method of the listener object. the constructed Notification will be: type "jmx.modelmbean.generic" source this ModelMBean instance sequence 1 **Throws:** [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps an IllegalArgumentException: The Notification text string passed in parameter is null.

### sendAttributeChangeNotification

void **sendAttributeChangeNotification**([AttributeChangeNotification](http://docs.google.com/javax/management/AttributeChangeNotification.html) notification)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Sends an attributeChangeNotification which is passed in to the registered attributeChangeNotification listeners on the ModelMBean.

**Parameters:**notification - The notification which is to be passed to the 'handleNotification' method of the listener object. **Throws:** [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps an IllegalArgumentException: The AttributeChangeNotification object passed in parameter is null.

### sendAttributeChangeNotification

void **sendAttributeChangeNotification**([Attribute](http://docs.google.com/javax/management/Attribute.html) oldValue,  
 [Attribute](http://docs.google.com/javax/management/Attribute.html) newValue)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html)

Sends an attributeChangeNotification which contains the old value and new value for the attribute to the registered AttributeChangeNotification listeners on the ModelMBean.

**Parameters:**oldValue - The original value for the AttributenewValue - The current value for the Attribute

The constructed attributeChangeNotification will be:  
 type "jmx.attribute.change"  
 source this ModelMBean instance  
 sequence 1  
 attributeName oldValue.getName()  
 attributeType oldValue's class  
 attributeOldValue oldValue.getValue()  
 attributeNewValue newValue.getValue()

**Throws:** [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps an IllegalArgumentException: An Attribute object passed in parameter is null or the names of the two Attribute objects in parameter are not the same.

### addAttributeChangeNotificationListener

void **addAttributeChangeNotificationListener**([NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener,  
 [String](http://docs.google.com/java/lang/String.html) attributeName,  
 [Object](http://docs.google.com/java/lang/Object.html) handback)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html),  
 [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html)

Registers an object which implements the NotificationListener interface as a listener. This object's 'handleNotification()' method will be invoked when any attributeChangeNotification is issued through or by the ModelMBean. This does not include other Notifications. They must be registered for independently. An AttributeChangeNotification will be generated for this attributeName.

**Parameters:**listener - The listener object which will handles notifications emitted by the registered MBean.attributeName - The name of the ModelMBean attribute for which to receive change notifications. If null, then all attribute changes will cause an attributeChangeNotification to be issued.handback - The context to be sent to the listener with the notification when a notification is emitted. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - The listener cannot be null. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps an IllegalArgumentException The attribute name passed in parameter does not exist.**See Also:**[removeAttributeChangeNotificationListener(javax.management.NotificationListener, java.lang.String)](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#removeAttributeChangeNotificationListener(javax.management.NotificationListener,%20java.lang.String))

### removeAttributeChangeNotificationListener

void **removeAttributeChangeNotificationListener**([NotificationListener](http://docs.google.com/javax/management/NotificationListener.html) listener,  
 [String](http://docs.google.com/java/lang/String.html) attributeName)  
 throws [MBeanException](http://docs.google.com/javax/management/MBeanException.html),  
 [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html),  
 [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html)

Removes a listener for attributeChangeNotifications from the RequiredModelMBean.

**Parameters:**listener - The listener name which was handling notifications emitted by the registered MBean. This method will remove all information related to this listener.attributeName - The attribute for which the listener no longer wants to receive attributeChangeNotifications. If null the listener will be removed for all attributeChangeNotifications. **Throws:** [ListenerNotFoundException](http://docs.google.com/javax/management/ListenerNotFoundException.html) - The listener is not registered in the MBean or is null. [MBeanException](http://docs.google.com/javax/management/MBeanException.html) - Wraps a distributed communication Exception. [RuntimeOperationsException](http://docs.google.com/javax/management/RuntimeOperationsException.html) - Wraps an IllegalArgumentException If the inAttributeName parameter does not correspond to an attribute name.**See Also:**[addAttributeChangeNotificationListener(javax.management.NotificationListener, java.lang.String, java.lang.Object)](http://docs.google.com/javax/management/modelmbean/ModelMBeanNotificationBroadcaster.html#addAttributeChangeNotificationListener(javax.management.NotificationListener,%20java.lang.String,%20java.lang.Object))

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