

---

# NSColorPanel Class Reference

User Experience: Controls



2007-02-28



Apple Inc.  
© 2007 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

## **NSColorPanel Class Reference 5**

---

Overview	5
Tasks	5
Obtaining the Shared Color-Panel Object	5
Configuring the Color Panel	6
Managing Color Lists	6
Setting Color Picker Modes	6
Setting Color	7
Getting Color Information	7
Responding to a Color Change	7
Class Methods	7
dragColor:withEvent:fromView:	7
setPickerMask:	8
setPickerMode:	8
sharedColorPanel	9
sharedColorPanelExists	9
Instance Methods	9
accessoryView	9
alpha	10
attachColorList:	10
color	11
detachColorList:	11
isContinuous	11
mode	12
setAccessoryView:	12
setAction:	13
setColor:	13
setContinuous:	14
setMode:	14
setShowsAlpha:	15
setTarget:	15
showsAlpha	16
Delegate Methods	16
changeColor:	16
Constants	17
Color Picker Mode Masks	17
NSColorPanelMode	18
Color Panel Modes	18
Notifications	20
NSColorPanelColorDidChangeNotification	20

**Document Revision History 21**

---

# NSColorPanel Class Reference

---

<b>Inherits from</b>	NSPanel : NSWindow : NSResponder : NSObject
<b>Conforms to</b>	NSUserInterfaceValidations (NSWindow) NSAnimatablePropertyContainer (NSWindow) NSCoding (NSResponder) NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/AppKit.framework
<b>Availability</b>	Available in Mac OS X v10.0 and later.
<b>Companion guide</b>	Color Programming Topics
<b>Declared in</b>	NSColorPanel.h
<b>Related sample code</b>	CWCocoaComponent QTAudioContextInsert QTAudioExtractionPanel QTCoreVideo301 QTKitPlayer

## Overview

The `NSColorPanel` class provides a standard user interface for selecting color in an application. It provides a number of standard color selection modes and, with the `NSColorPickingDefault` and `NSColorPickingCustom` protocols, allows an application to add its own color selection modes. It allows the user to save swatches containing frequently used colors.

## Tasks

### Obtaining the Shared Color-Panel Object

- + [sharedColorPanel](#) (page 9)  
Returns the shared `NSColorPanel` instance, creating it if necessary.
- + [sharedColorPanelExists](#) (page 9)  
Returns a Boolean value indicating whether the `NSColorPanel` has been created already.

## Configuring the Color Panel

- [accessoryView](#) (page 9)  
Returns the accessory view.
- [isContinuous](#) (page 11)  
Returns a Boolean value indicating whether the receiver continuously sends the action message to the target.
- [mode](#) (page 12)  
Returns the color picker mode of the receiver.
- [setAccessoryView:](#) (page 12)  
Sets the accessory view displayed in the receiver.
- [setAction:](#) (page 13)  
Sets the color panel's action message.
- [setContinuous:](#) (page 14)  
Sets the receiver to send the action message to its target continuously as the user sets the color.
- [setMode:](#) (page 14)  
Sets the mode of the receiver the mode is one of the modes allowed by the color mask.
- [setShowsAlpha:](#) (page 15)  
Tells the receiver whether or not to show alpha values and an opacity slider.
- [setTarget:](#) (page 15)  
Sets the target of the receiver.
- [showsAlpha](#) (page 16)  
Returns a Boolean value indicating whether or not the receiver shows alpha values and an opacity slider.

## Managing Color Lists

- [attachColorList:](#) (page 10)  
Adds the list of `NSColor` objects specified to all the color pickers in the receiver that display color lists by invoking [attachColorList:](#) (page 10) on all color pickers in the application.
- [detachColorList:](#) (page 11)  
Removes the list of colors from all the color pickers in the receiver that display color lists by invoking [detachColorList:](#) on all color pickers in the application.

## Setting Color Picker Modes

- + [setPickerMask:](#) (page 8)  
Determines which color selection modes are available in an application's `NSColorPanel`.
- + [setPickerMode:](#) (page 8)  
Specifies the color panel's initial picker.

## Setting Color

- + `dragColor:withEvent:fromView:` (page 7)  
Drags a color into a destination view from the specified source view.
- `setColor:` (page 13)  
Sets the color of the receiver.

## Getting Color Information

- `alpha` (page 10)  
Returns the receiver's current alpha value based on its opacity slider.
- `color` (page 11)  
Returns the currently selected color in the receiver.

## Responding to a Color Change

- `changeColor:` (page 16) *delegate method*  
Sent to the first responder when the user selects a color in an `NSColorPanel` object.

## Class Methods

### **dragColor:withEvent:fromView:**

Drags a color into a destination view from the specified source view.

```
+ (BOOL)dragColor:(NSColor *)color withEvent:(NSEvent *)anEvent fromView:(NSView *)sourceView
```

#### Parameters

*color*

The color to drag.

*anEvent*

The drag event.

*sourceView*

The view from which the color was dragged.

#### Return Value

YES

#### Discussion

This method is usually invoked by the `mouseDown:` method of *sourceView*. The dragging mechanism handles all subsequent events.

Because it is a class method, `dragColor:withEvent:fromView:` can be invoked whether or not the instance of `NSColorPanel` exists.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSColorPanel.h

**setPickerMask:**

Determines which color selection modes are available in an application's NSColorPanel.

```
+ (void)setPickerMask:(NSUInteger)mask
```

**Parameters**

*mask*

One or more logically ORed color mode masks described in [Color Panel Mode Masks](#) (page 17).

**Discussion**

This method has an effect only before an NSColorPanel object is instantiated.

If you create a class that implements the color-picking protocols (NSColorPickingDefault and NSColorPickingCustom), you may want to give it a unique mask—one different from those defined for the standard color pickers. To display your color picker, your application will need to logically OR that unique mask with the standard color mask constants when invoking this method.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [setPickerMode:](#) (page 8)

**Declared In**

NSColorPanel.h

**setPickerMode:**

Specifies the color panel's initial picker.

```
+ (void)setPickerMode:(NSColorPanelMode)mode
```

**Parameters**

*mode*

A constant specifying which color picker mode is initially visible. This is one of the symbolic constants described in [Color Panel Modes](#) (page 18).

**Discussion**

This method may be called at any time, whether or not an application's NSColorPanel has been instantiated.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [setPickerMask:](#) (page 8)

- [setMode:](#) (page 14)



**Declared In**

NSColorPanel.h

## sharedColorPanel

Returns the shared NSColorPanel instance, creating it if necessary.

```
+ (NSColorPanel *)sharedColorPanel
```

**Return Value**

The shared NSColorPanel instance.

**Availability**

Available in Mac OS X v10.0 and later.

**Related Sample Code**

CWCocoaComponent

QTAudioContextInsert

QTAudioExtractionPanel

QTCoreVideo301

QTKitPlayer

**Declared In**

NSColorPanel.h

## sharedColorPanelExists

Returns a Boolean value indicating whether the NSColorPanel has been created already.

```
+ (BOOL)sharedColorPanelExists
```

**Return Value**

YES if the NSColorPanel has been created already; otherwise NO.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [sharedColorPanel](#) (page 9)

**Declared In**

NSColorPanel.h

## Instance Methods

### accessoryView

Returns the accessory view.

- (NSView \*)accessoryView

#### Return Value

The accessory view or `nil` if there is none.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

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

#### Declared In

NSColorPanel.h

## alpha

Returns the receiver's current alpha value based on its opacity slider.

- (CGFloat)alpha

#### Return Value

The alpha value of the `NSColorPanel`. This is 1.0 (opaque) if the panel has no opacity slider.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

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

- [showsAlpha](#) (page 16)

#### Declared In

NSColorPanel.h

## attachColorList:

Adds the list of `NSColor` objects specified to all the color pickers in the receiver that display color lists by invoking [attachColorList:](#) (page 10) on all color pickers in the application.

- (void)attachColorList:(NSColorList \*)colorList

#### Parameters

*colorList*

The list of colors to add to the color pickers in the receiver.

#### Discussion

An application should use this method to add an `NSColorList` saved with a document in its file package or in a directory other than `NSColorList`'s standard search directories.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

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

**Declared In**

NSColorPanel.h

**color**

Returns the currently selected color in the receiver.

- (NSColor \*)color

**Return Value**

The currently selected color.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [setColor:](#) (page 13)**Declared In**

NSColorPanel.h

**detachColorList:**Removes the list of colors from all the color pickers in the receiver that display color lists by invoking `detachColorList:` on all color pickers in the application.

- (void)detachColorList:(NSColorList \*)colorList

**Parameters***colorList*The list of `NSColor` objects to remove from the color pickers in the color panel.**Discussion**Your application should use this method to remove an `NSColorList` saved with a document in its file package or in a directory other than `NSColorList`'s standard search directories.**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [attachColorList:](#) (page 10)**Declared In**

NSColorPanel.h

**isContinuous**

Returns a Boolean value indicating whether the receiver continuously sends the action message to the target.

- (BOOL)isContinuous

**Return Value**

YES if the receiver continuously sends the action message to the target as the user manipulates the color picker; otherwise NO.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [setContinuous:](#) (page 14)

**Declared In**

NSColorPanel.h

**mode**

Returns the color picker mode of the receiver.

- (NSColorPanelMode)mode

**Return Value**

A constant indicating the current color picker mode. See [Color Panel Modes](#) (page 18).

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [setPickerMode:](#) (page 8)

- [setMode:](#) (page 14)

**Declared In**

NSColorPanel.h

**setAccessoryView:**

Sets the accessory view displayed in the receiver.

- (void)setAccessoryView:(NSView \*)aView

**Parameters**

*aView*

The accessory view displayed in the receiver. The accessory view can be any custom view you want to display with `NSColorPanel`, such as a view offering color blends in a drawing program. The accessory view is displayed below the color picker and above the color swatches in the `NSColorPanel`. The `NSColorPanel` automatically resizes to accommodate the accessory view.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [accessoryView](#) (page 9)

**Related Sample Code**

QTCoreVideo301

**Declared In**

NSColorPanel.h

**setAction:**

Sets the color panel's action message.

- (void)setAction:(SEL)*action***Return Value**The action message. When you select a color in the color panel `NSColorPanel` sends its action to its target, provided that neither the action nor the target is `nil`. The action is `NULL` by default.**Discussion**See *Action Messages* for additional information on action messages.**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [setTarget:](#) (page 15)**Related Sample Code**

QTAudioContextInsert

QTAudioExtractionPanel

QTKitImport

QTKitPlayer

**Declared In**

NSColorPanel.h

**setColor:**

Sets the color of the receiver.

- (void)setColor:(NSColor \*)*color***Parameters***color*The color of the `NSColorPanel`.**Discussion**This method posts an [NSColorPanelColorDidChangeNotification](#) (page 20) with the receiver to the default notification center.**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [color](#) (page 11)**Related Sample Code**

QTAudioContextInsert

QTAudioExtractionPanel  
 QTKitImport  
 QTKitPlayer

**Declared In**

NSColorPanel.h

**setContinuous:**

Sets the receiver to send the action message to its target continuously as the user sets the color.

- (void)setContinuous:(BOOL)flag

**Parameters**

*flag*

YES to have the receiver send its action message to its target continuously as the color of the NSColorPanel is set by the user; otherwise NO. Set this to YES if, for example, you want to continuously update the color of the target.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [isContinuous](#) (page 11)

**Declared In**

NSColorPanel.h

**setMode:**

Sets the mode of the receiver the mode is one of the modes allowed by the color mask.

- (void)setMode:(NSColorPanelMode)mode

**Parameters**

*mode*

A constant specifying the mode of the color panel. These constants are described in [Color Picker Modes](#) (page 18). The color mask is set when you first create the shared instance of NSColorPanel for an application.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [setPickerMode:](#) (page 8)

- [mode](#) (page 12)

**Declared In**

NSColorPanel.h

## setShowsAlpha:

Tells the receiver whether or not to show alpha values and an opacity slider.

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

### Parameters

*flag*

YES to have the color panel show alpha values and an opacity slider; otherwise NO.

### Discussion

Note that calling the `NSColor` method `setIgnoresAlpha:` with a value of YES overrides any value set with this method.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [alpha](#) (page 10)
- [showsAlpha](#) (page 16)

### Related Sample Code

AnimatingViews  
 CWCocoaComponent  
 FunHouse  
 Quartz 2D Shadings

### Declared In

NSColorPanel.h

## setTarget:

Sets the target of the receiver.

```
- (void)setTarget:(id)target
```

### Parameters

*target*

The target of the receiver. When you select a color in the color panel `NSColorPanel` sends its action to its target, provided that neither the action nor the target is `nil`. The target is `nil` by default.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [setAction:](#) (page 13)
- [setContinuous:](#) (page 14)

### Related Sample Code

QTAudioContextInsert  
 QTAudioExtractionPanel  
 QTKitImport  
 QTKitPlayer

**Declared In**

NSColorPanel.h

**showsAlpha**

Returns a Boolean value indicating whether or not the receiver shows alpha values and an opacity slider.

- (BOOL)showsAlpha

**Return Value**

YES if the color picker shows alpha values and an opacity slider; otherwise NO.

**Discussion**

Note that calling the `NSColor` method `setIgnoresAlpha:` with a value of YES overrides any value set with `setShowsAlpha:` (page 15).

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [alpha](#) (page 10)
- [setShowsAlpha:](#) (page 15)

**Declared In**

NSColorPanel.h

## Delegate Methods

**changeColor:**

Sent to the first responder when the user selects a color in an `NSColorPanel` object.

- (void)changeColor:(id)sender

**Parameters**

*sender*

The `NSColorPanel` sending the message.

**Discussion**

When the user selects a color in an `NSColorPanel` object, the panel sends a `changeColor:` action message to the first responder. You can override this method in any responder that needs to respond to a color change.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSColorPanel.h



## Constants

### Color Picker Mode Masks

Used to specify in the `setPickerMask:` (page 8) class method which of the color modes the `NSColorPanel` can use.

```
enum {
    NSColorPanelGrayModeMask           = 0x00000001,
    NSColorPanelRGBModeMask           = 0x00000002,
    NSColorPanelCMYKModeMask          = 0x00000004,
    NSColorPanelHSBModeMask           = 0x00000008,
    NSColorPanelCustomPaletteModeMask = 0x00000010,
    NSColorPanelColorListModeMask     = 0x00000020,
    NSColorPanelWheelModeMask         = 0x00000040,
    NSColorPanelCrayonModeMask        = 0x00000080,
    NSColorPanelAllModesMask          = 0x0000ffff
};
```

### Constants

`NSColorPanelGrayModeMask`

**Grayscale-alpha.**

Available in Mac OS X v10.0 and later.

Declared in `NSColorPanel.h`.

`NSColorPanelRGBModeMask`

**Red-green-blue.**

Available in Mac OS X v10.0 and later.

Declared in `NSColorPanel.h`.

`NSColorPanelCMYKModeMask`

**Cyan-yellow-magenta-black.**

Available in Mac OS X v10.0 and later.

Declared in `NSColorPanel.h`.

`NSColorPanelHSBModeMask`

**Hue-saturation-brightness.**

Available in Mac OS X v10.0 and later.

Declared in `NSColorPanel.h`.

`NSColorPanelCustomPaletteModeMask`

**Custom palette.**

Available in Mac OS X v10.0 and later.

Declared in `NSColorPanel.h`.

`NSColorPanelColorListModeMask`

**Custom color list.**

Available in Mac OS X v10.0 and later.

Declared in `NSColorPanel.h`.

NSColorPanelWheelModeMask

Color wheel.

Available in Mac OS X v10.0 and later.

Declared in NSColorPanel.h.

NSColorPanelCrayonModeMask

Crayons.

Declared in NSColorPanel.h.

Available in Mac OS X v10.2 and later.

NSColorPanelAllModesMask

All of the above.

Available in Mac OS X v10.0 and later.

Declared in NSColorPanel.h.

#### Discussion

For more information, see “Choosing the Color Pickers in a Color Panel”.

#### Declared In

NSColorPanel.h

## NSColorPanelMode

A type defined for the enum constants specifying color panel modes.

```
typedef NSInteger NSColorPanelMode;
```

#### Availability

Available in Mac OS X v10.5 and later.

#### Declared In

NSColorPanel.h

## Color Panel Modes

Specify the active color mode used when an application’s instance of NSColorPanel is masked for more than one color mode.

```
enum {
    NSNoModeColorPanel           = -1,
    NSGrayModeColorPanel        = 0,
    NSRGBModeColorPanel          = 1,
    NSCMYKModeColorPanel        = 2,
    NSHSBModeColorPanel         = 3,
    NSCustomPaletteModeColorPanel = 4,
    NSColorListModeColorPanel   = 5,
    NSWheelModeColorPanel       = 6,
    NSCrayonModeColorPanel      = 7
};
```

**Constants**

**NSNoModeColorPanel**  
**Indicates no color panel mode.**  
**Available in Mac OS X version 10.5 and later.**  
**Declared in NSColorPanel.h.**

**NSGrayModeColorPanel**  
**Grayscale-alpha**  
**Available in Mac OS X v10.0 and later.**  
**Declared in NSColorPanel.h.**

**NSRGBModeColorPanel**  
**Red-green-blue**  
**Available in Mac OS X v10.0 and later.**  
**Declared in NSColorPanel.h.**

**NSCMYKModeColorPanel**  
**Cyan-yellow-magenta-black**  
**Available in Mac OS X v10.0 and later.**  
**Declared in NSColorPanel.h.**

**NSHSBModeColorPanel**  
**Hue-saturation-brightness**  
**Available in Mac OS X v10.0 and later.**  
**Declared in NSColorPanel.h.**

**NSCustomPaletteModeColorPanel**  
**Custom palette**  
**Available in Mac OS X v10.0 and later.**  
**Declared in NSColorPanel.h.**

**NSColorListModeColorPanel**  
**Custom color list**  
**Available in Mac OS X v10.0 and later.**  
**Declared in NSColorPanel.h.**

**NSWheelModeColorPanel**  
**Color wheel**  
**Available in Mac OS X v10.0 and later.**  
**Declared in NSColorPanel.h.**

NSCrayonModeColorPanel

Crayons.

Declared in `NSColorPanel.h`.

Available in Mac OS X v10.2 and later.

#### Discussion

These enum constants are specified or returned in the instance methods `mode` (page 12) and `setMode:` (page 14), and in the `setPickerMode:` (page 8) class method. For more information, see “Choosing the Color Pickers in a Color Panel”.

#### Declared In

`NSColorPanel.h`

## Notifications

### NSColorPanelColorDidChangeNotification

Posted when the color of the `NSColorPanel` is set, as when `setColor:` (page 13) is invoked.

The notification object is the notifying `NSColorPanel`. This notification does not contain a `userInfo` dictionary.

#### Availability

Available in Mac OS X v10.0 and later.

#### Declared In

`NSColorPanel.h`

# Document Revision History

---

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

Date	Notes
2007-02-28	Updated for Mac OS X version 10.5.
2006-05-23	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History