| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MetalSliderUI.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/MetalSeparatorUI.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.MetalPropertyListener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/metal/MetalSliderUI.html)    [**NO FRAMES**](http://docs.google.com/MetalSliderUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#tyjcwt) | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#1ci93xb) | [METHOD](#2bn6wsx) |

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

Class MetalSliderUI

[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.SliderUI](http://docs.google.com/javax/swing/plaf/SliderUI.html)  
 [javax.swing.plaf.basic.BasicSliderUI](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html)  
 **javax.swing.plaf.metal.MetalSliderUI**

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

A Java L&F implementation of SliderUI.

**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).

| **Nested Class Summary** | |
| --- | --- |
| protected  class | [**MetalSliderUI.MetalPropertyListener**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.MetalPropertyListener.html) |

| **Nested classes/interfaces inherited from class javax.swing.plaf.basic.**[**BasicSliderUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html) |
| --- |
| [BasicSliderUI.ActionScroller](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.ActionScroller.html), [BasicSliderUI.ChangeHandler](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.ChangeHandler.html), [BasicSliderUI.ComponentHandler](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.ComponentHandler.html), [BasicSliderUI.FocusHandler](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.FocusHandler.html), [BasicSliderUI.PropertyChangeHandler](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.PropertyChangeHandler.html), [BasicSliderUI.ScrollListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.ScrollListener.html), [BasicSliderUI.TrackListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.TrackListener.html) |

| **Field Summary** | |
| --- | --- |
| protected static [Color](http://docs.google.com/java/awt/Color.html) | [**darkShadowColor**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#darkShadowColor) |
| protected  boolean | [**filledSlider**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#filledSlider) |
| protected static [Color](http://docs.google.com/java/awt/Color.html) | [**highlightColor**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#highlightColor) |
| protected static [Icon](http://docs.google.com/javax/swing/Icon.html) | [**horizThumbIcon**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#horizThumbIcon)            A default horizontal thumb Icon. |
| protected  [String](http://docs.google.com/java/lang/String.html) | [**SLIDER\_FILL**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#SLIDER_FILL) |
| protected static [Color](http://docs.google.com/java/awt/Color.html) | [**thumbColor**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#thumbColor) |
| protected  int | [**TICK\_BUFFER**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#TICK_BUFFER) |
| protected static int | [**tickLength**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#tickLength) |
| protected static int | [**trackWidth**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#trackWidth) |
| protected static [Icon](http://docs.google.com/javax/swing/Icon.html) | [**vertThumbIcon**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#vertThumbIcon)            A default vertical thumb Icon. |

| **Fields inherited from class javax.swing.plaf.basic.**[**BasicSliderUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html) |
| --- |
| [changeListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#changeListener), [componentListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#componentListener), [contentRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#contentRect), [focusInsets](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#focusInsets), [focusListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#focusListener), [focusRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#focusRect), [insetCache](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#insetCache), [labelRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#labelRect), [leftToRightCache](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#leftToRightCache), [MAX\_SCROLL](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#MAX_SCROLL), [MIN\_SCROLL](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#MIN_SCROLL), [NEGATIVE\_SCROLL](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#NEGATIVE_SCROLL), [POSITIVE\_SCROLL](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#POSITIVE_SCROLL), [propertyChangeListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#propertyChangeListener), [scrollListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#scrollListener), [scrollTimer](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#scrollTimer), [slider](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#slider), [thumbRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#thumbRect), [tickRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#tickRect), [trackBuffer](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#trackBuffer), [trackListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#trackListener), [trackRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#trackRect) |

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

| **Method Summary** | |
| --- | --- |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**createPropertyChangeListener**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#createPropertyChangeListener(javax.swing.JSlider))([JSlider](http://docs.google.com/javax/swing/JSlider.html) slider) |
| static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) | [**createUI**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#createUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c) |
| protected  int | [**getThumbOverhang**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#getThumbOverhang())()            Returns the amount that the thumb goes past the slide bar. |
| protected  [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getThumbSize**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#getThumbSize())() |
| int | [**getTickLength**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#getTickLength())()            Gets the height of the tick area for horizontal sliders and the width of the tick area for vertical sliders. |
| protected  int | [**getTrackLength**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#getTrackLength())()            Returns the longer dimension of the slide bar. |
| protected  int | [**getTrackWidth**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#getTrackWidth())()            Returns the shorter dimension of the track. |
| void | [**installUI**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#installUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Configures the specified component appropriate for the look and feel. |
| void | [**paintFocus**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#paintFocus(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) g) |
| protected  void | [**paintMajorTickForHorizSlider**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#paintMajorTickForHorizSlider(java.awt.Graphics,%20java.awt.Rectangle,%20int))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds, int x) |
| protected  void | [**paintMajorTickForVertSlider**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#paintMajorTickForVertSlider(java.awt.Graphics,%20java.awt.Rectangle,%20int))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds, int y) |
| protected  void | [**paintMinorTickForHorizSlider**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#paintMinorTickForHorizSlider(java.awt.Graphics,%20java.awt.Rectangle,%20int))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds, int x) |
| protected  void | [**paintMinorTickForVertSlider**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#paintMinorTickForVertSlider(java.awt.Graphics,%20java.awt.Rectangle,%20int))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds, int y) |
| void | [**paintThumb**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#paintThumb(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) g) |
| void | [**paintTrack**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#paintTrack(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) g) |
| protected  void | [**scrollDueToClickInTrack**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.html#scrollDueToClickInTrack(int))(int dir)            This function is called when a mousePressed was detected in the track, not in the thumb. |

| **Methods inherited from class javax.swing.plaf.basic.**[**BasicSliderUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html) |
| --- |
| [calculateContentRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateContentRect()), [calculateFocusRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateFocusRect()), [calculateGeometry](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateGeometry()), [calculateLabelRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateLabelRect()), [calculateThumbLocation](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateThumbLocation()), [calculateThumbSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateThumbSize()), [calculateTickRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateTickRect()), [calculateTrackBuffer](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateTrackBuffer()), [calculateTrackRect](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#calculateTrackRect()), [createChangeListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#createChangeListener(javax.swing.JSlider)), [createComponentListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#createComponentListener(javax.swing.JSlider)), [createFocusListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#createFocusListener(javax.swing.JSlider)), [createScrollListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#createScrollListener(javax.swing.JSlider)), [createTrackListener](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#createTrackListener(javax.swing.JSlider)), [drawInverted](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#drawInverted()), [getBaseline](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getBaseline(javax.swing.JComponent,%20int,%20int)), [getBaselineResizeBehavior](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getBaselineResizeBehavior(javax.swing.JComponent)), [getFocusColor](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getFocusColor()), [getHeightOfHighValueLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getHeightOfHighValueLabel()), [getHeightOfLowValueLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getHeightOfLowValueLabel()), [getHeightOfTallestLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getHeightOfTallestLabel()), [getHighestValue](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getHighestValue()), [getHighestValueLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getHighestValueLabel()), [getHighlightColor](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getHighlightColor()), [getLowestValue](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getLowestValue()), [getLowestValueLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getLowestValueLabel()), [getMaximumSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getMaximumSize(javax.swing.JComponent)), [getMinimumHorizontalSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getMinimumHorizontalSize()), [getMinimumSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getMinimumSize(javax.swing.JComponent)), [getMinimumVerticalSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getMinimumVerticalSize()), [getPreferredHorizontalSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getPreferredHorizontalSize()), [getPreferredSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getPreferredSize(javax.swing.JComponent)), [getPreferredVerticalSize](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getPreferredVerticalSize()), [getShadowColor](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getShadowColor()), [getWidthOfHighValueLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getWidthOfHighValueLabel()), [getWidthOfLowValueLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getWidthOfLowValueLabel()), [getWidthOfWidestLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#getWidthOfWidestLabel()), [installDefaults](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#installDefaults(javax.swing.JSlider)), [installKeyboardActions](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#installKeyboardActions(javax.swing.JSlider)), [installListeners](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#installListeners(javax.swing.JSlider)), [isDragging](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#isDragging()), [labelsHaveSameBaselines](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#labelsHaveSameBaselines()), [paint](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paint(java.awt.Graphics,%20javax.swing.JComponent)), [paintHorizontalLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paintHorizontalLabel(java.awt.Graphics,%20int,%20java.awt.Component)), [paintLabels](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paintLabels(java.awt.Graphics)), [paintTicks](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paintTicks(java.awt.Graphics)), [paintVerticalLabel](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paintVerticalLabel(java.awt.Graphics,%20int,%20java.awt.Component)), [recalculateIfInsetsChanged](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#recalculateIfInsetsChanged()), [recalculateIfOrientationChanged](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#recalculateIfOrientationChanged()), [scrollByBlock](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#scrollByBlock(int)), [scrollByUnit](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#scrollByUnit(int)), [setThumbLocation](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#setThumbLocation(int,%20int)), [uninstallKeyboardActions](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#uninstallKeyboardActions(javax.swing.JSlider)), [uninstallListeners](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#uninstallListeners(javax.swing.JSlider)), [uninstallUI](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#uninstallUI(javax.swing.JComponent)), [valueForXPosition](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#valueForXPosition(int)), [valueForYPosition](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#valueForYPosition(int)), [xPositionForValue](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#xPositionForValue(int)), [yPositionForValue](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#yPositionForValue(int)), [yPositionForValue](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#yPositionForValue(int,%20int,%20int)) |

| **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)), [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** |
| --- |

### TICK\_BUFFER

protected final int **TICK\_BUFFER**

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

### filledSlider

protected boolean **filledSlider**

### thumbColor

protected static [Color](http://docs.google.com/java/awt/Color.html) **thumbColor**

### highlightColor

protected static [Color](http://docs.google.com/java/awt/Color.html) **highlightColor**

### darkShadowColor

protected static [Color](http://docs.google.com/java/awt/Color.html) **darkShadowColor**

### trackWidth

protected static int **trackWidth**

### tickLength

protected static int **tickLength**

### horizThumbIcon

protected static [Icon](http://docs.google.com/javax/swing/Icon.html) **horizThumbIcon**

A default horizontal thumb Icon. This field might not be used. To change the Icon used by this delgate directly set it using the Slider.horizontalThumbIcon UIManager property.

### vertThumbIcon

protected static [Icon](http://docs.google.com/javax/swing/Icon.html) **vertThumbIcon**

A default vertical thumb Icon. This field might not be used. To change the Icon used by this delgate directly set it using the Slider.verticalThumbIcon UIManager property.

### SLIDER\_FILL

protected final [String](http://docs.google.com/java/lang/String.html) **SLIDER\_FILL**

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

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

### MetalSliderUI

public **MetalSliderUI**()

| **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)

### installUI

public void **installUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

**Description copied from class:** [**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html#installUI(javax.swing.JComponent)) Configures the specified component appropriate for the look and feel. This method is invoked when the ComponentUI instance is being installed as the UI delegate on the specified component. This method should completely configure the component for the look and feel, including the following:

1. Install any default property values for color, fonts, borders, icons, opacity, etc. on the component. Whenever possible, property values initialized by the client program should *not* be overridden.
2. Install a LayoutManager on the component if necessary.
3. Create/add any required sub-components to the component.
4. Create/install event listeners on the component.
5. Create/install a PropertyChangeListener on the component in order to detect and respond to component property changes appropriately.
6. Install keyboard UI (mnemonics, traversal, etc.) on the component.
7. Initialize any appropriate instance data.

**Overrides:**[installUI](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#installUI(javax.swing.JComponent)) in class [BasicSliderUI](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html) **Parameters:**c - the component where this UI delegate is being installed**See Also:**[ComponentUI.uninstallUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#uninstallUI(javax.swing.JComponent)), [JComponent.setUI(javax.swing.plaf.ComponentUI)](http://docs.google.com/javax/swing/JComponent.html#setUI(javax.swing.plaf.ComponentUI)), [JComponent.updateUI()](http://docs.google.com/javax/swing/JComponent.html#updateUI())

### createPropertyChangeListener

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

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

### paintThumb

public void **paintThumb**([Graphics](http://docs.google.com/java/awt/Graphics.html) g)

**Overrides:**[paintThumb](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paintThumb(java.awt.Graphics)) in class [BasicSliderUI](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html)

### paintTrack

public void **paintTrack**([Graphics](http://docs.google.com/java/awt/Graphics.html) g)

**Overrides:**[paintTrack](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paintTrack(java.awt.Graphics)) in class [BasicSliderUI](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html)

### paintFocus

public void **paintFocus**([Graphics](http://docs.google.com/java/awt/Graphics.html) g)

**Overrides:**[paintFocus](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#paintFocus(java.awt.Graphics)) in class [BasicSliderUI](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html)

### getThumbSize

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

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

### getTickLength

public int **getTickLength**()

Gets the height of the tick area for horizontal sliders and the width of the tick area for vertical sliders. BasicSliderUI uses the returned value to determine the tick area rectangle.

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

### getTrackWidth

protected int **getTrackWidth**()

Returns the shorter dimension of the track.

### getTrackLength

protected int **getTrackLength**()

Returns the longer dimension of the slide bar. (The slide bar is only the part that runs directly under the thumb)

### getThumbOverhang

protected int **getThumbOverhang**()

Returns the amount that the thumb goes past the slide bar.

### scrollDueToClickInTrack

protected void **scrollDueToClickInTrack**(int dir)

**Description copied from class:** [**BasicSliderUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.html#scrollDueToClickInTrack(int)) This function is called when a mousePressed was detected in the track, not in the thumb. The default behavior is to scroll by block. You can override this method to stop it from scrolling or to add additional behavior.

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

### paintMinorTickForHorizSlider

protected void **paintMinorTickForHorizSlider**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds,  
 int x)

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

### paintMajorTickForHorizSlider

protected void **paintMajorTickForHorizSlider**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds,  
 int x)

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

### paintMinorTickForVertSlider

protected void **paintMinorTickForVertSlider**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds,  
 int y)

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

### paintMajorTickForVertSlider

protected void **paintMajorTickForVertSlider**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [Rectangle](http://docs.google.com/java/awt/Rectangle.html) tickBounds,  
 int y)

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

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MetalSliderUI.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/MetalSeparatorUI.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/metal/MetalSliderUI.MetalPropertyListener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/metal/MetalSliderUI.html)    [**NO FRAMES**](http://docs.google.com/MetalSliderUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#tyjcwt) | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#1ci93xb) | [METHOD](#2bn6wsx) |

[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).