| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractDocument.BranchElement.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/AbstractDocument.AttributeContext.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/AbstractDocument.Content.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/AbstractDocument.BranchElement.html)    [**NO FRAMES**](http://docs.google.com/AbstractDocument.BranchElement.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#lnxbz9) |

## **javax.swing.text**

Class AbstractDocument.BranchElement

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

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html), [Element](http://docs.google.com/javax/swing/text/Element.html), [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html), [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Direct Known Subclasses:** [DefaultStyledDocument.SectionElement](http://docs.google.com/javax/swing/text/DefaultStyledDocument.SectionElement.html), [HTMLDocument.BlockElement](http://docs.google.com/javax/swing/text/html/HTMLDocument.BlockElement.html) **Enclosing class:**[AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html)

public class **AbstractDocument.BranchElement**extends [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html)

Implements a composite element that contains other elements.

**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** | |
| --- | --- |

| **Nested classes/interfaces inherited from interface javax.swing.text.**[**AttributeSet**](http://docs.google.com/javax/swing/text/AttributeSet.html) |
| --- |
| [AttributeSet.CharacterAttribute](http://docs.google.com/javax/swing/text/AttributeSet.CharacterAttribute.html), [AttributeSet.ColorAttribute](http://docs.google.com/javax/swing/text/AttributeSet.ColorAttribute.html), [AttributeSet.FontAttribute](http://docs.google.com/javax/swing/text/AttributeSet.FontAttribute.html), [AttributeSet.ParagraphAttribute](http://docs.google.com/javax/swing/text/AttributeSet.ParagraphAttribute.html) |

| **Field Summary** | |
| --- | --- |

| **Fields inherited from interface javax.swing.text.**[**AttributeSet**](http://docs.google.com/javax/swing/text/AttributeSet.html) |
| --- |
| [NameAttribute](http://docs.google.com/javax/swing/text/AttributeSet.html#NameAttribute), [ResolveAttribute](http://docs.google.com/javax/swing/text/AttributeSet.html#ResolveAttribute) |

| **Constructor Summary** | |
| --- | --- |
| [**AbstractDocument.BranchElement**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#AbstractDocument.BranchElement(javax.swing.text.Element,%20javax.swing.text.AttributeSet))([Element](http://docs.google.com/javax/swing/text/Element.html) parent, [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) a)            Constructs a composite element that initially contains no children. |

| **Method Summary** | |
| --- | --- |
| [Enumeration](http://docs.google.com/java/util/Enumeration.html) | [**children**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#children())()            Returns the children of the receiver as an Enumeration. |
| boolean | [**getAllowsChildren**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#getAllowsChildren())()            Returns true if the receiver allows children. |
| [Element](http://docs.google.com/javax/swing/text/Element.html) | [**getElement**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#getElement(int))(int index)            Gets a child element. |
| int | [**getElementCount**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#getElementCount())()            Gets the number of children for the element. |
| int | [**getElementIndex**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#getElementIndex(int))(int offset)            Gets the child element index closest to the given model offset. |
| int | [**getEndOffset**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#getEndOffset())()            Gets the ending offset in the model for the element. |
| [String](http://docs.google.com/java/lang/String.html) | [**getName**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#getName())()            Gets the element name. |
| int | [**getStartOffset**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#getStartOffset())()            Gets the starting offset in the model for the element. |
| boolean | [**isLeaf**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#isLeaf())()            Checks whether the element is a leaf. |
| [Element](http://docs.google.com/javax/swing/text/Element.html) | [**positionToElement**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#positionToElement(int))(int pos)            Gets the child element that contains the given model position. |
| void | [**replace**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#replace(int,%20int,%20javax.swing.text.Element%5B%5D))(int offset, int length, [Element](http://docs.google.com/javax/swing/text/Element.html)[] elems)            Replaces content with a new set of elements. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html#toString())()            Converts the element to a string. |

| **Methods inherited from class javax.swing.text.**[**AbstractDocument.AbstractElement**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) |
| --- |
| [addAttribute](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#addAttribute(java.lang.Object,%20java.lang.Object)), [addAttributes](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#addAttributes(javax.swing.text.AttributeSet)), [containsAttribute](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#containsAttribute(java.lang.Object,%20java.lang.Object)), [containsAttributes](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#containsAttributes(javax.swing.text.AttributeSet)), [copyAttributes](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#copyAttributes()), [dump](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#dump(java.io.PrintStream,%20int)), [getAttribute](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttribute(java.lang.Object)), [getAttributeCount](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttributeCount()), [getAttributeNames](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttributeNames()), [getAttributes](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttributes()), [getChildAt](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getChildAt(int)), [getChildCount](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getChildCount()), [getDocument](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getDocument()), [getIndex](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getIndex(javax.swing.tree.TreeNode)), [getParent](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getParent()), [getParentElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getParentElement()), [getResolveParent](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getResolveParent()), [isDefined](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#isDefined(java.lang.Object)), [isEqual](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#isEqual(javax.swing.text.AttributeSet)), [removeAttribute](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#removeAttribute(java.lang.Object)), [removeAttributes](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#removeAttributes(javax.swing.text.AttributeSet)), [removeAttributes](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#removeAttributes(java.util.Enumeration)), [setResolveParent](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#setResolveParent(javax.swing.text.AttributeSet)) |

| **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)) |

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

### AbstractDocument.BranchElement

public **AbstractDocument.BranchElement**([Element](http://docs.google.com/javax/swing/text/Element.html) parent,  
 [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) a)

Constructs a composite element that initially contains no children.

**Parameters:**parent - The parent elementa - the attributes for the element**Since:** 1.4

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

### positionToElement

public [Element](http://docs.google.com/javax/swing/text/Element.html) **positionToElement**(int pos)

Gets the child element that contains the given model position.

**Parameters:**pos - the position >= 0 **Returns:**the element, null if none

### replace

public void **replace**(int offset,  
 int length,  
 [Element](http://docs.google.com/javax/swing/text/Element.html)[] elems)

Replaces content with a new set of elements.

**Parameters:**offset - the starting offset >= 0length - the length to replace >= 0elems - the new elements

### toString

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

Converts the element to a string.

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

### getName

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

Gets the element name.

**Specified by:**[getName](http://docs.google.com/javax/swing/text/Element.html#getName()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Overrides:**[getName](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getName()) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Returns:**the element name

### getStartOffset

public int **getStartOffset**()

Gets the starting offset in the model for the element.

**Specified by:**[getStartOffset](http://docs.google.com/javax/swing/text/Element.html#getStartOffset()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Specified by:**[getStartOffset](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getStartOffset()) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Returns:**the offset >= 0**See Also:**[Document](http://docs.google.com/javax/swing/text/Document.html), [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html)

### getEndOffset

public int **getEndOffset**()

Gets the ending offset in the model for the element.

**Specified by:**[getEndOffset](http://docs.google.com/javax/swing/text/Element.html#getEndOffset()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Specified by:**[getEndOffset](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getEndOffset()) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Returns:**the offset >= 0 **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if this element has no children**See Also:**[Document](http://docs.google.com/javax/swing/text/Document.html), [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html)

### getElement

public [Element](http://docs.google.com/javax/swing/text/Element.html) **getElement**(int index)

Gets a child element.

**Specified by:**[getElement](http://docs.google.com/javax/swing/text/Element.html#getElement(int)) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Specified by:**[getElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getElement(int)) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Parameters:**index - the child index, >= 0 && < getElementCount() **Returns:**the child element, null if none

### getElementCount

public int **getElementCount**()

Gets the number of children for the element.

**Specified by:**[getElementCount](http://docs.google.com/javax/swing/text/Element.html#getElementCount()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Specified by:**[getElementCount](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getElementCount()) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Returns:**the number of children >= 0

### getElementIndex

public int **getElementIndex**(int offset)

Gets the child element index closest to the given model offset.

**Specified by:**[getElementIndex](http://docs.google.com/javax/swing/text/Element.html#getElementIndex(int)) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Specified by:**[getElementIndex](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getElementIndex(int)) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Parameters:**offset - the offset >= 0 **Returns:**the element index >= 0

### isLeaf

public boolean **isLeaf**()

Checks whether the element is a leaf.

**Specified by:**[isLeaf](http://docs.google.com/javax/swing/text/Element.html#isLeaf()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Specified by:**[isLeaf](http://docs.google.com/javax/swing/tree/TreeNode.html#isLeaf()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)**Specified by:**[isLeaf](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#isLeaf()) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Returns:**true if a leaf

### getAllowsChildren

public boolean **getAllowsChildren**()

Returns true if the receiver allows children.

**Specified by:**[getAllowsChildren](http://docs.google.com/javax/swing/tree/TreeNode.html#getAllowsChildren()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)**Specified by:**[getAllowsChildren](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAllowsChildren()) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Returns:**true if the receiver allows children, otherwise false

### children

public [Enumeration](http://docs.google.com/java/util/Enumeration.html) **children**()

Returns the children of the receiver as an Enumeration.

**Specified by:**[children](http://docs.google.com/javax/swing/tree/TreeNode.html#children()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)**Specified by:**[children](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#children()) in class [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **Returns:**the children of the receiver

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractDocument.BranchElement.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/AbstractDocument.AttributeContext.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/AbstractDocument.Content.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/AbstractDocument.BranchElement.html)    [**NO FRAMES**](http://docs.google.com/AbstractDocument.BranchElement.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [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).