| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NavigationFilter.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/text/MutableAttributeSet.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/NavigationFilter.FilterBypass.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/NavigationFilter.html)    [**NO FRAMES**](http://docs.google.com/NavigationFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

## **javax.swing.text**

Class NavigationFilter

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

public class **NavigationFilter**extends [Object](http://docs.google.com/java/lang/Object.html)

NavigationFilter can be used to restrict where the cursor can be positioned. When the default cursor positioning actions attempt to reposition the cursor they will call into the NavigationFilter, assuming the JTextComponent has a non-null NavigationFilter set. In this manner the NavigationFilter can effectively restrict where the cursor can be positioned. Similarly DefaultCaret will call into the NavigationFilter when the user is changing the selection to further restrict where the cursor can be positioned.

Subclasses can conditionally call into supers implementation to restrict where the cursor can be placed, or call directly into the FilterBypass.

**Since:** 1.4 **See Also:**[Caret](http://docs.google.com/javax/swing/text/Caret.html), [DefaultCaret](http://docs.google.com/javax/swing/text/DefaultCaret.html), [View](http://docs.google.com/javax/swing/text/View.html)

| **Nested Class Summary** | |
| --- | --- |
| static class | [**NavigationFilter.FilterBypass**](http://docs.google.com/javax/swing/text/NavigationFilter.FilterBypass.html)            Used as a way to circumvent calling back into the caret to position the cursor. |

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

| **Method Summary** | |
| --- | --- |
| int | [**getNextVisualPositionFrom**](http://docs.google.com/javax/swing/text/NavigationFilter.html#getNextVisualPositionFrom(javax.swing.text.JTextComponent,%20int,%20javax.swing.text.Position.Bias,%20int,%20javax.swing.text.Position.Bias%5B%5D))([JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) text, int pos, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias, int direction, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasRet)            Returns the next visual position to place the caret at from an existing position. |
| void | [**moveDot**](http://docs.google.com/javax/swing/text/NavigationFilter.html#moveDot(javax.swing.text.NavigationFilter.FilterBypass,%20int,%20javax.swing.text.Position.Bias))([NavigationFilter.FilterBypass](http://docs.google.com/javax/swing/text/NavigationFilter.FilterBypass.html) fb, int dot, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias)            Invoked prior to the Caret moving the dot. |
| void | [**setDot**](http://docs.google.com/javax/swing/text/NavigationFilter.html#setDot(javax.swing.text.NavigationFilter.FilterBypass,%20int,%20javax.swing.text.Position.Bias))([NavigationFilter.FilterBypass](http://docs.google.com/javax/swing/text/NavigationFilter.FilterBypass.html) fb, int dot, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias)            Invoked prior to the Caret setting the dot. |

| **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)) |

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

### NavigationFilter

public **NavigationFilter**()

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

### setDot

public void **setDot**([NavigationFilter.FilterBypass](http://docs.google.com/javax/swing/text/NavigationFilter.FilterBypass.html) fb,  
 int dot,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias)

Invoked prior to the Caret setting the dot. The default implementation calls directly into the FilterBypass with the passed in arguments. Subclasses may wish to conditionally call super with a different location, or invoke the necessary method on the FilterBypass

**Parameters:**fb - FilterBypass that can be used to mutate caret positiondot - the position >= 0bias - Bias to place the dot at

### moveDot

public void **moveDot**([NavigationFilter.FilterBypass](http://docs.google.com/javax/swing/text/NavigationFilter.FilterBypass.html) fb,  
 int dot,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias)

Invoked prior to the Caret moving the dot. The default implementation calls directly into the FilterBypass with the passed in arguments. Subclasses may wish to conditionally call super with a different location, or invoke the necessary methods on the FilterBypass.

**Parameters:**fb - FilterBypass that can be used to mutate caret positiondot - the position >= 0bias - Bias for new location

### getNextVisualPositionFrom

public int **getNextVisualPositionFrom**([JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) text,  
 int pos,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias,  
 int direction,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasRet)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Returns the next visual position to place the caret at from an existing position. The default implementation simply forwards the method to the root View. Subclasses may wish to further restrict the location based on additional criteria.

**Parameters:**text - JTextComponent containing textpos - Position used in determining next positionbias - Bias used in determining next positiondirection - the direction from the current position that can be thought of as the arrow keys typically found on a keyboard. This will be one of the following values:

* SwingConstants.WEST
* SwingConstants.EAST
* SwingConstants.NORTH
* SwingConstants.SOUTH

biasRet - Used to return resulting Bias of next position **Returns:**the location within the model that best represents the next location visual position **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if direction doesn't have one of the legal values above

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NavigationFilter.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/text/MutableAttributeSet.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/NavigationFilter.FilterBypass.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/NavigationFilter.html)    [**NO FRAMES**](http://docs.google.com/NavigationFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

[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).