

---

# NSSegmentedCell Class Reference

User Experience: Windows & Views



2009-05-06



Apple Inc.  
© 2009 Apple 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, and Mac OS 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

## **[NSSegmentedCell Class Reference](#)** 5

---

<a href="#">Overview</a>	5
<a href="#">Tasks</a>	5
<a href="#">Specifying the Number of Segments</a>	5
<a href="#">Specifying the Selected Segment</a>	5
<a href="#">Specifying the Tracking Mode</a>	6
<a href="#">Configuring Individual Segments</a>	6
<a href="#">Drawing Custom Content</a>	7
<a href="#">Specifying Segment Visual Styles</a>	7
<a href="#">Instance Methods</a>	7
<a href="#">drawSegment:inFrame:withView:</a>	7
<a href="#">imageForSegment:</a>	8
<a href="#">imageScalingForSegment:</a>	8
<a href="#">interiorBackgroundStyleForSegment:</a>	9
<a href="#">isEnabledForSegment:</a>	9
<a href="#">isSelectedForSegment:</a>	10
<a href="#">labelForSegment:</a>	10
<a href="#">makeNextSegmentKey</a>	10
<a href="#">makePreviousSegmentKey</a>	11
<a href="#">menuForSegment:</a>	11
<a href="#">segmentCount</a>	12
<a href="#">segmentStyle</a>	12
<a href="#">selectedSegment</a>	12
<a href="#">selectSegmentWithTag:</a>	13
<a href="#">setEnabled:forSegment:</a>	13
<a href="#">setImage:forSegment:</a>	13
<a href="#">setImageScaling:forSegment:</a>	14
<a href="#">setLabel:forSegment:</a>	14
<a href="#">setMenu:forSegment:</a>	15
<a href="#">setSegmentCount:</a>	15
<a href="#">setSegmentStyle:</a>	16
<a href="#">setSelected:forSegment:</a>	16
<a href="#">setSelectedSegment:</a>	17
<a href="#">setTag:forSegment:</a>	17
<a href="#">setToolTip:forSegment:</a>	18
<a href="#">setTrackingMode:</a>	18
<a href="#">setWidth:forSegment:</a>	18
<a href="#">tagForSegment:</a>	19
<a href="#">toolTipForSegment:</a>	19
<a href="#">trackingMode</a>	20
<a href="#">widthForSegment:</a>	20

## CONTENTS

Constants 21  
    NSSegmentSwitchTracking 21

**Document Revision History 23**

---

**Index 25**

---

# NSSegmentedCell Class Reference

---

<b>Inherits from</b>	NSActionCell : NSCell : NSObject
<b>Conforms to</b>	NSCoding (NSCell) NSCopying (NSCell) NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/AppKit.framework
<b>Availability</b>	Available in Mac OS X v10.3 and later.
<b>Companion guide</b>	Segmented Controls Programming Guide for Cocoa
<b>Declared in</b>	NSSegmentedCell.h

## Overview

The `NSSegmentedCell` class implements the appearance and behavior of a horizontal button divided into multiple segments. This class is used in conjunction with the `NSSegmentedControl` class to implement a segmented control.

You can customize the attributes of a segmented control using the methods of `NSSegmentedCell`. To customize the appearance of individual segments, you can also subclass and override the [drawSegment:inFrame:withView:](#) (page 7) method.

## Tasks

### Specifying the Number of Segments

- [setSegmentCount:](#) (page 15)  
Sets the number of segments in the receiver.
- [segmentCount](#) (page 12)  
Returns the number of segments in the receiver.

### Specifying the Selected Segment

- [setSelectedSegment:](#) (page 17)  
Sets the selected segment of the receiver.

- [setSelected:forSegment:](#) (page 16)  
Sets the selection state of the specified segment.
- [selectSegmentWithTag:](#) (page 13)  
Selects the segment with the specified tag.
- [makeNextSegmentKey](#) (page 10)  
Selects the next segment.
- [makePreviousSegmentKey](#) (page 11)  
Selects the previous segment.
- [selectedSegment](#) (page 12)  
Returns the index of the selected segment of the receiver.
- [isSelectedForSegment:](#) (page 10)  
Returns a Boolean value indicating whether the specified segment is selected,

## Specifying the Tracking Mode

- [setTrackingMode:](#) (page 18)  
Sets the tracking mode for the receiver.
- [trackingMode](#) (page 20)  
Returns the tracking mode of the receiver.

## Configuring Individual Segments

- [setLabel:forSegment:](#) (page 14)  
Sets the label for the specified segment.
- [labelForSegment:](#) (page 10)  
Returns the label of the specified segment.
- [setImage:forSegment:](#) (page 13)  
Sets the image for the specified segment.
- [imageForSegment:](#) (page 8)  
Returns the image associated with the specified segment.
- [setImageScaling:forSegment:](#) (page 14)  
Sets the image scaling mode for the specified segment.
- [imageScalingForSegment:](#) (page 8)  
Returns the image scaling mode associated with the specified segment.
- [setWidth:forSegment:](#) (page 18)  
Sets the width of the specified segment.
- [widthForSegment:](#) (page 20)  
Returns the width of the specified segment.
- [setEnabled:forSegment:](#) (page 13)  
Sets the enabled state of the specified segment
- [isEnabledForSegment:](#) (page 9)  
Returns a Boolean value indicating whether the specified segment is enabled.

- [setMenu:forSegment:](#) (page 15)  
Sets the menu for the specified segment.
- [menuForSegment:](#) (page 11)  
Returns the menu for the specified segment.
- [setToolTip:forSegment:](#) (page 18)  
Sets the tool tip for the specified segment.
- [toolTipForSegment:](#) (page 19)  
Returns the tool tip of the specified segment.
- [setTag:forSegment:](#) (page 17)  
Sets the tag for the specified segment.
- [tagForSegment:](#) (page 19)  
Returns the tag of the specified segment.

## Drawing Custom Content

- [drawSegment:inFrame:withView:](#) (page 7)  
Draws the segment in the specified view.

## Specifying Segment Visual Styles

- [interiorBackgroundStyleForSegment:](#) (page 9)  
Returns the interior background style for the specified segment.
- [segmentStyle](#) (page 12)  
Returns the visual style used to display the receiver.
- [setSegmentStyle:](#) (page 16)  
Sets the visual style used to display the receiver.

## Instance Methods

### **drawSegment:inFrame:withView:**

Draws the segment in the specified view.

```
(void)drawSegment:(NSInteger)segment inFrame:(NSRect)frame withView:(NSView *)controlView
```

#### Parameters

*segment*

The index of the segment to draw. This method raises an `NSRangeException` if the index is out of bounds.

*frame*

The rectangle in which to draw the segment. This rectangle is specified in user space coordinates of the specified view.

*controlView*

The view in which to draw the segment.

#### Discussion

You can override this method to provide a custom appearance for segmented controls. You should not call this method directly. It is called for you automatically by the control when it needs to be redrawn.

#### Availability

Available in Mac OS X v10.3 and later.

#### See Also

- [drawWithFrame:inView:](#) (NSCell)

#### Declared In

NSSegmentedCell.h

## imageForSegment:

Returns the image associated with the specified segment.

- (NSImage \*)imageForSegment:(NSInteger)segment

#### Parameters

*segment*

The index of the segment whose image you want to get. This method raises an `NSRangeException` if the index is out of bounds.

#### Availability

Available in Mac OS X v10.3 and later.

#### See Also

- [setImage:forSegment:](#) (page 13)

#### Declared In

NSSegmentedCell.h

## imageScalingForSegment:

Returns the image scaling mode associated with the specified segment.

- (NSImageScaling)imageScalingForSegment:(NSInteger)segment

#### Parameters

*segment*

The index of the segment whose image scaling mode you want to get. This method raises an `NSRangeException` if the index is out of bounds.

#### Return Value

The scaling mode in use for the specified segment. For the possible values see `Segmented Control Visual Styles`. If no value has been explicitly set `NSImageScaleProportionallyDown` is returned.

#### Availability

Available in Mac OS X v10.5 and later.

**See Also**

- [setImageScaling:forSegment:](#) (page 14)

**Declared In**

NSSegmentedCell.h

**interiorBackgroundStyleForSegment:**

Returns the interior background style for the specified segment.

- (NSBackgroundStyle)interiorBackgroundStyleForSegment:(NSInteger)segment

**Parameters**

*segment*

The index of the segment whose background style you want to get. This method raises an `NSRangeException` if the index is out of bounds..

**Return Value**

The background style to use for specified segment See Background Styles in NSCell for possible values.

**Discussion**

The interior background style describes the surface drawn onto in `drawInteriorWithFrame:inView:`.

This is both an override point and a useful method to call. In a custom segment cell with a custom bezel you can override this method to describe the surface on a per-segment basis.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

NSSegmentedCell.h

**isEnabledForSegment:**

Returns a Boolean value indicating whether the specified segment is enabled.

- (BOOL)isEnabledForSegment:(NSInteger)segment

**Parameters**

*segment*

The index of the segment whose enabled state you want to get. This method raises an `NSRangeException` if the index is out of bounds.

**Return Value**

YES if the segment is enabled; otherwise, NO.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [setEnabled:forSegment:](#) (page 13)

**Declared In**

NSSegmentedCell.h

## isSelectedForSegment:

Returns a Boolean value indicating whether the specified segment is selected,

- (BOOL)isSelectedForSegment:(NSInteger)segment

### Parameters

*segment*

The index of the segment whose selection state you want to get. This method raises an `NSRangeException` if the index is out of bounds.

### Return Value

YES if the segment is selected; otherwise, NO.

### Availability

Available in Mac OS X v10.3 and later.

### See Also

- [setSelected:forSegment:](#) (page 16)

### Declared In

`NSSegmentedCell.h`

## labelForSegment:

Returns the label of the specified segment.

- (NSString \*)labelForSegment:(NSInteger)segment

### Parameters

*segment*

The index of the segment whose label you want to get. This method raises an `NSRangeException` if the index is out of bounds.

### Availability

Available in Mac OS X v10.3 and later.

### See Also

- [setLabel:forSegment:](#) (page 14)

### Declared In

`NSSegmentedCell.h`

## makeNextSegmentKey

Selects the next segment.

- (void)makeNextSegmentKey

### Discussion

The next segment is the one to the right of the currently selected segment. For the last segment, the selection wraps back to the beginning of the control.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [makePreviousSegmentKey](#) (page 11)

**Declared In**

NSSegmentedCell.h

**makePreviousSegmentKey**

Selects the previous segment.

- (void)makePreviousSegmentKey

**Discussion**

The previous segment is the one to the left of the currently selected segment. For the first segment, the selection wraps around to the last segment of the control.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [makeNextSegmentKey](#) (page 10)

**Declared In**

NSSegmentedCell.h

**menuForSegment:**

Returns the menu for the specified segment.

- (NSMenu \*)menuForSegment:(NSInteger)segment

**Parameters**

*segment*

The index of the segment whose menu you want to get. This method raises an `NSRangeException` if the index is out of bounds.

**Return Value**

The menu associated with the segment; otherwise, `nil`.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [setMenu:forSegment:](#) (page 15)

**Declared In**

NSSegmentedCell.h

## segmentCount

Returns the number of segments in the receiver.

- (NSInteger)segmentCount

### Return Value

The number of segments in the receiver.

### Availability

Available in Mac OS X v10.3 and later.

### See Also

- [setSegmentCount:](#) (page 15)

### Declared In

NSSegmentedCell.h

## segmentStyle

Returns the visual style used to display the receiver.

- (NSSegmentStyle)segmentStyle

### Return Value

An `NSSegmentStyle` value that specifies the visual display used by the receiver. For possible values see “Segmented Control Visual Styles” in *NSSegmentedControl Class Reference*.

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

NSSegmentedCell.h

## selectedSegment

Returns the index of the selected segment of the receiver.

- (NSInteger)selectedSegment

### Return Value

The index of the currently selected segment, or -1 if no segment is selected. If the receiver allows multiple selections, this method returns the most recently selected segment.

### Availability

Available in Mac OS X v10.3 and later.

### See Also

- [setSelectedSegment:](#) (page 17)

### Declared In

NSSegmentedCell.h

## selectSegmentWithTag:

Selects the segment with the specified tag.

```
- (BOOL)selectSegmentWithTag:(NSInteger) tag
```

### Parameters

*tag*

The tag associated with the desired segment.

### Return Value

YES if the segment was selected successfully; otherwise, NO.

### Discussion

Typically, you use Interface Builder to specify the tag for each segment. You may also set this value programmatically using the [setTag:forSegment:](#) (page 17) method.

### Availability

Available in Mac OS X v10.4 and later.

### See Also

- [setTag:forSegment:](#) (page 17)

### Declared In

NSSegmentedCell.h

## setEnabled:forSegment:

Sets the enabled state of the specified segment

```
- (void)setEnabled:(BOOL) flag forSegment:(NSInteger) segment
```

### Parameters

*flag*

YES to enable the segment; otherwise, NO to disable it.

*segment*

The index of the segment you want to enable or disable. This method raises an `NSRangeException` if the index is out of bounds.

### Availability

Available in Mac OS X v10.3 and later.

### See Also

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

### Declared In

NSSegmentedCell.h

## setImage:forSegment:

Sets the image for the specified segment.

```
- (void)setImage:(NSImage *) image forSegment:(NSInteger) segment
```

**Parameters***image*

The image to apply to the segment or `nil` if you want to clear the existing image. Images are not scaled to fit inside a segment. If the image is larger than the available space, it is clipped.

*segment*

The index of the segment whose image you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [imageForSegment:](#) (page 8)

**Declared In**

`NSSegmentedCell.h`

**setImageScaling:forSegment:**

Sets the image scaling mode for the specified segment.

```
- (void)setImageScaling:(NSImageScaling)scaling forSegment:(NSInteger)segment
```

**Parameters***scaling*

The scaling mode to assign to the specified segment. For the possible values see `Segmented Control Visual Styles`.

*segment*

The index of the segment whose image scaling mode you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Discussion**

The image scaling mode for a segment affects how the image inside the corresponding cell is positioned and resized when the cell itself grows or shrinks. The image scaling mode does not itself cause the cell to change size in any way. If a cell does not contain an image, the scaling mode has no effect.

**Availability**

Available in Mac OS X v10.5 and later.

**See Also**

- [imageScalingForSegment:](#) (page 8)

**Declared In**

`NSSegmentedCell.h`

**setLabel:forSegment:**

Sets the label for the specified segment.

```
- (void)setLabel:(NSString *)label forSegment:(NSInteger)segment
```

**Parameters***label*

The label you want to display in the segment. If the width of the string is greater than the width of the segment, the string's text is truncated during drawing.

*segment*

The index of the segment whose label you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [labelForSegment:](#) (page 10)

**Declared In**

`NSSegmentedCell.h`

**setMenu:forSegment:**

Sets the menu for the specified segment.

```
- (void)setMenu:(NSMenu *)menu forSegment:(NSInteger)segment
```

**Parameters***menu*

The menu you want to add to the segment or `nil` to clear the current menu. This menu is displayed when the user clicks and holds the mouse button while the mouse is over the segment.

*segment*

The index of the segment whose menu you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Discussion**

Adding a menu to a segment allows that segment to be used as a pop-up button.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

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

**Declared In**

`NSSegmentedCell.h`

**setSegmentCount:**

Sets the number of segments in the receiver.

```
- (void)setSegmentCount:(NSInteger)count
```

**Parameters***count*

The number of segments the receiver should have. If this value is less than the number of segments currently in the receiver, segments are removed from the right of the control. Similarly, if the number is greater than the current number of segments, the new segments are added on the right. This value must be between 0 and 2049.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [segmentCount](#) (page 12)

**Declared In**

NSSegmentedCell.h

**setSegmentStyle:**

Sets the visual style used to display the receiver.

```
- (void)setSegmentStyle:(NSSegmentStyle)segmentStyle
```

**Parameters***segmentStyle*

An `NSSegmentStyle` value that specifies the visual display used by the receiver. For possible values see “Segmented Control Visual Styles” in *NSSegmentedControl Class Reference* in *NSSegmentedControl Class Reference*.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

NSSegmentedCell.h

**setSelected:forSegment:**

Sets the selection state of the specified segment.

```
- (void)setSelected:(BOOL)flag forSegment:(NSInteger)segment
```

**Parameters***flag*

YES if you want to select the segment; otherwise, NO.

*segment*

The index of the segment whose selection state you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Discussion**

If the receiver allows only a single selection, this method deselects any other selected segments.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [isSelectedForSegment:](#) (page 10)

**Declared In**

NSSegmentedCell.h

**setSelectedSegment:**

Sets the selected segment of the receiver.

- (void)setSelectedSegment:(NSInteger)selectedSegment

**Parameters**

*selectedSegment*

The zero-based index of the desired segment. This method raises an `NSRangeException` if the index is out of bounds.

**Discussion**

If the receiver allows multiple selections, this method selects the specified segment using [setSelected:forSegment:](#) (page 16).

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [selectedSegment](#) (page 12)

**Declared In**

NSSegmentedCell.h

**setTag:forSegment:**

Sets the tag for the specified segment.

- (void)setTag:(NSInteger)tag forSegment:(NSInteger)segment

**Parameters**

*tag*

The tag of the segment.

*segment*

The index of the segment whose tool tag you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [tagForSegment:](#) (page 19)

**Declared In**

NSSegmentedCell.h

**setToolTip:forSegment:**

Sets the tool tip for the specified segment.

```
- (void)setToolTip:(NSString *)toolTip forSegment:(NSInteger)segment
```

**Parameters**

*toolTip*

The text of the tool tip you want to display for the segment.

*segment*

The index of the segment whose tool tip you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Discussion**

Tool tips are currently not displayed.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [toolTipForSegment:](#) (page 19)

**Declared In**

`NSSegmentedCell.h`

**setTrackingMode:**

Sets the tracking mode for the receiver.

```
- (void)setTrackingMode:(NSSegmentSwitchTracking)trackingMode
```

**Parameters**

*trackingMode*

The tracking mode to use for the segments. Possible values for *trackingMode* are described in [NSSegmentSwitchTracking](#) (page 21).

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [trackingMode](#) (page 20)

**Declared In**

`NSSegmentedCell.h`

**setWidth:forSegment:**

Sets the width of the specified segment.

```
- (void)setWidth:(CGFloat)width forSegment:(NSInteger)segment
```

**Parameters***width*

The width of the segment, measured in points. Specify the value 0 if you want the segment to be sized to fit the available space automatically.

*segment*

The index of the segment whose width you want to set. This method raises an `NSRangeException` if the index is out of bounds.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [widthForSegment:](#) (page 20)

**Declared In**

`NSSegmentedCell.h`

**tagForSegment:**

Returns the tag of the specified segment.

- (NSInteger)tagForSegment:(NSInteger)segment

**Parameters***segment*

The index of the segment whose tool tag you want to get. This method raises an `NSRangeException` if the index is out of bounds.

**Return Value**

The tag of the segment.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [setTag:forSegment:](#) (page 17)

**Declared In**

`NSSegmentedCell.h`

**toolTipForSegment:**

Returns the tool tip of the specified segment.

- (NSString \*)toolTipForSegment:(NSInteger)segment

**Parameters***segment*

The index of the segment whose tool tip you want to get. This method raises an `NSRangeException` if the index is out of bounds.

**Return Value**

The text of the tool tip.

**Discussion**

Tool tips are currently not displayed.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [setToolTip:forSegment:](#) (page 18)

**Declared In**

NSSegmentedCell.h

**trackingMode**

Returns the tracking mode of the receiver.

- (NSSegmentSwitchTracking)trackingMode

**Return Value**

The tracking mode used for the segments Possible values for *trackingMode* are described in [NSSegmentSwitchTracking](#) (page 21).The default value is [NSSegmentSwitchTrackingSelectOne](#) (page 21).

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [setTrackingMode:](#) (page 18)

**Declared In**

NSSegmentedCell.h

**widthForSegment:**

Returns the width of the specified segment.

- (CGFloat)widthForSegment:(NSInteger)segment

**Parameters**

*segment*

The index of the segment whose width you want to get. This method raises an `NSRangeException` if the index is out of bounds.

**Availability**

Available in Mac OS X v10.3 and later.

**See Also**

- [setWidth:forSegment:](#) (page 18)

**Declared In**

NSSegmentedCell.h

## Constants

### NSSegmentSwitchTracking

An `NSSegmentedCell` object uses the following constants, which describe the various tracking modes for a cell. You access these values using `setTrackingMode:` (page 18) and `trackingMode` (page 20) methods of the class.

```
typedef enum {
    NSSegmentSwitchTrackingSelectOne = 0,
    NSSegmentSwitchTrackingSelectAny = 1,
    NSSegmentSwitchTrackingMomentary = 2
} NSSegmentSwitchTracking;
```

#### Constants

`NSSegmentSwitchTrackingSelectOne`  
**Only one segment may be selected.**  
 Available in Mac OS X v10.3 and later.  
 Declared in `NSSegmentedCell.h`.

`NSSegmentSwitchTrackingSelectAny`  
**Any segment can be selected.**  
 Available in Mac OS X v10.3 and later.  
 Declared in `NSSegmentedCell.h`.

`NSSegmentSwitchTrackingMomentary`  
**A segment is selected only when tracking.**  
 Available in Mac OS X v10.3 and later.  
 Declared in `NSSegmentedCell.h`.

#### Availability

Available in Mac OS X v10.3 and later.

#### Declared In

`NSSegmentedCell.h`



# Document Revision History

---

This table describes the changes to *NSSegmentedCell Class Reference*.

Date	Notes
2009-05-06	Added missing API.
2009-01-06	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History

# Index

---

## D

---

`drawSegment:inFrame:withView:` **instance method** [7](#)

## I

---

`imageForSegment:` **instance method** [8](#)  
`imageScalingForSegment:` **instance method** [8](#)  
`interiorBackgroundStyleForSegment:` **instance method** [9](#)  
`isEnabledForSegment:` **instance method** [9](#)  
`isSelectedForSegment:` **instance method** [10](#)

## L

---

`labelForSegment:` **instance method** [10](#)

## M

---

`makeNextSegmentKey` **instance method** [10](#)  
`makePreviousSegmentKey` **instance method** [11](#)  
`menuForSegment:` **instance method** [11](#)

## N

---

`NSSegmentSwitchTracking` **data type** [21](#)  
`NSSegmentSwitchTrackingMomentary` **constant** [21](#)  
`NSSegmentSwitchTrackingSelectAny` **constant** [21](#)  
`NSSegmentSwitchTrackingSelectOne` **constant** [21](#)

## S

---

`segmentCount` **instance method** [12](#)  
`segmentStyle` **instance method** [12](#)  
`selectedSegment` **instance method** [12](#)  
`selectSegmentWithTag:` **instance method** [13](#)  
`setEnabled:forSegment:` **instance method** [13](#)  
`setImage:forSegment:` **instance method** [13](#)  
`setImageScaling:forSegment:` **instance method** [14](#)  
`setLabel:forSegment:` **instance method** [14](#)  
`setMenu:forSegment:` **instance method** [15](#)  
`setSegmentCount:` **instance method** [15](#)  
`setSegmentStyle:` **instance method** [16](#)  
`setSelected:forSegment:` **instance method** [16](#)  
`setSelectedSegment:` **instance method** [17](#)  
`setTag:forSegment:` **instance method** [17](#)  
`setToolTip:forSegment:` **instance method** [18](#)  
`setTrackingMode:` **instance method** [18](#)  
`setWidth:forSegment:` **instance method** [18](#)

## T

---

`tagForSegment:` **instance method** [19](#)  
`toolTipForSegment:` **instance method** [19](#)  
`trackingMode` **instance method** [20](#)

## W

---

`widthForSegment:` **instance method** [20](#)