
NSShadow Class Reference

Graphics & Animation: 2D Drawing



2009-02-04



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.

Java and all Java-based trademarks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. 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

NSShadow Class Reference 5

Overview	5
Adopted Protocols	6
Tasks	6
Creating a Shadow	6
Managing a Shadow	6
Setting the Shadow	6
Instance Methods	7
init	7
set	7
setShadowBlurRadius:	7
setShadowColor:	8
setShadowOffset:	8
shadowBlurRadius	9
shadowColor	9
shadowOffset	10

Document Revision History 11

Index 13

NSShadow Class Reference

Inherits from	NSObject
Conforms to	NSCoding NSCopying NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.3 and later.
Companion guide	Cocoa Drawing Guide
Declared in	NSShadow.h
Related sample code	ComplexBrowser DockTile Reducer SpeedometerView WebKitPluginWithJavaScript

Overview

An `NSShadow` object encapsulates the attributes used to create a drop shadow during drawing operations.

Shadows are always drawn in the default user coordinate space, regardless of any transformations applied to that space. This means that rotations, translations and other transformations of the current transformation matrix (the CTM) do not affect the resulting shadow. Another way to think about this is that changes to the CTM do not move or change the apparent position of the shadow's light source.

There are two positional parameters for a shadow: an x-offset and a y-offset. These values are expressed using a single `NSSize` data type and using the units of the default user coordinate space. Positive values for these offsets extend up and to the right.

In addition to its positional parameters, a shadow also contains a blur radius, which specifies how much a drawn object's image mask is blurred before it is composited onto the destination. A value of 0 means there is no blur. Larger values give correspondingly larger amounts of blurring.

An `NSShadow` object may be used in one of two ways. First, it may be set, like a color or a font, in which case its attributes are applied to all content drawn thereafter—or at least until another shadow is applied or a previous graphics state is restored. Second, it may be used as the value for the `NSShadowAttributeName` text attribute, in which case it is applied to the glyphs corresponding to the characters bearing this attribute.

Adopted Protocols

NSCoding

`encodeWithCoder:`

`initWithCoder:`

NSCopying

`copyWithZone:`

Tasks

Creating a Shadow

- `init` (page 7)
Returns an `NSShadow` object initialized with default values.

Managing a Shadow

- `setShadowOffset:` (page 8)
Sets the offset values for the receiver.
- `shadowOffset` (page 10)
Returns the offset values for the receiver.
- `setShadowBlurRadius:` (page 7)
Sets the blur radius of the receiver.
- `shadowBlurRadius` (page 9)
Returns the blur radius of the receiver.
- `setShadowColor:` (page 8)
Sets the shadow color for the receiver.
- `shadowColor` (page 9)
Returns the color for the receiver.

Setting the Shadow

- `set` (page 7)
Sets the shadow of subsequent drawing operations to the shadow represented by the receiver.

Instance Methods

init

Returns an `NSShadow` object initialized with default values.

```
- (id)init
```

Return Value

An `NSShadow` object initialized with 0 as its offset, 0 as its blur radius, and the default color as its color. The returned object may be different from the original receiver.

Availability

Available in Mac OS X v10.3 and later.

Declared In

`NSShadow.h`

set

Sets the shadow of subsequent drawing operations to the shadow represented by the receiver.

```
- (void)set
```

Discussion

The shadow attributes of the receiver are used until another shadow is set or until the graphics state is restored.

Availability

Available in Mac OS X v10.3 and later.

Related Sample Code

`DockTile`

`Reducer`

`SpeedometerView`

`WebKitPluginStarter`

`WebKitPluginWithJavaScript`

Declared In

`NSShadow.h`

setShadowBlurRadius:

Sets the blur radius of the receiver.

```
- (void)setShadowBlurRadius:(CGFloat)va1
```

Parameters*val*

The blur radius, as measured in the default user coordinate space. A value of 0 indicates no blur, while larger values produce correspondingly larger blurring. This value must not be negative.

Availability

Available in Mac OS X v10.3 and later.

See Also

- [shadowBlurRadius](#) (page 9)

Related Sample Code

ComplexBrowser

DockTile

OpenCL NBody Simulation Example

SpeedometerView

WebKitPluginWithJavaScript

Declared In

NSShadow.h

setShadowColor:

Sets the shadow color for the receiver.

```
- (void)setShadowColor:(NSColor *)color
```

Parameters*color*

The shadow color, which must be convertible to an RGBA color. Specify `nil` if you do not want the shadow to be drawn. Your color may contain alpha information.

Availability

Available in Mac OS X v10.3 and later.

See Also

- [shadowColor](#) (page 9)

Related Sample Code

ComplexBrowser

DockTile

OpenCL NBody Simulation Example

SpeedometerView

WebKitPluginWithJavaScript

Declared In

NSShadow.h

setShadowOffset:

Sets the offset values for the receiver.

- (void)setShadowOffset:(NSSize)offset

Parameters

offset

The horizontal and vertical offset values, specified using the `width` and `height` fields of the `NSSize` data type. These offsets are measured using the default user coordinate space and are not affected by custom transformations. This means that positive values always extend up and to the right from the user's perspective.

Availability

Available in Mac OS X v10.3 and later.

See Also

- [shadowOffset](#) (page 10)

Related Sample Code

ComplexBrowser

DockTile

OpenCL NBody Simulation Example

SpeedometerView

WebKitPluginWithJavaScript

Declared In

NSShadow.h

shadowBlurRadius

Returns the blur radius of the receiver.

- (CGFloat)shadowBlurRadius

Return Value

The blur radius, as measured in the default user coordinate space. A value of 0 indicates no blur, while larger values produce correspondingly larger blurring. The default value is 0.

Availability

Available in Mac OS X v10.3 and later.

See Also

- [setShadowBlurRadius:](#) (page 7)

Related Sample Code

FunHouse

Declared In

NSShadow.h

shadowColor

Returns the color for the receiver.

- (NSColor *)shadowColor

Return Value

The current shadow color. A `nil` shadow color indicates the shadow is not to be drawn. The default shadow color is black with an alpha of 1/3.

Availability

Available in Mac OS X v10.3 and later.

See Also

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

Related Sample Code

FunHouse

Declared In

NSShadow.h

shadowOffset

Returns the offset values for the receiver.

- (NSSize)shadowOffset

Return Value

The horizontal and vertical offset values, specified using the `width` and `height` fields of the `NSSize` data type. These offsets are measured using the default user coordinate space and are not affected by custom transformations. This means that positive values always extend up and to the right from the user's perspective.

Availability

Available in Mac OS X v10.3 and later.

See Also

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

Related Sample Code

FunHouse

Declared In

NSShadow.h

Document Revision History

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

Date	Notes
2009-02-04	Added a description for NSShadow's init method.
2007-01-30	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

I

init [instance method 7](#)

S

set [instance method 7](#)
setShadowBlurRadius: [instance method 7](#)
setShadowColor: [instance method 8](#)
setShadowOffset: [instance method 8](#)
shadowBlurRadius [instance method 9](#)
shadowColor [instance method 9](#)
shadowOffset [instance method 10](#)