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

## **javax.swing.text**

Class AsyncBoxView.ChildLocator

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

**Enclosing class:**[AsyncBoxView](http://docs.google.com/javax/swing/text/AsyncBoxView.html)

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

A class to manage the effective position of the child views in a localized area while changes are being made around the localized area. The AsyncBoxView may be continuously changing, but the visible area needs to remain fairly stable until the layout thread decides to publish an update to the parent.

**Since:** 1.3

| **Field Summary** | |
| --- | --- |
| protected  [Rectangle](http://docs.google.com/java/awt/Rectangle.html) | [**childAlloc**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#childAlloc)            A shape to use for the child allocation to avoid creating a lot of garbage. |
| protected  [Rectangle](http://docs.google.com/java/awt/Rectangle.html) | [**lastAlloc**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#lastAlloc)            The last seen allocation (for repainting when changes are flushed upward). |
| protected  [AsyncBoxView.ChildState](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildState.html) | [**lastValidOffset**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#lastValidOffset)            The location of the last offset calculation that is valid. |

| **Constructor Summary** | |
| --- | --- |
| [**AsyncBoxView.ChildLocator**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#AsyncBoxView.ChildLocator())()            construct a child locator. |

| **Method Summary** | |
| --- | --- |
| void | [**childChanged**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#childChanged(javax.swing.text.AsyncBoxView.ChildState))([AsyncBoxView.ChildState](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildState.html) cs)            Notification that a child changed. |
| protected  [Shape](http://docs.google.com/java/awt/Shape.html) | [**getChildAllocation**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#getChildAllocation(int))(int index)            Fetch the allocation to use for a child view. |
| [Shape](http://docs.google.com/java/awt/Shape.html) | [**getChildAllocation**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#getChildAllocation(int,%20java.awt.Shape))(int index, [Shape](http://docs.google.com/java/awt/Shape.html) a)            Fetch the allocation to use for a child view. |
| int | [**getViewIndexAtPoint**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#getViewIndexAtPoint(float,%20float,%20java.awt.Shape))(float x, float y, [Shape](http://docs.google.com/java/awt/Shape.html) a)            Fetches the child view index at the given point. |
| protected  int | [**getViewIndexAtVisualOffset**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#getViewIndexAtVisualOffset(float))(float targetOffset)            Locate the view responsible for an offset into the box along the major axis. |
| void | [**paintChildren**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#paintChildren(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) g)            Paint the children that intersect the clip area. |
| protected  void | [**setAllocation**](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildLocator.html#setAllocation(java.awt.Shape))([Shape](http://docs.google.com/java/awt/Shape.html) a)            Copy the currently allocated shape into the Rectangle used to store the current allocation. |

| **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** |
| --- |

### lastValidOffset

protected [AsyncBoxView.ChildState](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildState.html) **lastValidOffset**

The location of the last offset calculation that is valid.

### lastAlloc

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

The last seen allocation (for repainting when changes are flushed upward).

### childAlloc

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

A shape to use for the child allocation to avoid creating a lot of garbage.

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

### AsyncBoxView.ChildLocator

public **AsyncBoxView.ChildLocator**()

construct a child locator.

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

### childChanged

public void **childChanged**([AsyncBoxView.ChildState](http://docs.google.com/javax/swing/text/AsyncBoxView.ChildState.html) cs)

Notification that a child changed. This can effect whether or not new offset calculations are needed. This is called by a ChildState object that has changed it's major span. This can therefore be called by multiple threads.

### paintChildren

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

Paint the children that intersect the clip area.

### getChildAllocation

public [Shape](http://docs.google.com/java/awt/Shape.html) **getChildAllocation**(int index,  
 [Shape](http://docs.google.com/java/awt/Shape.html) a)

Fetch the allocation to use for a child view. This will update the offsets for all children not yet updated before the given index.

### getViewIndexAtPoint

public int **getViewIndexAtPoint**(float x,  
 float y,  
 [Shape](http://docs.google.com/java/awt/Shape.html) a)

Fetches the child view index at the given point. This is called by the various View methods that need to calculate which child to forward a message to. This should be called by a block synchronized on this object, and would typically be followed with one or more calls to getChildAllocation that should also be in the synchronized block.

**Parameters:**x - the X coordinate >= 0y - the Y coordinate >= 0a - the allocation to the View **Returns:**the nearest child index

### getChildAllocation

protected [Shape](http://docs.google.com/java/awt/Shape.html) **getChildAllocation**(int index)

Fetch the allocation to use for a child view. *This does not update the offsets in the ChildState records.*

### setAllocation

protected void **setAllocation**([Shape](http://docs.google.com/java/awt/Shape.html) a)

Copy the currently allocated shape into the Rectangle used to store the current allocation. This would be a floating point rectangle in a Java2D-specific implmentation.

### getViewIndexAtVisualOffset

protected int **getViewIndexAtVisualOffset**(float targetOffset)

Locate the view responsible for an offset into the box along the major axis. Make sure that offsets are set on the ChildState objects up to the given target span past the desired offset.

**Returns:**index of the view representing the given visual location (targetOffset), or -1 if no view represents that location

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