| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MetalScrollBarUI.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/plaf/metal/MetalRootPaneUI.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollButton.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/metal/MetalScrollBarUI.html)    [**NO FRAMES**](http://docs.google.com/MetalScrollBarUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#2et92p0) | [FIELD](#tyjcwt) | [CONSTR](#4d34og8) | [METHOD](#2s8eyo1) | DETAIL: [FIELD](#lnxbz9) | [CONSTR](#1y810tw) | [METHOD](#2xcytpi) |

## **javax.swing.plaf.metal**

Class MetalScrollBarUI

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.swing.plaf.ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html)  
 [javax.swing.plaf.ScrollBarUI](http://docs.google.com/javax/swing/plaf/ScrollBarUI.html)  
 [javax.swing.plaf.basic.BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)  
 **javax.swing.plaf.metal.MetalScrollBarUI**

**All Implemented Interfaces:** [LayoutManager](http://docs.google.com/java/awt/LayoutManager.html), [SwingConstants](http://docs.google.com/javax/swing/SwingConstants.html)

public class **MetalScrollBarUI**extends [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

Implementation of ScrollBarUI for the Metal Look and Feel

| **Nested Class Summary** | |
| --- | --- |

| **Nested classes/interfaces inherited from class javax.swing.plaf.basic.**[**BasicScrollBarUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html) |
| --- |
| [BasicScrollBarUI.ArrowButtonListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.ArrowButtonListener.html), [BasicScrollBarUI.ModelListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.ModelListener.html), [BasicScrollBarUI.PropertyChangeHandler](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.PropertyChangeHandler.html), [BasicScrollBarUI.ScrollListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.ScrollListener.html), [BasicScrollBarUI.TrackListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.TrackListener.html) |

| **Field Summary** | |
| --- | --- |
| protected  javax.swing.plaf.metal.MetalBumps | [**bumps**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#bumps) |
| protected  [MetalScrollButton](http://docs.google.com/javax/swing/plaf/metal/MetalScrollButton.html) | [**decreaseButton**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#decreaseButton) |
| static [String](http://docs.google.com/java/lang/String.html) | [**FREE\_STANDING\_PROP**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#FREE_STANDING_PROP) |
| protected  [MetalScrollButton](http://docs.google.com/javax/swing/plaf/metal/MetalScrollButton.html) | [**increaseButton**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#increaseButton) |
| protected  boolean | [**isFreeStanding**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#isFreeStanding) |
| protected  int | [**scrollBarWidth**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#scrollBarWidth) |

| **Fields inherited from class javax.swing.plaf.basic.**[**BasicScrollBarUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html) |
| --- |
| [buttonListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#buttonListener), [decrButton](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#decrButton), [DECREASE\_HIGHLIGHT](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#DECREASE_HIGHLIGHT), [incrButton](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#incrButton), [INCREASE\_HIGHLIGHT](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#INCREASE_HIGHLIGHT), [isDragging](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#isDragging), [maximumThumbSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#maximumThumbSize), [minimumThumbSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#minimumThumbSize), [modelListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#modelListener), [NO\_HIGHLIGHT](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#NO_HIGHLIGHT), [propertyChangeListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#propertyChangeListener), [scrollbar](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#scrollbar), [scrollListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#scrollListener), [scrollTimer](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#scrollTimer), [thumbDarkShadowColor](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#thumbDarkShadowColor), [thumbLightShadowColor](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#thumbLightShadowColor), [thumbRect](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#thumbRect), [trackColor](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#trackColor), [trackHighlight](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#trackHighlight), [trackHighlightColor](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#trackHighlightColor), [trackListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#trackListener), [trackRect](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#trackRect) |

| **Fields inherited from interface javax.swing.**[**SwingConstants**](http://docs.google.com/javax/swing/SwingConstants.html) |
| --- |
| [BOTTOM](http://docs.google.com/javax/swing/SwingConstants.html#BOTTOM), [CENTER](http://docs.google.com/javax/swing/SwingConstants.html#CENTER), [EAST](http://docs.google.com/javax/swing/SwingConstants.html#EAST), [HORIZONTAL](http://docs.google.com/javax/swing/SwingConstants.html#HORIZONTAL), [LEADING](http://docs.google.com/javax/swing/SwingConstants.html#LEADING), [LEFT](http://docs.google.com/javax/swing/SwingConstants.html#LEFT), [NEXT](http://docs.google.com/javax/swing/SwingConstants.html#NEXT), [NORTH](http://docs.google.com/javax/swing/SwingConstants.html#NORTH), [NORTH\_EAST](http://docs.google.com/javax/swing/SwingConstants.html#NORTH_EAST), [NORTH\_WEST](http://docs.google.com/javax/swing/SwingConstants.html#NORTH_WEST), [PREVIOUS](http://docs.google.com/javax/swing/SwingConstants.html#PREVIOUS), [RIGHT](http://docs.google.com/javax/swing/SwingConstants.html#RIGHT), [SOUTH](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH), [SOUTH\_EAST](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH_EAST), [SOUTH\_WEST](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH_WEST), [TOP](http://docs.google.com/javax/swing/SwingConstants.html#TOP), [TRAILING](http://docs.google.com/javax/swing/SwingConstants.html#TRAILING), [VERTICAL](http://docs.google.com/javax/swing/SwingConstants.html#VERTICAL), [WEST](http://docs.google.com/javax/swing/SwingConstants.html#WEST) |

| **Constructor Summary** | |
| --- | --- |
| [**MetalScrollBarUI**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#MetalScrollBarUI())() |

| **Method Summary** | |
| --- | --- |
| protected  void | [**configureScrollBarColors**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#configureScrollBarColors())() |
| protected  [JButton](http://docs.google.com/javax/swing/JButton.html) | [**createDecreaseButton**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#createDecreaseButton(int))(int orientation)            Returns the view that represents the decrease view. |
| protected  [JButton](http://docs.google.com/javax/swing/JButton.html) | [**createIncreaseButton**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#createIncreaseButton(int))(int orientation)            Returns the view that represents the increase view. |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**createPropertyChangeListener**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#createPropertyChangeListener())() |
| static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) | [**createUI**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#createUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c) |
| protected  [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getMinimumThumbSize**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#getMinimumThumbSize())()            Return the smallest acceptable size for the thumb. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getPreferredSize**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#getPreferredSize(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            A vertical scrollbar's preferred width is the maximum of preferred widths of the (non null) increment/decrement buttons, and the minimum width of the thumb. |
| protected  void | [**installDefaults**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#installDefaults())() |
| protected  void | [**installListeners**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#installListeners())() |
| protected  void | [**paintThumb**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#paintThumb(java.awt.Graphics,%20javax.swing.JComponent,%20java.awt.Rectangle))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [JComponent](http://docs.google.com/javax/swing/JComponent.html) c, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) thumbBounds) |
| protected  void | [**paintTrack**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#paintTrack(java.awt.Graphics,%20javax.swing.JComponent,%20java.awt.Rectangle))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [JComponent](http://docs.google.com/javax/swing/JComponent.html) c, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) trackBounds) |
| protected  void | [**setThumbBounds**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollBarUI.html#setThumbBounds(int,%20int,%20int,%20int))(int x, int y, int width, int height)            This is overridden only to increase the invalid area. |

| **Methods inherited from class javax.swing.plaf.basic.**[**BasicScrollBarUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html) |
| --- |
| [addLayoutComponent](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#addLayoutComponent(java.lang.String,%20java.awt.Component)), [createArrowButtonListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#createArrowButtonListener()), [createModelListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#createModelListener()), [createScrollListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#createScrollListener()), [createTrackListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#createTrackListener()), [getMaximumSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getMaximumSize(javax.swing.JComponent)), [getMaximumThumbSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getMaximumThumbSize()), [getSupportsAbsolutePositioning](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getSupportsAbsolutePositioning()), [getThumbBounds](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getThumbBounds()), [getTrackBounds](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getTrackBounds()), [installComponents](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#installComponents()), [installKeyboardActions](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#installKeyboardActions()), [installUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#installUI(javax.swing.JComponent)), [isThumbRollover](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#isThumbRollover()), [layoutContainer](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#layoutContainer(java.awt.Container)), [layoutHScrollbar](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#layoutHScrollbar(javax.swing.JScrollBar)), [layoutVScrollbar](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#layoutVScrollbar(javax.swing.JScrollBar)), [minimumLayoutSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#minimumLayoutSize(java.awt.Container)), [paint](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#paint(java.awt.Graphics,%20javax.swing.JComponent)), [paintDecreaseHighlight](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#paintDecreaseHighlight(java.awt.Graphics)), [paintIncreaseHighlight](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#paintIncreaseHighlight(java.awt.Graphics)), [preferredLayoutSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#preferredLayoutSize(java.awt.Container)), [removeLayoutComponent](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#removeLayoutComponent(java.awt.Component)), [scrollByBlock](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#scrollByBlock(int)), [scrollByUnit](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#scrollByUnit(int)), [setThumbRollover](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#setThumbRollover(boolean)), [uninstallComponents](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#uninstallComponents()), [uninstallDefaults](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#uninstallDefaults()), [uninstallKeyboardActions](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#uninstallKeyboardActions()), [uninstallListeners](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#uninstallListeners()), [uninstallUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#uninstallUI(javax.swing.JComponent)) |

| **Methods inherited from class javax.swing.plaf.**[**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html) |
| --- |
| [contains](http://docs.google.com/javax/swing/plaf/ComponentUI.html#contains(javax.swing.JComponent,%20int,%20int)), [getAccessibleChild](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getAccessibleChild(javax.swing.JComponent,%20int)), [getAccessibleChildrenCount](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getAccessibleChildrenCount(javax.swing.JComponent)), [getBaseline](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaseline(javax.swing.JComponent,%20int,%20int)), [getBaselineResizeBehavior](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaselineResizeBehavior(javax.swing.JComponent)), [getMinimumSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMinimumSize(javax.swing.JComponent)), [update](http://docs.google.com/javax/swing/plaf/ComponentUI.html#update(java.awt.Graphics,%20javax.swing.JComponent)) |

| **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)) |

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

### bumps

protected javax.swing.plaf.metal.MetalBumps **bumps**

### increaseButton

protected [MetalScrollButton](http://docs.google.com/javax/swing/plaf/metal/MetalScrollButton.html) **increaseButton**

### decreaseButton

protected [MetalScrollButton](http://docs.google.com/javax/swing/plaf/metal/MetalScrollButton.html) **decreaseButton**

### scrollBarWidth

protected int **scrollBarWidth**

### FREE\_STANDING\_PROP

public static final [String](http://docs.google.com/java/lang/String.html) **FREE\_STANDING\_PROP**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.swing.plaf.metal.MetalScrollBarUI.FREE_STANDING_PROP)

### isFreeStanding

protected boolean **isFreeStanding**

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

### MetalScrollBarUI

public **MetalScrollBarUI**()

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

### createUI

public static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **createUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

### installDefaults

protected void **installDefaults**()

**Overrides:**[installDefaults](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#installDefaults()) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### installListeners

protected void **installListeners**()

**Overrides:**[installListeners](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#installListeners()) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### createPropertyChangeListener

protected [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) **createPropertyChangeListener**()

**Overrides:**[createPropertyChangeListener](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#createPropertyChangeListener()) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### configureScrollBarColors

protected void **configureScrollBarColors**()

**Overrides:**[configureScrollBarColors](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#configureScrollBarColors()) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### getPreferredSize

public [Dimension](http://docs.google.com/java/awt/Dimension.html) **getPreferredSize**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

**Description copied from class:** [**BasicScrollBarUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getPreferredSize(javax.swing.JComponent)) A vertical scrollbar's preferred width is the maximum of preferred widths of the (non null) increment/decrement buttons, and the minimum width of the thumb. The preferred height is the sum of the preferred heights of the same parts. The basis for the preferred size of a horizontal scrollbar is similar.

The preferredSize is only computed once, subsequent calls to this method just return a cached size.

**Overrides:**[getPreferredSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getPreferredSize(javax.swing.JComponent)) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html) **Parameters:**c - the JScrollBar that's delegating this method to us **Returns:**the preferred size of a Basic JScrollBar**See Also:**[BasicScrollBarUI.getMaximumSize(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getMaximumSize(javax.swing.JComponent)), [ComponentUI.getMinimumSize(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMinimumSize(javax.swing.JComponent))

### createDecreaseButton

protected [JButton](http://docs.google.com/javax/swing/JButton.html) **createDecreaseButton**(int orientation)

Returns the view that represents the decrease view.

**Overrides:**[createDecreaseButton](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#createDecreaseButton(int)) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### createIncreaseButton

protected [JButton](http://docs.google.com/javax/swing/JButton.html) **createIncreaseButton**(int orientation)

Returns the view that represents the increase view.

**Overrides:**[createIncreaseButton](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#createIncreaseButton(int)) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### paintTrack

protected void **paintTrack**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [JComponent](http://docs.google.com/javax/swing/JComponent.html) c,  
 [Rectangle](http://docs.google.com/java/awt/Rectangle.html) trackBounds)

**Overrides:**[paintTrack](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#paintTrack(java.awt.Graphics,%20javax.swing.JComponent,%20java.awt.Rectangle)) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### paintThumb

protected void **paintThumb**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [JComponent](http://docs.google.com/javax/swing/JComponent.html) c,  
 [Rectangle](http://docs.google.com/java/awt/Rectangle.html) thumbBounds)

**Overrides:**[paintThumb](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#paintThumb(java.awt.Graphics,%20javax.swing.JComponent,%20java.awt.Rectangle)) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html)

### getMinimumThumbSize

protected [Dimension](http://docs.google.com/java/awt/Dimension.html) **getMinimumThumbSize**()

**Description copied from class:** [**BasicScrollBarUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getMinimumThumbSize()) Return the smallest acceptable size for the thumb. If the scrollbar becomes so small that this size isn't available, the thumb will be hidden.

**Warning** : the value returned by this method should not be be modified, it's a shared static constant.

**Overrides:**[getMinimumThumbSize](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getMinimumThumbSize()) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html) **Returns:**The smallest acceptable size for the thumb.**See Also:**[BasicScrollBarUI.getMaximumThumbSize()](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getMaximumThumbSize())

### setThumbBounds

protected void **setThumbBounds**(int x,  
 int y,  
 int width,  
 int height)

This is overridden only to increase the invalid area. This ensures that the "Shadow" below the thumb is invalidated

**Overrides:**[setThumbBounds](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#setThumbBounds(int,%20int,%20int,%20int)) in class [BasicScrollBarUI](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html) **See Also:**[BasicScrollBarUI.getThumbBounds()](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#getThumbBounds())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MetalScrollBarUI.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/plaf/metal/MetalRootPaneUI.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollButton.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/metal/MetalScrollBarUI.html)    [**NO FRAMES**](http://docs.google.com/MetalScrollBarUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#2et92p0) | [FIELD](#tyjcwt) | [CONSTR](#4d34og8) | [METHOD](#2s8eyo1) | DETAIL: [FIELD](#lnxbz9) | [CONSTR](#1y810tw) | [METHOD](#2xcytpi) |

[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).