

---

# NSSplitView

<b>Inherits From:</b>	NSView : NSResponder : NSObject
<b>Conforms To:</b>	NSCoding (from NSResponder) NSObject (from NSObject)
<b>Declared In:</b>	AppKit/NSSplitView.h

## Class Description

An NSSplitView object lets several views share a region within a window. The NSSplitView resizes its subviews so that each subview is the same width as the NSSplitView, and the total of the subviews' heights (plus the total of the dividers' thicknesses) is equal to the height of the NSSplitView. The NSSplitView positions its subviews so that the first subview is at the top of the NSSplitView, and each successive subview is positioned below the previous one. The user can set the height of two subviews by moving a horizontal bar called the *divider*, which makes one subview smaller and the other larger. Programmatically, you adjust the relative height of subviews simply by modifying the frame of each of the subviews.

To add a view to an NSSplitView, you use the NSView method **addSubview:**. When the NSSplitView is displayed, it checks to see if its subviews are properly tiled. If not, it invokes the delegate method **splitView:resizeSubviewsWithOldSize:**, allowing the delegate to specify the heights of specific subviews. If the delegate doesn't implement this method, the NSSplitView sends **adjustSubviews** to itself to yield the default tiling behavior.

When a mouse-down occurs in an NSSplitView's divider, the NSSplitView determines the limits of the divider's travel and tracks the mouse to allow the user to drag the divider within these limits. With the following mouse-up, the NSSplitView resizes the two affected subviews, informs the delegate that the subviews were resized, and displays the affected views and divider. The NSSplitView's delegate can constrain the travel of specific dividers by implementing the method **splitView:constrainMinCoordinate:maxCoordinate:ofSubviewAt:**.

## Method Types

Managing component views	– adjustSubviews – dividerThickness – drawDividerInRect:
Assigning a delegate	– delegate – setDelegate:

## Notifications

When the subviews within a split view have been resized

NSSplitViewDidResizeSubviewsNotification

When the subviews within a split view are about to be resized

NSSplitViewWillResizeSubviewsNotification

## Instance Methods

### adjustSubviews

– (void)**adjustSubviews**

Adjusts the heights of the NSSplitView’s subviews so the total height (including the dividers) fills the NSSplitView. The subviews are resized proportionally; the size of a subview relative to the other subviews doesn’t change.

**See also:** – **setDelegate:**, – **setFrame:** (NSView)

### delegate

– (id)**delegate**

Returns the NSSplitView’s delegate.

### dividerThickness

– (float)**dividerThickness**

Returns the thickness of the divider. By default, this size is the height of the “dimple” + 1.0. You can override this method to change the divider’s height, if necessary.

**See also:** – **drawDividerInRect:**

### drawDividerInRect:

– (void)**drawDividerInRect:**(NSRect)*aRect*

Draws a divider between two of the NSSplitView’s subviews. *aRect* describes the entire divider rectangle in the NSSplitView’s coordinates, which are flipped. The default implementation composites a default “dimple” image to the center of *aRect*; if you override this method and use a different icon to identify the divider, you may want to change the height of the divider.

**See also:** – **dividerThickness**, – **compositeToPoint:** (NSImage)

---

## **setDelegate:**

– (void)**setDelegate:**(id)*anObject*

Makes *anObject* the NSSplitView’s delegate. The notification messages that the delegate can expect to receive are listed at the end of the NSSplitView class specification. The delegate doesn’t need to implement all of the delegate methods.

## **Methods Implemented by the Delegate**

### **splitView:constrainMinCoordinate:maxCoordinate:ofSubviewAt:**

– (void)**splitView:**(NSSplitView \*)*sender* **constrainMinCoordinate:**(float \*)*min*  
**maxCoordinate:**(float \*)*max* **ofSubviewAt:**(int)*offset*

Allows the delegate to constrain the y coordinate limits of a divider when the user drags the mouse. This method is invoked before the NSSplitView begins tracking the mouse to position a divider. When this method is invoked, the limits have already been set and are stored in *min* (the topmost limit) and *max* (the bottommost limit). You may further constrain the limits by setting the variables indicated by *min* and *max*, but you cannot extend the divider limits. *min* and *max* are specified in the NSSplitView’s flipped coordinate system. The divider to be repositioned is indicated by *offset*, an index that counts the dividers from top to bottom starting with divider 0.

### **splitView:resizeSubviewsWithOldSize:**

– (void)**splitView:**(NSSplitView \*)*sender* **resizeSubviewsWithOldSize:**(NSSize)*oldSize*

Allows the delegate to specify custom sizing behavior for the subviews of the NSSplitView. If the delegate implements this method, **splitView:resizeSubviewsWithOldSize:** is invoked after the NSSplitView is resized. The size of the NSSplitView before the user resized it is indicated by *oldSize*; the subviews should be resized such that the sum of the heights of the subviews plus the sum of the thickness of the dividers equals the height of the NSSplitView’s new frame. You can get the height of a divider through the **dividerThickness** method.

**See also:** – **adjustSubviews**, – **setFrame:** (NSView)

### **splitViewDidResizeSubviews:**

– (void)**splitViewDidResizeSubviews:**(NSNotification \*)*aNotification*

Sent by the default notification center to the delegate; *aNotification* is always an NSSplitViewDidResizeSubviewsNotification. If the delegate implements this method, the delegate is

automatically registered to receive this notification. This method is invoked after the NSSplitView resizes two of its subviews in response to the repositioning of a divider.

### **splitViewWillResizeSubviews:**

– (void)**splitViewWillResizeSubviews:**(NSNotification \*)*aNotification*

Sent by the default notification center to the delegate; *aNotification* is always an NSSplitViewWillResizeSubviewsNotification. If the delegate implements this method, the delegate is automatically registered to receive this notification. This method is invoked before the NSSplitView resizes two of its subviews in response to the repositioning of a divider.

## **Notifications**

NSSplitView declares and posts the following notifications. In addition, it posts notifications that are declared by its superclass, NSView. See the NSView class specification for more information.

### **NSSplitViewDidResizeSubviewsNotification**

**Notification Object**                      The notifying NSSplitView.

**Userinfo**                                      none

Posted after the NSSplitView changes the sizes of some or all of its subviews.

**See also:** – splitViewDidResizeSubviews:

### **NSSplitViewWillResizeSubviewsNotification**

**Notification Object**                      The notifying NSSplitView.

**Userinfo**                                      none

Posted before the NSSplitView changes the sizes of some or all of its subviews.

**See also:** – splitViewWillResizeSubviews: