NSBox Class Reference

Cocoa > Graphics & Imaging

2008-10-15

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

Simultaneously published in the United States and Canada.

Contents

NSBox Class 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	Boxes
Declared in	NSBox.h
Related sample code	Aperture Image Resizer CoreRecipes EnhancedDataBurn LayerBackedOpenGLView Quartz Composer QCTV

Overview

The `NSBox` class implements simple views that can title themselves and draw a border around their content. These objects are known as **boxes**. You can use box to group, visually, some number of other views.

Subclassing Notes

An `NSBox` object is a view that draws a line around its rectangular bounds and that displays a title on or near the line (or might display neither line nor title). You can adjust the style of the line (bezel, grooved, or plain) as well as the placement and font of the title. An `NSBox` also has a content view to which other views can be added; it thus offers a way for an application to group related views. You could create a custom subclass of `NSBox` that alters or augments its appearance or that modifies its grouping behavior. For example, you might add color to the lines or background, add a new line style, or have the views in the group automatically snap to an invisible grid when added.

Methods to Override

You must override the `drawRect:` method (inherited from `NSView`) if you want to customize the appearance of your `NSBox` objects. Depending on the visual effect you're trying to achieve, you may have to invoke `super`'s implementation first. For example, if you are compositing a small image in a corner of the box, you

would invoke the superclass implementation first. If you're adding a new style of line, you would provide a way to store a request for this line type (such as a boolean instance variable and related accessor methods). Then, in `drawRect:`, if a request for this line type exists, you would draw the entire view yourself (that is, without calling `super`). Otherwise, you would invoke the superclass implementation.

If you wish to change grouping behavior or other behavioral characteristics of the `NSBox` class, consider overriding `setContentView:` (page 13), `sizeToFit` (page 18), or `addSubview:` (inherited from `NSView`).

Special Considerations

If you are drawing the custom `NSBox` entirely by yourself, and you want it to look exactly like the superclass object (except for your changes), it may take some effort and time to get the details right.

Tasks

Configuring Boxes

- `borderRect` (page 8)
  Returns the rectangle in which the receiver's border is drawn.
- `boxType` (page 9)
  Returns the receiver's box type.
- `setBoxType:` (page 13)
  Sets the box type.
- `borderType` (page 8)
  Returns the receiver's border type.
- `setBorderType:` (page 12)
  Sets the border type to *`aType`*, which must be a valid border type.
- `isTransparent` (page 11)
  Indicates whether the receiver is transparent.
- `setTransparent:` (page 17)
  Specifies whether the receiver is transparent.
- `title` (page 18)
  Returns the receiver's title.
- `setTitle:` (page 15)
  Sets the title of the box and marks the region of the receiver within the title rectangle as needing display.
- `titleFont` (page 19)
  Returns the font object used to draw the receiver's title.
- `setTitleFont:` (page 16)
  Sets the font object used to draw the receiver's title and marks the region of the receiver within the title rectangle as needing display.
- `titlePosition` (page 19)
  Returns a constant representing the title position.

- `setTitlePosition:` (page 16)
  Sets the position of the box's title.
- `setTitleWithMnemonic:` (page 17)
  Sets the title of the receiver with a character denoted as an access key.
- `titleCell` (page 19)
  Returns the cell used to display the receiver's title.
- `titleRect` (page 20)
  Returns the rectangle in which the receiver's title is drawn.

Customizing

- `borderColor` (page 8)
  Returns the color of the receiver's border when the receiver is a custom box with a simple line border.
- `setBorderColor:` (page 11)
  Specifies the receiver's border color.
- `borderWidth` (page 9)
  Returns the width of the receiver's border when the receiver is a custom box with a simple line border.
- `setBorderWidth:` (page 12)
  Specifies the receiver's border width.
- `cornerRadius` (page 10)
  Returns the radius of the receiver's corners when the receiver is a custom box with a simple line border.
- `setCornerRadius:` (page 14)
  Specifies the receiver's corner radius.
- `fillColor` (page 11)
  Returns the color of the receiver's background when the receiver is a custom box with a simple line border.
- `setFillColor:` (page 15)
  Specifies the receiver's fill color.

Managing Content

- `contentView` (page 10)
  Returns the receiver's content view.
- `setContentView:` (page 13)
  Sets the receiver's content view.
- `contentViewMargins` (page 10)
  Returns the distances between the border and the content view.
- `setContentViewMargins:` (page 14)
  Sets the horizontal and vertical distance between the border of the receiver and its content view.

Sizing

- `setFrameFromContentFrame:` (page 15)
  Places the receiver so its content view lies on the specified frame.
- `sizeToFit` (page 18)
  Resizes and moves the receiver's content view so it just encloses its subviews.

Instance Methods

borderColor

Returns the color of the receiver's border when the receiver is a custom box with a simple line border.

```
- (NSColor *)borderColor
```

Return Value
The receiver's border color. It must be a custom box—that is, it has a type of `NSBoxCustom` (page 22)—and it must have a border style of `NSLineBorder`.

Availability
Available in Mac OS X v10.5 and later.

See Also
- `setBorderColor:` (page 11)

Declared In
`NSBox.h`

borderRect

Returns the rectangle in which the receiver's border is drawn.

```
- (NSRect)borderRect
```

Return Value
The rectangle in which the border of the `NSBox` is drawn.

Availability
Available in Mac OS X v10.0 and later.

Declared In
`NSBox.h`

borderType

Returns the receiver's border type.

```
- (NSBorderType)borderType
```

Return Value
A constant describing the type of border. Border types are defined in `NSView.h`. Currently, the following border types are defined: `NSNoBorder`,`NSLineBorder`,`NSBezelBorder`, `NSGrooveBorder`.

By default, the border type of an `NSBox` is `NSGrooveBorder`.

Availability
Available in Mac OS X v10.0 and later.

See Also
`- setBorderType:` (page 12)

Declared In
`NSBox.h`

borderWidth

Returns the width of the receiver's border when the receiver is a custom box with a simple line border.

```
- (CGFloat)borderWidth
```

Return Value
The receiver's border width. It must be a custom box—that is, it has a type of `NSBoxCustom` (page 22)—and it must have a border style of `NSLineBorder`.

Availability
Available in Mac OS X v10.5 and later.

See Also
`- setBorderWidth:` (page 12)

Declared In
`NSBox.h`

boxType

Returns the receiver's box type.

```
- (NSBoxType)boxType
```

Return Value
A constant describing the type of box. These constants are described in `NSBoxType` (page 21). By default, the box type of an `NSBox` is `NSBoxPrimary`.

Availability
Available in Mac OS X v10.0 and later.

See Also
`- setBoxType:` (page 13)

Declared In
`NSBox.h`

contentView

Returns the receiver's content view.

```
- (id)contentView
```

Return Value
The content view of the `NSBox` object. The content view is created automatically when the box is created and resized as the box is resized (you should never send frame-altering messages directly to a box's content view). You can replace it with an `NSView` of your own through the `setContentView:` (page 13) method.

Availability
Available in Mac OS X v10.0 and later.

See Also
`- setContentView:` (page 13)

Related Sample Code
Quartz Composer QCTV

Declared In
`NSBox.h`

contentViewMargins

Returns the distances between the border and the content view.

```
- (NSSize)contentViewMargins
```

Return Value
The width (the horizontal distance between the innermost edge of the border and the content view) and height (the vertical distance between the innermost edge of the border and the content view) describing the distance between the border and the content view. By default, these are both 5.0 in the box's coordinate system.

Availability
Available in Mac OS X v10.0 and later.

See Also
`- setContentViewMargins:` (page 14)

Declared In
`NSBox.h`

cornerRadius

Returns the radius of the receiver's corners when the receiver is a custom box with a simple line border.

```
- (CGFloat)cornerRadius
```

Return Value
The receiver's corner radius. It must be a custom box—that is, it has a type of `NSBoxCustom` (page 22)—and it must have a border style of `NSLineBorder`.

Availability
Available in Mac OS X v10.5 and later.

See Also
- `setCornerRadius:` (page 14)

Declared In
`NSBox.h`

fillColor

Returns the color of the receiver's background when the receiver is a custom box with a simple line border.

```
- (NSColor *)fillColor
```

Return Value
The receiver's fill color. It must be a custom box—that is, it has a type of `NSBoxCustom` (page 22)—and it must have a border style of `NSLineBorder`.

Availability
Available in Mac OS X v10.5 and later.

See Also
- `setFillColor:` (page 15)

Declared In
`NSBox.h`

isTransparent

Indicates whether the receiver is transparent.

```
- (BOOL)isTransparent
```

Return Value
`YES` when the receiver is transparent, `NO` otherwise.

Availability
Available in Mac OS X v10.5 and later.

See Also
- `setTransparent:` (page 17)

Declared In
`NSBox.h`

setBorderColor:

Specifies the receiver's border color.

```
- (void)setBorderColor:(NSColor *)borderColor
```

Parameters

borderColor

Border color for the receiver.

Special Considerations

Functional only when the receiver's box type (`boxType` (page 9)) is `NSBoxCustom` and its border type (`borderType` (page 8)) is `NSLineBorder`.

Availability

Available in Mac OS X v10.5 and later.

See Also

`- borderColor` (page 8)

Declared In

`NSBox.h`

setBorderType:

Sets the border type to *aType*, which must be a valid border type.

```
- (void)setBorderType:(NSBorderType)aType
```

Parameters

aType

A constant describing the type of border. Border types are defined in `NSView.h`. Currently, the following border types are defined: `NSNoBorder`,`NSLineBorder`,`NSBezelBorder`, `NSGrooveBorder`.

Discussion

If the size of the new border is different from that of the old border, the content view is resized to absorb the difference, and the box is marked for redisplay.

Availability

Available in Mac OS X v10.0 and later.

See Also

`- borderType` (page 8)

`- setNeedsDisplay:` (`NSView`)

Declared In

`NSBox.h`

setBorderWidth:

Specifies the receiver's border width.

```
- (void)setBorderWidth:(CGFloat)borderWidth
```

Parameters

borderWidth

Border width for the receiver.

Special Considerations

Functional only when the receiver's box type (`boxType` (page 9)) is `NSBoxCustom` and its border type (`borderType` (page 8)) is `NSLineBorder`.

Availability
Available in Mac OS X v10.5 and later.

See Also
- `borderWidth` (page 9)

Declared In
`NSBox.h`

setBoxType:

Sets the box type.

```
- (void)setBoxType:(NSBoxType)boxType
```

Parameters

boxType

A constant describing the type of box; this must be a valid box type. These constants are described in `NSBoxType` (page 21).

Availability
Available in Mac OS X v10.0 and later.

See Also
- `boxType` (page 9)

Declared In
`NSBox.h`

setContentView:

Sets the receiver's content view.

```
- (void)setContentView:(NSView *)aView
```

Parameters

aView

The new content view. The `NSView` object is resized to fit within the box's current content area and the box is marked for redisplay.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `contentView` (page 10)
- `setFrameFromContentFrame:` (page 15)
- `sizeToFit` (page 18)
- `setNeedsDisplay:` (NSView)

Related Sample Code
Quartz Composer QCTV

Declared In
`NSBox.h`

setContentViewMargins:

Sets the horizontal and vertical distance between the border of the receiver and its content view.

```
- (void)setContentViewMargins:(NSSize)offsetSize
```

Parameters

offsetSize

The width and height of the offset between the box's border and content view. The horizontal value is applied (reckoned in the box's coordinate system) fully and equally to the left and right sides of the box. The vertical value is similarly applied to the top and bottom.

Discussion
Unlike changing a box's other attributes, such as its title position or border type, changing the offsets doesn't automatically resize the content view. In general, you should send a `sizeToFit` (page 18) message to the box after changing the size of its offsets. This message causes the content view to remain unchanged while the box is sized to fit around it.

Availability
Available in Mac OS X v10.0 and later.

See Also
`- contentViewMargins` (page 10)

Declared In
`NSBox.h`

setCornerRadius:

Specifies the receiver's corner radius.

```
- (void)setCornerRadius:(CGFloat)cornerRadius
```

Parameters

cornerRadius

Corner radius for the receiver.

Special Considerations

Functional only when the receiver's box type (`boxType` (page 9)) is `NSBoxCustom` and its border type (`borderType` (page 8)) is `NSLineBorder`.

Availability
Available in Mac OS X v10.5 and later.

See Also
`- cornerRadius` (page 10)

Declared In
NSBox.h

setFillColor:

Specifies the receiver's fill color.

```
- (void)setFillColor:(NSColor *)fillColor
```

Parameters

fillColor

Fill color for the receiver.

Special Considerations

Functional only when the receiver's box type (boxType (page 9)) is NSBoxCustom and its border type (borderType (page 8)) is NSLineBorder.

Availability
Available in Mac OS X v10.5 and later.

See Also
- fillColor (page 11)

Declared In
NSBox.h

setFrameFromContentFrame:

Places the receiver so its content view lies on the specified frame.

```
- (void)setFrameFromContentFrame:(NSRect)contentFrame
```

Parameters

contentFrame

The rectangle specifying the frame of the box's content view, reckoned in the coordinate system of the box's superview. The box is marked for redisplay.

Availability
Available in Mac OS X v10.0 and later.

See Also
- setContentViewMargins: (page 14)
- setFrame: (NSView)
- setNeedsDisplay: (NSView)

Declared In
NSBox.h

setTitle:

Sets the title of the box and marks the region of the receiver within the title rectangle as needing display.

```
- (void)setTitle:(NSString *)aString
```

Parameters

aString

The new title of the `NSBox`. The default title of an `NSBox` is "Title." If the size of the new title is different from that of the old title, the content view is resized to absorb the difference.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `title` (page 18)
- `titleRect` (page 20)
- `setNeedsDisplayInRect:` (`NSView`)

Related Sample Code
Quartz Composer QCTV

Declared In
`NSBox.h`

setTitleFont:

Sets the font object used to draw the receiver's title and marks the region of the receiver within the title rectangle as needing display.

```
- (void)setTitleFont:(NSFont *)aFont
```

Parameters

aFont

The `NSFont` object used to draw the box's title.

Discussion
By default, the title is drawn using the small system font (obtained using (`smallSystemFontSize` as the parameter of `systemFontOfSize:`, both `NSFont` class methods). If the size of the new font is different from that of the old font, the content view is resized to absorb the difference.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `titleFont` (page 19)
- `setNeedsDisplayInRect:` (`NSView`)

Declared In
`NSBox.h`

setTitlePosition:

Sets the position of the box's title.

```
- (void)setTitlePosition:(NSTitlePosition)aPosition
```

Parameters

aPosition

A constant describing the position of the box's title. These constants are described in `NSTitlePosition` (page 20). The default position is `NSAtTop`.

Discussion

If the new title position changes the size of the box's border area, the content view is resized to absorb the difference, and the box is marked as needing redisplay.

Availability

Available in Mac OS X v10.0 and later.

See Also

`- titlePosition` (page 19)

`- setNeedsDisplay:` (`NSView`)

Declared In

`NSBox.h`

setTitleWithMnemonic:

Sets the title of the receiver with a character denoted as an access key.

```
- (void)setTitleWithMnemonic:(NSString *)aString
```

Discussion

Mnemonics are not supported in Mac OS X.

By default, a box's title is "Title." The content view is not automatically resized, and the box is not marked for redisplay.

Availability

Available in Mac OS X v10.0 and later.

See Also

`- setTitleWithMnemonic:` (`NSCell`)

Declared In

`NSBox.h`

setTransparent:

Specifies whether the receiver is transparent.

```
- (void)setTransparent:(BOOL)transparent
```

Parameters

transparent

`YES` makes the receiver transparent.

`NO` makes the receiver opaque.

Availability

Available in Mac OS X v10.5 and later.

See Also
- `isTransparent` (page 11)

Declared In
`NSBox.h`

sizeToFit

Resizes and moves the receiver's content view so it just encloses its subviews.

```
- (void)sizeToFit
```

Discussion
The receiver is then moved and resized to wrap around the content view. The receiver's width is constrained so its title will be fully displayed.

You should invoke this method after:

- Adding a subview (to the content view)
- Altering the size or location of such a subview
- Setting the margins around the content view

The mechanism by which the content view is moved and resized depends on whether the object responds to its own `sizeToFit` message: If it does respond, then that message is sent, and the content view is expected to be so modified. If the content view doesn't respond, the box moves and resizes the content view itself.

Availability
Available in Mac OS X v10.0 and later.

Declared In
`NSBox.h`

title

Returns the receiver's title.

```
- (NSString *)title
```

Return Value
The title of the `NSBox`. By default, a box's title is "Title."

Availability
Available in Mac OS X v10.0 and later.

See Also
- `setTitle:` (page 15)

Related Sample Code
Quartz Composer QCTV

Declared In
`NSBox.h`

titleCell

Returns the cell used to display the receiver's title.

```
- (id)titleCell
```

Return Value
The `NSCell` object used to display the title.

Availability
Available in Mac OS X v10.0 and later.

Declared In
`NSBox.h`

titleFont

Returns the font object used to draw the receiver's title.

```
- (NSFont *)titleFont
```

Return Value
The `NSFont` object used to draw the title.

Discussion
By default, the title is drawn using the small system font (obtained using (`smallSystemFontSize` as the parameter of `systemFontOfSize:`, both `NSFont` class methods).

Availability
Available in Mac OS X v10.0 and later.

See Also
`- setTitleFont:` (page 16)

Declared In
`NSBox.h`

titlePosition

Returns a constant representing the title position.

```
- (NSTitlePosition)titlePosition
```

Return Value
A constant representing the position of the receiver's title. See `NSTitlePosition` (page 20) for a list of these constants.

Availability
Available in Mac OS X v10.0 and later.

See Also
`- setTitlePosition:` (page 16)

Declared In
NSBox.h

titleRect

Returns the rectangle in which the receiver's title is drawn.

```
- (NSRect)titleRect
```

Return Value
The rectangle in which the title is drawn.

Availability
Available in Mac OS X v10.0 and later.

See Also
- setTitlePosition: (page 16)
- setTitle: (page 15)
- setTitleFont: (page 16)
- setFrameFromContentFrame: (page 15)
- sizeToFit (page 18)

Declared In
NSBox.h

Constants

NSTitlePosition

Specify the location of a box's title with respect to its border.

```
typedef enum _NSTitlePosition {
    NSNoTitle     = 0,
    NSAboveTop    = 1,
    NSAtTop       = 2,
    NSBelowTop    = 3,
    NSAboveBottom = 4,
    NSAtBottom    = 5,
    NSBelowBottom = 6
} NSTitlePosition;
```

Constants
NSNoTitle
The box has no title.
Available in Mac OS X v10.0 and later.
Declared in NSBox.h.

`NSAboveTop`
: Title positioned above the box's top border.

  Available in Mac OS X v10.0 and later.

  Declared in `NSBox.h`.

`NSAtTop`
: Title positioned within the box's top border.

  Available in Mac OS X v10.0 and later.

  Declared in `NSBox.h`.

`NSBelowTop`
: Title positioned below the box's top border.

  Available in Mac OS X v10.0 and later.

  Declared in `NSBox.h`.

`NSAboveBottom`
: Title positioned above the box's bottom border.

  Available in Mac OS X v10.0 and later.

  Declared in `NSBox.h`.

`NSAtBottom`
: Title positioned within the box's bottom border.

  Available in Mac OS X v10.0 and later.

  Declared in `NSBox.h`.

`NSBelowBottom`
: Title positioned below the box's bottom border.

  Available in Mac OS X v10.0 and later.

  Declared in `NSBox.h`.

Availability
Available in Mac OS X v10.0 and later.

Declared In
`NSBox.h`

NSBoxType

These constants and data type identifies box types, which, in conjunction with a box's border type, define the appearance of the box.

```
enum {
    NSBoxPrimary   = 0,
    NSBoxSecondary = 1,
    NSBoxSeparator = 2,
    NSBoxOldStyle  = 3,
    NSBoxCustom    = 4
};
typedef NSUInteger NSBoxType;
```

Constants

`NSBoxPrimary`

Specifies the primary box appearance. This is the default box type.

Available in Mac OS X v10.0 and later.

Declared in `NSBox.h`.

`NSBoxSecondary`

Specifies the secondary box appearance.

Available in Mac OS X v10.0 and later.

Declared in `NSBox.h`.

`NSBoxSeparator`

Specifies that the box is a separator.

Available in Mac OS X v10.0 and later.

Declared in `NSBox.h`.

`NSBoxOldStyle`

Specifies that the box is a Mac OS X v10.2–style box.

Available in Mac OS X v10.0 and later.

Declared in `NSBox.h`.

`NSBoxCustom`

Specifies that the appearance of the box is determined entirely by the by box-configuration methods, without automatically applying Apple human interface guidelines. See "`Customizing`" (page 7) for details.

Available in Mac OS X v10.5 and later.

Declared in `NSBox.h`.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSBox.h`

Document Revision History

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

Date	Notes
2008-10-15	Corrected descriptions of getter methods in "Customizing" method group, setTitleFont:, and titleFont.
2007-05-30	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

Document Revision History

Index