| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.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 PACKAGE**](http://docs.google.com/javax/management/monitor/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/javax/management/relation/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/openmbean/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

## Package javax.management.openmbean

Provides the open data types and Open MBean descriptor classes.

**See:**

[**Description**](#3znysh7)

| **Interface Summary** | |
| --- | --- |
| [**CompositeData**](http://docs.google.com/javax/management/openmbean/CompositeData.html) | The CompositeData interface specifies the behavior of a specific type of complex *open data* objects which represent *composite data* structures. |
| [**CompositeDataView**](http://docs.google.com/javax/management/openmbean/CompositeDataView.html) | A Java class can implement this interface to indicate how it is to be converted into a CompositeData by the MXBean framework. |
| [**OpenMBeanAttributeInfo**](http://docs.google.com/javax/management/openmbean/OpenMBeanAttributeInfo.html) | Describes an attribute of an open MBean. |
| [**OpenMBeanConstructorInfo**](http://docs.google.com/javax/management/openmbean/OpenMBeanConstructorInfo.html) | Describes a constructor of an Open MBean. |
| [**OpenMBeanInfo**](http://docs.google.com/javax/management/openmbean/OpenMBeanInfo.html) | Describes an Open MBean: an Open MBean is recognized as such if its [getMBeanInfo()](http://docs.google.com/javax/management/DynamicMBean.html#getMBeanInfo()) method returns an instance of a class which implements the [OpenMBeanInfo](http://docs.google.com/javax/management/openmbean/OpenMBeanInfo.html) interface, typically [OpenMBeanInfoSupport](http://docs.google.com/javax/management/openmbean/OpenMBeanInfoSupport.html). |
| [**OpenMBeanOperationInfo**](http://docs.google.com/javax/management/openmbean/OpenMBeanOperationInfo.html) | Describes an operation of an Open MBean. |
| [**OpenMBeanParameterInfo**](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfo.html) | Describes a parameter used in one or more operations or constructors of an open MBean. |
| [**TabularData**](http://docs.google.com/javax/management/openmbean/TabularData.html) | The TabularData interface specifies the behavior of a specific type of complex *open data* objects which represent *tabular data* structures. |

| **Class Summary** | |
| --- | --- |
| [**ArrayType<T>**](http://docs.google.com/javax/management/openmbean/ArrayType.html) | The ArrayType class is the *open type* class whose instances describe all *open data* values which are n-dimensional arrays of *open data* values. |
| [**CompositeDataInvocationHandler**](http://docs.google.com/javax/management/openmbean/CompositeDataInvocationHandler.html) | An [InvocationHandler](http://docs.google.com/java/lang/reflect/InvocationHandler.html) that forwards getter methods to a [CompositeData](http://docs.google.com/javax/management/openmbean/CompositeData.html). |
| [**CompositeDataSupport**](http://docs.google.com/javax/management/openmbean/CompositeDataSupport.html) | The CompositeDataSupport class is the *open data* class which implements the CompositeData interface. |
| [**CompositeType**](http://docs.google.com/javax/management/openmbean/CompositeType.html) | The CompositeType class is the *open type* class whose instances describe the types of [CompositeData](http://docs.google.com/javax/management/openmbean/CompositeData.html) values. |
| [**OpenMBeanAttributeInfoSupport**](http://docs.google.com/javax/management/openmbean/OpenMBeanAttributeInfoSupport.html) | Describes an attribute of an open MBean. |
| [**OpenMBeanConstructorInfoSupport**](http://docs.google.com/javax/management/openmbean/OpenMBeanConstructorInfoSupport.html) | Describes a constructor of an Open MBean. |
| [**OpenMBeanInfoSupport**](http://docs.google.com/javax/management/openmbean/OpenMBeanInfoSupport.html) | The OpenMBeanInfoSupport class describes the management information of an *open MBean*: it is a subclass of [MBeanInfo](http://docs.google.com/javax/management/MBeanInfo.html), and it implements the [OpenMBeanInfo](http://docs.google.com/javax/management/openmbean/OpenMBeanInfo.html) interface. |
| [**OpenMBeanOperationInfoSupport**](http://docs.google.com/javax/management/openmbean/OpenMBeanOperationInfoSupport.html) | Describes an operation of an Open MBean. |
| [**OpenMBeanParameterInfoSupport**](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfoSupport.html) | Describes a parameter used in one or more operations or constructors of an open MBean. |
| [**OpenType<T>**](http://docs.google.com/javax/management/openmbean/OpenType.html) | The OpenType class is the parent abstract class of all classes which describe the actual *open type* of open data values. |
| [**SimpleType<T>**](http://docs.google.com/javax/management/openmbean/SimpleType.html) | The SimpleType class is the *open type* class whose instances describe all *open data* values which are neither arrays, nor [CompositeData](http://docs.google.com/javax/management/openmbean/CompositeData.html) values, nor [TabularData](http://docs.google.com/javax/management/openmbean/TabularData.html) values. |
| [**TabularDataSupport**](http://docs.google.com/javax/management/openmbean/TabularDataSupport.html) | The TabularDataSupport class is the *open data* class which implements the TabularData and the Map interfaces, and which is internally based on a hash map data structure. |
| [**TabularType**](http://docs.google.com/javax/management/openmbean/TabularType.html) | The TabularType class is the  *open type* class whose instances describe the types of [TabularData](http://docs.google.com/javax/management/openmbean/TabularData.html) values. |

| **Exception Summary** | |
| --- | --- |
| [**InvalidKeyException**](http://docs.google.com/javax/management/openmbean/InvalidKeyException.html) | This runtime exception is thrown to indicate that a method parameter which was expected to be an item name of a *composite data* or a row index of a *tabular data* is not valid. |
| [**InvalidOpenTypeException**](http://docs.google.com/javax/management/openmbean/InvalidOpenTypeException.html) | This runtime exception is thrown to indicate that the *open type* of an *open data* value is not the one expected. |
| [**KeyAlreadyExistsException**](http://docs.google.com/javax/management/openmbean/KeyAlreadyExistsException.html) | This runtime exception is thrown to indicate that the index of a row to be added to a *tabular data* instance is already used to refer to another row in this *tabular data* instance. |
| [**OpenDataException**](http://docs.google.com/javax/management/openmbean/OpenDataException.html) | This checked exception is thrown when an *open type*, an *open data* or an *open MBean metadata info* instance could not be constructed because one or more validity constraints were not met. |

## Package javax.management.openmbean Description

Provides the open data types and Open MBean descriptor classes. An *Open MBean* is an MBean where the types of attributes and of operation parameters and return values are built using a small set of predefined Java classes. Open MBeans facilitate operation with remote management programs that do not necessarily have access to application-specific types, including non-Java programs.

Every MBean has an [MBeanInfo](http://docs.google.com/javax/management/MBeanInfo.html) with information about the MBean itself, and its attributes, operations, constructors, and notifications. In an Open MBean, this MBeanInfo implements the [OpenMBeanInfo](http://docs.google.com/javax/management/openmbean/OpenMBeanInfo.html) interface, usually by being an instance of [OpenMBeanInfoSupport](http://docs.google.com/javax/management/openmbean/OpenMBeanInfoSupport.html).

The attribute information returned by [MBeanInfo.getAttributes](http://docs.google.com/javax/management/MBeanInfo.html#getAttributes()) for an Open MBean is an array of objects implementing [OpenMBeanAttributeInfo](http://docs.google.com/javax/management/openmbean/OpenMBeanAttributeInfo.html), usually instances of [OpenMBeanAttributeInfoSupport](http://docs.google.com/javax/management/openmbean/OpenMBeanAttributeInfoSupport.html). In addition to the usual information about attributes, an OpenMBeanAttributeInfo specifies the [OpenType](http://docs.google.com/javax/management/openmbean/OpenType.html) of the attribute. The possible OpenType values are predefined, which is what ensures that remote managers will understand them.

Similar remarks apply to the parameter types of operations and constructors, and to the return types of operations.

There is a distinction between an attribute's Java language type, as returned by [getType()](http://docs.google.com/javax/management/MBeanAttributeInfo.html#getType()), and its OpenType, as returned by [getOpenType()](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfo.html#getOpenType()). For example, if the Java language type is java.lang.String, the OpenType will be [SimpleType.String](http://docs.google.com/javax/management/openmbean/SimpleType.html#STRING). If the Java language type is [CompositeData](http://docs.google.com/javax/management/openmbean/CompositeData.html), the OpenType will be a [CompositeType](http://docs.google.com/javax/management/openmbean/CompositeType.html) that describes the items in the CompositeData instances for the attribute.

## Default values and constraints

In Open MBeans, attributes and parameters can have default values and/or constraints associated with them in the OpenMBeanAttributeInfo or OpenMBeanParameterInfo. There are two ways to specify these constraints. Either the values are directly specified as parameters to one of the constructors of OpenMBeanAttributeInfoSupport or OpenMBeanParameterInfoSupport, for example [OpenMBeanParameterInfoSupport.OpenMBeanParameterInfoSupport( String, String, OpenType, Object, Object[])](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfoSupport.html#OpenMBeanParameterInfoSupport(java.lang.String,%20java.lang.String,%20javax.management.openmbean.OpenType,%20T,%20T%5B%5D)); or the values are specified in a [Descriptor](http://docs.google.com/javax/management/Descriptor.html) given as a parameter to one of the constructors.

When a Descriptor is used, the fields of interest are these:

* defaultValue defines the value returned by [getDefaultValue()](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfo.html#getDefaultValue());
* minValue defines the value returned by [getMinValue()](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfo.html#getMinValue());
* maxValue defines the value returned by [getMaxValue()](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfo.html#getMaxValue());
* legalValues defines the values returned by [getLegalValues()](http://docs.google.com/javax/management/openmbean/OpenMBeanParameterInfo.html#getLegalValues()).

For defaultValue, minValue, and maxValue, the associated value must either be of the Java type corresponding to openType, or be a string that can be converted into that type. The conversion uses the static method valueOf(String) if it finds one; otherwise a constructor with a single String parameter if it finds one; otherwise it fails.

For legalValues, the associated value must be either an array or a Set, and the elements of the array or set must be convertible as described for defaultValue etc.

The following conditions must be met for these fields:

* the values must be of the appropriate type, or be strings that can be converted to the appropriate type as explained above;
* if legalValues is present then neither minValue nor maxValue must be present;
* if defaultValue is present then it must satisfy the constraints defined by legalValues, minValue, or maxValue when any of these is also present;
* if minValue and maxValue are both present then minValue must not be greater than maxValue.

**Since:** 1.5 **See Also:** [Java SE 6 Platform documentation on JMX technology](http://docs.google.com/technotes/guides/jmx/), in particular the  [JMX Specification, version 1.4](http://docs.google.com/technotes/guides/jmx/JMX_1_4_specification.pdf)

| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.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 PACKAGE**](http://docs.google.com/javax/management/monitor/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/javax/management/relation/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/openmbean/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[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).