| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultBoundedRangeModel.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/swing/DebugGraphics.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/DefaultButtonModel.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/DefaultBoundedRangeModel.html)    [**NO FRAMES**](http://docs.google.com/DefaultBoundedRangeModel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#17dp8vu) | [METHOD](#lnxbz9) |

## **javax.swing**

Class DefaultBoundedRangeModel

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.swing.DefaultBoundedRangeModel**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html)

public class **DefaultBoundedRangeModel**extends [Object](http://docs.google.com/java/lang/Object.html)implements [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html), [Serializable](http://docs.google.com/java/io/Serializable.html)

A generic implementation of BoundedRangeModel.

**Warning:** Serialized objects of this class will not be compatible with future Swing releases. The current serialization support is appropriate for short term storage or RMI between applications running the same version of Swing. As of 1.4, support for long term storage of all JavaBeansTM has been added to the java.beans package. Please see [XMLEncoder](http://docs.google.com/java/beans/XMLEncoder.html).

**See Also:**[BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html)

| **Field Summary** | |
| --- | --- |
| protected  [ChangeEvent](http://docs.google.com/javax/swing/event/ChangeEvent.html) | [**changeEvent**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#changeEvent)            Only one ChangeEvent is needed per model instance since the event's only (read-only) state is the source property. |
| protected  [EventListenerList](http://docs.google.com/javax/swing/event/EventListenerList.html) | [**listenerList**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#listenerList)            The listeners waiting for model changes. |

| **Constructor Summary** | |
| --- | --- |
| [**DefaultBoundedRangeModel**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#DefaultBoundedRangeModel())()            Initializes all of the properties with default values. |
| [**DefaultBoundedRangeModel**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#DefaultBoundedRangeModel(int,%20int,%20int,%20int))(int value, int extent, int min, int max)            Initializes value, extent, minimum and maximum. |

| **Method Summary** | |
| --- | --- |
| void | [**addChangeListener**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#addChangeListener(javax.swing.event.ChangeListener))([ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) l)            Adds a ChangeListener. |
| protected  void | [**fireStateChanged**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#fireStateChanged())()            Runs each ChangeListener's stateChanged method. |
| [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html)[] | [**getChangeListeners**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getChangeListeners())()            Returns an array of all the change listeners registered on this DefaultBoundedRangeModel. |
| int | [**getExtent**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getExtent())()            Returns the model's extent. |
| | <T extends [EventListener](http://docs.google.com/java/util/EventListener.html)>  T[] | | --- | | [**getListeners**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getListeners(java.lang.Class))([Class](http://docs.google.com/java/lang/Class.html)<T> listenerType)            Returns an array of all the objects currently registered as *Foo*Listeners upon this model. |
| int | [**getMaximum**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getMaximum())()            Returns the model's maximum. |
| int | [**getMinimum**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getMinimum())()            Returns the model's minimum. |
| int | [**getValue**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getValue())()            Returns the model's current value. |
| boolean | [**getValueIsAdjusting**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getValueIsAdjusting())()            Returns true if the value is in the process of changing as a result of actions being taken by the user. |
| void | [**removeChangeListener**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#removeChangeListener(javax.swing.event.ChangeListener))([ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) l)            Removes a ChangeListener. |
| void | [**setExtent**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setExtent(int))(int n)            Sets the extent to *n* after ensuring that *n* is greater than or equal to zero and falls within the model's constraints: |
| void | [**setMaximum**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setMaximum(int))(int n)            Sets the maximum to *n* after ensuring that *n* that the other three properties obey the model's constraints: |
| void | [**setMinimum**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setMinimum(int))(int n)            Sets the minimum to *n* after ensuring that *n* that the other three properties obey the model's constraints: |
| void | [**setRangeProperties**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setRangeProperties(int,%20int,%20int,%20int,%20boolean))(int newValue, int newExtent, int newMin, int newMax, boolean adjusting)            Sets all of the BoundedRangeModel properties after forcing the arguments to obey the usual constraints: |
| void | [**setValue**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setValue(int))(int n)            Sets the current value of the model. |
| void | [**setValueIsAdjusting**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setValueIsAdjusting(boolean))(boolean b)            Sets the valueIsAdjusting property. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#toString())()            Returns a string that displays all of the BoundedRangeModel properties. |

| **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()), [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)) |

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

### changeEvent

protected transient [ChangeEvent](http://docs.google.com/javax/swing/event/ChangeEvent.html) **changeEvent**

Only one ChangeEvent is needed per model instance since the event's only (read-only) state is the source property. The source of events generated here is always "this".

### listenerList

protected [EventListenerList](http://docs.google.com/javax/swing/event/EventListenerList.html) **listenerList**

The listeners waiting for model changes.

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

### DefaultBoundedRangeModel

public **DefaultBoundedRangeModel**()

Initializes all of the properties with default values. Those values are:

* value = 0
* extent = 0
* minimum = 0
* maximum = 100
* adjusting = false

### DefaultBoundedRangeModel

public **DefaultBoundedRangeModel**(int value,  
 int extent,  
 int min,  
 int max)

Initializes value, extent, minimum and maximum. Adjusting is false. Throws an IllegalArgumentException if the following constraints aren't satisfied:

min <= value <= value+extent <= max

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

### getValue

public int **getValue**()

Returns the model's current value.

**Specified by:**[getValue](http://docs.google.com/javax/swing/BoundedRangeModel.html#getValue()) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Returns:**the model's current value**See Also:**[setValue(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setValue(int)), [BoundedRangeModel.getValue()](http://docs.google.com/javax/swing/BoundedRangeModel.html#getValue())

### getExtent

public int **getExtent**()

Returns the model's extent.

**Specified by:**[getExtent](http://docs.google.com/javax/swing/BoundedRangeModel.html#getExtent()) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Returns:**the model's extent**See Also:**[setExtent(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setExtent(int)), [BoundedRangeModel.getExtent()](http://docs.google.com/javax/swing/BoundedRangeModel.html#getExtent())

### getMinimum

public int **getMinimum**()

Returns the model's minimum.

**Specified by:**[getMinimum](http://docs.google.com/javax/swing/BoundedRangeModel.html#getMinimum()) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Returns:**the model's minimum**See Also:**[setMinimum(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setMinimum(int)), [BoundedRangeModel.getMinimum()](http://docs.google.com/javax/swing/BoundedRangeModel.html#getMinimum())

### getMaximum

public int **getMaximum**()

Returns the model's maximum.

**Specified by:**[getMaximum](http://docs.google.com/javax/swing/BoundedRangeModel.html#getMaximum()) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Returns:**the model's maximum**See Also:**[setMaximum(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setMaximum(int)), [BoundedRangeModel.getMaximum()](http://docs.google.com/javax/swing/BoundedRangeModel.html#getMaximum())

### setValue

public void **setValue**(int n)

Sets the current value of the model. For a slider, that determines where the knob appears. Ensures that the new value, *n* falls within the model's constraints:

minimum <= value <= value+extent <= maximum

**Specified by:**[setValue](http://docs.google.com/javax/swing/BoundedRangeModel.html#setValue(int)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**n - the model's new value**See Also:**[BoundedRangeModel.setValue(int)](http://docs.google.com/javax/swing/BoundedRangeModel.html#setValue(int))

### setExtent

public void **setExtent**(int n)

Sets the extent to *n* after ensuring that *n* is greater than or equal to zero and falls within the model's constraints:

minimum <= value <= value+extent <= maximum

**Specified by:**[setExtent](http://docs.google.com/javax/swing/BoundedRangeModel.html#setExtent(int)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**n - the model's new extent**See Also:**[BoundedRangeModel.setExtent(int)](http://docs.google.com/javax/swing/BoundedRangeModel.html#setExtent(int))

### setMinimum

public void **setMinimum**(int n)

Sets the minimum to *n* after ensuring that *n* that the other three properties obey the model's constraints:

minimum <= value <= value+extent <= maximum

**Specified by:**[setMinimum](http://docs.google.com/javax/swing/BoundedRangeModel.html#setMinimum(int)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**n - the model's new minimum**See Also:**[getMinimum()](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getMinimum()), [BoundedRangeModel.setMinimum(int)](http://docs.google.com/javax/swing/BoundedRangeModel.html#setMinimum(int))

### setMaximum

public void **setMaximum**(int n)

Sets the maximum to *n* after ensuring that *n* that the other three properties obey the model's constraints:

minimum <= value <= value+extent <= maximum

**Specified by:**[setMaximum](http://docs.google.com/javax/swing/BoundedRangeModel.html#setMaximum(int)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**n - the model's new maximum**See Also:**[BoundedRangeModel.setMaximum(int)](http://docs.google.com/javax/swing/BoundedRangeModel.html#setMaximum(int))

### setValueIsAdjusting

public void **setValueIsAdjusting**(boolean b)

Sets the valueIsAdjusting property.

**Specified by:**[setValueIsAdjusting](http://docs.google.com/javax/swing/BoundedRangeModel.html#setValueIsAdjusting(boolean)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**b - true if the upcoming changes to the value property are part of a series**See Also:**[getValueIsAdjusting()](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getValueIsAdjusting()), [setValue(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setValue(int)), [BoundedRangeModel.setValueIsAdjusting(boolean)](http://docs.google.com/javax/swing/BoundedRangeModel.html#setValueIsAdjusting(boolean))

### getValueIsAdjusting

public boolean **getValueIsAdjusting**()

Returns true if the value is in the process of changing as a result of actions being taken by the user.

**Specified by:**[getValueIsAdjusting](http://docs.google.com/javax/swing/BoundedRangeModel.html#getValueIsAdjusting()) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Returns:**the value of the valueIsAdjusting property**See Also:**[setValue(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setValue(int)), [BoundedRangeModel.getValueIsAdjusting()](http://docs.google.com/javax/swing/BoundedRangeModel.html#getValueIsAdjusting())

### setRangeProperties

public void **setRangeProperties**(int newValue,  
 int newExtent,  
 int newMin,  
 int newMax,  
 boolean adjusting)

Sets all of the BoundedRangeModel properties after forcing the arguments to obey the usual constraints:

minimum <= value <= value+extent <= maximum

At most, one ChangeEvent is generated.

**Specified by:**[setRangeProperties](http://docs.google.com/javax/swing/BoundedRangeModel.html#setRangeProperties(int,%20int,%20int,%20int,%20boolean)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**newValue - an int giving the current valuenewExtent - an int giving the amount by which the value can "jump"newMin - an int giving the minimum valuenewMax - an int giving the maximum valueadjusting - a boolean, true if a series of changes are in progress**See Also:**[BoundedRangeModel.setRangeProperties(int, int, int, int, boolean)](http://docs.google.com/javax/swing/BoundedRangeModel.html#setRangeProperties(int,%20int,%20int,%20int,%20boolean)), [setValue(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setValue(int)), [setExtent(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setExtent(int)), [setMinimum(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setMinimum(int)), [setMaximum(int)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setMaximum(int)), [setValueIsAdjusting(boolean)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setValueIsAdjusting(boolean))

### addChangeListener

public void **addChangeListener**([ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) l)

Adds a ChangeListener. The change listeners are run each time any one of the Bounded Range model properties changes.

**Specified by:**[addChangeListener](http://docs.google.com/javax/swing/BoundedRangeModel.html#addChangeListener(javax.swing.event.ChangeListener)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**l - the ChangeListener to add**See Also:**[removeChangeListener(javax.swing.event.ChangeListener)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#removeChangeListener(javax.swing.event.ChangeListener)), [BoundedRangeModel.addChangeListener(javax.swing.event.ChangeListener)](http://docs.google.com/javax/swing/BoundedRangeModel.html#addChangeListener(javax.swing.event.ChangeListener))

### removeChangeListener

public void **removeChangeListener**([ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) l)

Removes a ChangeListener.

**Specified by:**[removeChangeListener](http://docs.google.com/javax/swing/BoundedRangeModel.html#removeChangeListener(javax.swing.event.ChangeListener)) in interface [BoundedRangeModel](http://docs.google.com/javax/swing/BoundedRangeModel.html) **Parameters:**l - the ChangeListener to remove**See Also:**[addChangeListener(javax.swing.event.ChangeListener)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#addChangeListener(javax.swing.event.ChangeListener)), [BoundedRangeModel.removeChangeListener(javax.swing.event.ChangeListener)](http://docs.google.com/javax/swing/BoundedRangeModel.html#removeChangeListener(javax.swing.event.ChangeListener))

### getChangeListeners

public [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html)[] **getChangeListeners**()

Returns an array of all the change listeners registered on this DefaultBoundedRangeModel.

**Returns:**all of this model's ChangeListeners or an empty array if no change listeners are currently registered**Since:** 1.4 **See Also:**[addChangeListener(javax.swing.event.ChangeListener)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#addChangeListener(javax.swing.event.ChangeListener)), [removeChangeListener(javax.swing.event.ChangeListener)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#removeChangeListener(javax.swing.event.ChangeListener))

### fireStateChanged

protected void **fireStateChanged**()

Runs each ChangeListener's stateChanged method.

**See Also:**[setRangeProperties(int, int, int, int, boolean)](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#setRangeProperties(int,%20int,%20int,%20int,%20boolean)), [EventListenerList](http://docs.google.com/javax/swing/event/EventListenerList.html)

### toString

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

Returns a string that displays all of the BoundedRangeModel properties.

**Overrides:**[toString](http://docs.google.com/java/lang/Object.html#toString()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a string representation of the object.

### getListeners

public <T extends [EventListener](http://docs.google.com/java/util/EventListener.html)> T[] **getListeners**([Class](http://docs.google.com/java/lang/Class.html)<T> listenerType)

Returns an array of all the objects currently registered as *Foo*Listeners upon this model. *Foo*Listeners are registered using the add*Foo*Listener method.

You can specify the listenerType argument with a class literal, such as *Foo*Listener.class. For example, you can query a DefaultBoundedRangeModel instance m for its change listeners with the following code:

ChangeListener[] cls = (ChangeListener[])(m.getListeners(ChangeListener.class));

If no such listeners exist, this method returns an empty array.

**Parameters:**listenerType - the type of listeners requested; this parameter should specify an interface that descends from java.util.EventListener **Returns:**an array of all objects registered as *Foo*Listeners on this model, or an empty array if no such listeners have been added **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if listenerType doesn't specify a class or interface that implements java.util.EventListener**Since:** 1.3 **See Also:**[getChangeListeners()](http://docs.google.com/javax/swing/DefaultBoundedRangeModel.html#getChangeListeners())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultBoundedRangeModel.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/swing/DebugGraphics.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/DefaultButtonModel.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/DefaultBoundedRangeModel.html)    [**NO FRAMES**](http://docs.google.com/DefaultBoundedRangeModel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#17dp8vu) | [METHOD](#lnxbz9) |

[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).