
NSClipView Class Objective-C Reference

[Cocoa > Graphics & Imaging](#)



2006-05-23



Apple Inc.
© 2006 Apple Computer, Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, Cocoa, Mac, Mac OS, Objective-C, and Quartz are trademarks of Apple Inc., registered in the United States and other countries.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY

DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

NSClipView Class Objective-C Reference 5

Class at a Glance	5
Overview	6
Interaction With NSScrollView	6
Tasks	6
Setting the Document View	6
Scrolling	7
Determining Scrolling Efficiency	7
Getting the Visible Portion	7
Setting the Document Cursor	7
Working with Background Color	7
Overriding NSView Methods	8
Instance Methods	8
autoscroll:	8
backgroundColor	8
constrainScrollPoint:	9
copiesOnScroll	9
documentCursor	9
documentRect	10
documentView	10
documentVisibleRect	10
drawsBackground	11
scrollToPoint:	11
setBackground-color:	11
setCopiesOnScroll:	12
setDocumentCursor:	12
setDocumentView:	12
setDrawsBackground:	13
viewBoundsChanged:	13
viewFrameChanged:	14

Document Revision History 15

Index 17

NSClipView Class Objective-C Reference

Inherits from	NSView : NSResponder : NSObject
Conforms to	NSAnimatablePropertyContainer (NSView) NSCoding (NSResponder) NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.0 and later.
Companion guide	Cocoa Drawing Guide
Declared in	NSClipView.h
Related sample code	Quartz Composer WWDC 2005 TextEdit TextEditPlus

Class at a Glance

An NSClipView contains and scrolls the document view displayed by an NSScrollView. You normally don't need to program with NSClipViews, as NSScrollView handles most of the details of their operation.

Principal Attributes

- Efficient scrolling by copying drawn portions of the document view
- Monitoring of document view for automatic update

Interface Builder

- initWithFrame:
 Initializes the NSClipView.

Commonly Used Methods

[setDocumentView:](#) (page 12)

Sets the view scrolled within the NSClipView.

[setCopiesOnScroll:](#) (page 12)

Sets whether the NSClipView copies drawn portions of the document view during scrolling.

Overview

An NSClipView holds the document view of an NSScrollView, clipping the document view to its frame, handling the details of scrolling in an efficient manner, and updating the NSScrollView when the document view's size or position changes. You don't normally use the NSClipView class directly; it's provided primarily as the scrolling machinery for the NSScrollView class. However, you might use the NSClipView class to implement a class similar to NSScrollView.

Interaction With NSScrollView

When using an NSClipView within an NSScrollView (the usual configuration), you should issue messages that control background drawing state to the NSScrollView, rather than messaging the NSClipView directly. This recommendation applies to the following methods:

- - setBackgroundColor:
- - backgroundColor
- - setDrawsBackground:
- - drawsBackground

The NSClipView methods are intended for when the NSClipView is used independently of a containing NSScrollView. In the usual case, NSScrollView should be allowed to manage the background-drawing properties of its associated NSClipView.

There is only one background-drawing state per NSScrollView/NSClipView pair. The two objects do not maintain independent and distinct `drawsBackground` and `backgroundColor` properties; rather, NSScrollView's accessors for these properties largely defer to the associated NSClipView and allow the NSClipView to maintain the state. In Mac OS X v10.2 and earlier system versions, NSScrollView maintained a cache of the last state it set for its NSClipView. If the NSClipView was sent a `setDrawsBackground:` message directly, the cache might not reflect the state accurately. This caching of state has been removed in Mac OS X v10.3.

It is also important to note that sending a `setDrawsBackground:` message with a parameter of NO to an NSScrollView has the added effect of sending the NSClipView a `setCopiesOnScroll:` message with a parameter of NO. The side effect of sending the `setDrawsBackground:` message directly to the NSClipView is the appearance of "trails" (vestiges of previous drawing) in the document view as it is scrolled.

Tasks

Setting the Document View

- [setDocumentView:](#) (page 12)
Sets the receiver's document view to *aView*, removing any previous document view, and sets the origin of the receiver's bounds rectangle to the origin of *aView's* frame rectangle.
- [documentView](#) (page 10)
Returns the receiver's document view.

Scrolling

- [scrollToPoint:](#) (page 11)
Changes the origin of the receiver's bounds rectangle to *newOrigin*.
- [autoscroll:](#) (page 8)
Scrolls the receiver proportionally to *theEvent*'s distance outside of it.
- [constrainScrollPoint:](#) (page 9)
Returns a scroll point adjusted from *proposedNewOrigin*, if necessary, to guarantee the receiver will still lie within its document view.

Determining Scrolling Efficiency

- [setCopiesOnScroll:](#) (page 12)
Controls whether the receiver copies rendered images while scrolling.
- [copiesOnScroll](#) (page 9)
Returns YES if the receiver copies its existing rendered image while scrolling (only drawing exposed portions of its document view), NO if it forces its contents to be redrawn each time.

Getting the Visible Portion

- [documentRect](#) (page 10)
Returns the rectangle defining the document view's frame, adjusted to the size of the receiver if the document view is smaller.
- [documentVisibleRect](#) (page 10)
Returns the exposed rectangle of the receiver's document view, in the document view's own coordinate system.

Setting the Document Cursor

- [setDocumentCursor:](#) (page 12)
Sets the cursor object used over the receiver to *aCursor*.
- [documentCursor](#) (page 9)
Returns the cursor object used when the cursor lies over the receiver.

Working with Background Color

- [drawsBackground](#) (page 11)
Returns YES if the receiver draws its background color.
- [setDrawsBackground:](#) (page 13)
Sets whether the receiver draws its background color, depending on the Boolean value *flag*.
- [setBackground-color:](#) (page 11)
Sets the receiver's background color to *aColor*.

- [backgroundColor](#) (page 8)
Returns the color of the receiver's background.

Overriding NSView Methods

- [viewBoundsChanged:](#) (page 13)
Handles an `NSViewBoundsDidChangeNotification`, passed in the *aNotification* argument, by updating a containing `NSScrollView` based on the new bounds.
- [viewFrameChanged:](#) (page 14)
Handles an `NSViewFrameDidChangeNotification`, passed in the *aNotification* argument, by updating a containing `NSScrollView` based on the new frame.

Instance Methods

autoscroll:

Scrolls the receiver proportionally to *theEvent's* distance outside of it.

- (BOOL)autoscroll:(NSEvent *)*theEvent*

Discussion

theEvent's location should be expressed in the window's base coordinate system (which it normally is), not the receiving `NSClipView's`. Returns YES if any scrolling is performed; otherwise returns NO.

Never invoke this method directly; instead, the `NSClipView's` document view should repeatedly send itself `autoscroll:` messages when the cursor is dragged outside the `NSClipView's` frame during a modal event loop initiated by a mouse-down event. The `NSView` class implements `autoscroll:` to forward the message to the receiver's superview; thus the message is ultimately forwarded to the `NSClipView`.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSClipView.h`

backgroundColor

Returns the color of the receiver's background.

- (NSColor *)backgroundColor

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setBackground-color:](#) (page 11)

Declared In

NSClipView.h

constrainScrollPoint:

Returns a scroll point adjusted from *proposedNewOrigin*, if necessary, to guarantee the receiver will still lie within its document view.

- (NSPoint)constrainScrollPoint:(NSPoint)*proposedNewOrigin*

Discussion

For example, if *proposedNewOrigin*'s y coordinate lies to the left of the document view's origin, then the y coordinate returned is set to that of the document view's origin.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [scrollToPoint:](#) (page 11)

Declared In

NSClipView.h

copiesOnScroll

Returns YES if the receiver copies its existing rendered image while scrolling (only drawing exposed portions of its document view), NO if it forces its contents to be redrawn each time.

- (BOOL)copiesOnScroll

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setCopiesOnScroll:](#) (page 12)

Declared In

NSClipView.h

documentCursor

Returns the cursor object used when the cursor lies over the receiver.

- (NSCursor *)documentCursor

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setDocumentCursor:](#) (page 12)

Declared In

NSClipView.h

documentRect

Returns the rectangle defining the document view's frame, adjusted to the size of the receiver if the document view is smaller.

- (NSRect)documentRect

Discussion

In other words, this rectangle is always at least as large as the receiver itself.

The document rectangle is used in conjunction with an NSClipView's bounds rectangle to determine values for the indicators of relative position and size between the NSClipView and its document view. For example, NSScrollView uses these rectangles to set the size and position of the knobs in its scrollers. When the document view is much larger than the NSClipView, the knob is small; when the document view is near the same size, the knob is large; and when the document view is the same size or smaller, there is no knob.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [reflectScrolledClipView:](#) (NSScrollView)
- [documentVisibleRect](#) (page 10)

Declared In

NSClipView.h

documentView

Returns the receiver's document view.

- (id)documentView

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setDocumentView:](#) (page 12)

Declared In

NSClipView.h

documentVisibleRect

Returns the exposed rectangle of the receiver's document view, in the document view's own coordinate system.

- (NSRect)documentVisibleRect

Discussion

Note that this rectangle doesn't reflect the effects of any clipping that may occur above the NSClipView itself. To get the portion of the document view that's guaranteed to be visible, send it a `visibleRect` message.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [documentRect](#) (page 10)

Declared In

NSClipView.h

drawsBackground

Returns YES if the receiver draws its background color.

- (BOOL)drawsBackground

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setDrawsBackground:](#) (page 13)

Declared In

NSClipView.h

scrollToPoint:

Changes the origin of the receiver's bounds rectangle to *newOrigin*.

- (void)scrollToPoint:(NSPoint)newOrigin

Availability

Available in Mac OS X v10.0 and later.

See Also

- [constrainScrollPoint:](#) (page 9)

Related Sample Code

WhackedTV

Declared In

NSClipView.h

setBackground-color:

Sets the receiver's background color to *aColor*.

- (void)setBackgroundColor:(NSColor *)aColor

Availability

Available in Mac OS X v10.0 and later.

See Also

- [backgroundColor](#) (page 8)

Declared In

NSClipView.h

setCopiesOnScroll:

Controls whether the receiver copies rendered images while scrolling.

- (void)setCopiesOnScroll:(BOOL)*flag*

Discussion

If *flag* is YES, the receiver copies the existing rendered image to its new location while scrolling and only draws exposed portions of its document view. If *flag* is NO, the receiver always forces its document view to draw itself on scrolling.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [copiesOnScroll](#) (page 9)

Declared In

NSClipView.h

setDocumentCursor:

Sets the cursor object used over the receiver to *aCursor*.

- (void)setDocumentCursor:(NSCursor *)*aCursor*

Availability

Available in Mac OS X v10.0 and later.

See Also

- [documentCursor](#) (page 9)

Declared In

NSClipView.h

setDocumentView:

Sets the receiver's document view to *aView*, removing any previous document view, and sets the origin of the receiver's bounds rectangle to the origin of *aView*'s frame rectangle.

- (void)setDocumentView:(NSView *)*aView*

Discussion

If the receiver is contained in an `NSScrollView`, you should send the `NSScrollView` a `setDocumentView:` message instead, so it can perform whatever updating it needs.

In the process of setting the document view, this method registers the receiver for the notifications `NSViewFrameDidChangeNotification` and `NSViewBoundsDidChangeNotification`, adjusts the key view loop to include the new document view, and updates a parent `NSScrollView`'s display if needed using `reflectScrolledClipView:`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [documentView](#) (page 10)

Declared In

`NSClipView.h`

setDrawsBackground:

Sets whether the receiver draws its background color, depending on the Boolean value *flag*.

```
- (void)setDrawsBackground:(BOOL)flag
```

Discussion

If your `NSClipView` is enclosed in an `NSScrollView`, you should send the `setDrawsBackground:` message to the `NSScrollView`. Sending a `setDrawsBackground:` message with a parameter of `NO` to an `NSScrollView` has the added effect of sending the `NSClipView` a `setCopiesOnScroll:` message with a parameter of `NO`. The side effect of sending the `setDrawsBackground:` message directly to the `NSClipView` is the appearance of “trails” (vestiges of previous drawing) in the document view as it is scrolled.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [drawsBackground](#) (page 11)

Declared In

`NSClipView.h`

viewBoundsChanged:

Handles an `NSViewBoundsDidChangeNotification`, passed in the *aNotification* argument, by updating a containing `NSScrollView` based on the new bounds.

```
- (void)viewBoundsChanged:(NSNotification *)aNotification
```

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSClipView.h`

viewFrameChanged:

Handles an `NSViewFrameDidChangeNotification`, passed in the *aNotification* argument, by updating a containing `NSScrollView` based on the new frame.

```
- (void)viewFrameChanged:(NSNotification *)aNotification
```

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSClipView.h`

Document Revision History

This table describes the changes to *NSClipView Class Objective-C Reference*.

Date	Notes
2006-05-23	First publication of this content as a separate document.

REVISION HISTORY

Document Revision History

Index

viewFrameChanged: instance method [14](#)

A

autoscroll: instance method [8](#)

B

backgroundColor instance method [8](#)

C

constrainScrollPoint: instance method [9](#)

copiesOnScroll instance method [9](#)

D

documentCursor instance method [9](#)

documentRect instance method [10](#)

documentView instance method [10](#)

documentVisibleRect instance method [10](#)

drawsBackground instance method [11](#)

S

scrollToPoint: instance method [11](#)

setBackgroundcolor: instance method [11](#)

setCopiesOnScroll: instance method [12](#)

setDocumentCursor: instance method [12](#)

setDocumentView: instance method [12](#)

setDrawsBackground: instance method [13](#)

V

viewBoundsChanged: instance method [13](#)