NSScreen Class Reference

Cocoa > Graphics & Imaging

2009-01-06

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

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

NSScreen Class Reference 5

Overview 5
Tasks 6
  Getting NSScreen Objects 6
  Getting Screen Information 6
Class Methods 6
  deepestScreen 6
  mainScreen 7
  screens 7
Instance Methods 8
  depth 8
  deviceDescription 8
  frame 9
  supportedWindowDepths 9
  userSpaceScaleFactor 10
  visibleFrame 10

Document Revision History 11

Index 13

NSScreen Class Reference

Inherits from	NSObject
Conforms to	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	NSScreen.h
Related sample code	ColorMatching MyPhoto QTQuartzPlayer RoundTransparentWindow UIElementInspector

Overview

An NSScreen object describes the attributes of a computer's monitor, or screen. An application may use an NSScreen object to retrieve information about a screen and use this information to decide what to display upon that screen. For example, an application may use the deepestScreen (page 6) method to find out which of the available screens can best represent color and then may choose to display all of its windows on that screen.

The application object should be created before you use the methods in this class, so that the application object can make the necessary connection to the window system. You can make sure the application object exists by invoking the sharedApplication method of NSApplication. If you created your application with Xcode, the application object is automatically created for you during initialization.

Note: The NSScreen class is for getting information about the available displays only. If you need additional information or want to change the attributes relating to a display, you must use Quartz Services. For more information, see *Quartz Display Services Reference*.

Tasks

Getting NSScreen Objects

+ mainScreen (page 7)
    Returns the NSScreen object containing the window with the keyboard focus.

+ deepestScreen (page 6)
    Returns an NSScreen object representing the screen that can best represent color.

+ screens (page 7)
    Returns an array of NSScreen objects representing all of the screens available on the system.

Getting Screen Information

- depth (page 8)
    Returns the receiver's current bit depth and colorspace information.

- frame (page 9)
    Returns the dimensions and location of the receiver.

- supportedWindowDepths (page 9)
    Returns a zero-terminated array of the window depths supported by the receiver.

- deviceDescription (page 8)
    Returns the device dictionary for the screen.

- userSpaceScaleFactor (page 10)
    Returns the scaling factor from user space to device space on the screen represented by the receiver.

- visibleFrame (page 10)
    Returns the current location and dimensions of the visible screen.

Class Methods

deepestScreen

Returns an NSScreen object representing the screen that can best represent color.

```
+ (NSScreen *)deepestScreen
```

Return Value
The screen with the highest bit depth.

Discussion
This method always returns an object, even if there is only one screen and it is not a color screen.

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

Declared In
`NSScreen.h`

mainScreen

Returns the `NSScreen` object containing the window with the keyboard focus.

```
+ (NSScreen *)mainScreen
```

Return Value
The main screen object.

Discussion
The main screen is not necessarily the same screen that contains the menu bar or has its origin at (0, 0). The main screen refers to the screen containing the window that is currently receiving keyboard events. It is the main screen because it is the one with which the user is most likely interacting.

The screen containing the menu bar is always the first object (index 0) in the array returned by the `screens` method.

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

See Also
`+ screens` (page 7)

Related Sample Code
ColorMatching

Link Snoop

MyPhoto

QTQuartzPlayer

RoundTransparentWindow

Declared In
`NSScreen.h`

screens

Returns an array of `NSScreen` objects representing all of the screens available on the system.

```
+ (NSArray *)screens
```

Return Value
An array of the `NSScreen` objects available on the current system or `nil` if the screen information could not be obtained from the window system.

Discussion
The screen at index 0 in the returned array corresponds to the primary screen of the user's system. This is the screen that contains the menu bar and whose origin is at the point (0, 0). In the case of mirroring, the first screen is the largest drawable display; if all screens are the same size, it is the screen with the highest pixel depth. This primary screen may not be the same as the one returned by the `mainScreen` (page 7) method, which returns the screen with the active window.

The array should not be cached. Screens can be added, removed, or dynamically reconfigured at any time. When the display configuration is changed, the default notification center sends a `NSApplicationDidChangeScreenParametersNotification` notification.

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

Declared In
`NSScreen.h`

Instance Methods

depth

Returns the receiver's current bit depth and colorspace information.

```
- (NSWindowDepth)depth
```

Return Value
The window depth information. This value cannot be used directly. You must pass it to a function such as `NSBitsPerPixelFromDepth` or `NSColorSpaceFromDepth` to obtain a concrete value for the desired information.

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

Declared In
`NSScreen.h`

deviceDescription

Returns the device dictionary for the screen.

```
- (NSDictionary *)deviceDescription
```

Return Value
A dictionary containing the attributes of the receiver's screen. For the list of keys you can use to retrieve values from the returned dictionary, see `Display Deviceâ  Descriptions`.

Discussion
In addition to the display device constants described in *NSWindow Class Reference*, you can also retrieve the `CGDirectDisplayID` value associated with the screen from this dictionary. To access this value, specify the Objective-C string `@"NSScreenNumber"` as the key when requesting the item from the dictionary. The value associated with this key is an `NSNumber` object containing the display ID value. This string is only valid when used as a key for the dictionary returned by this method.

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

Declared In
`NSScreen.h`

frame

Returns the dimensions and location of the receiver.

```
- (NSRect)frame
```

Return Value
The full screen rectangle at the current resolution. This rectangle includes any space currently occupied by the menu bar and dock.

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

See Also
`- visibleFrame` (page 10)

Related Sample Code
ColorMatching

RoundTransparentWindow

Declared In
`NSScreen.h`

supportedWindowDepths

Returns a zero-terminated array of the window depths supported by the receiver.

```
- (const NSWindowDepth *)supportedWindowDepths
```

Return Value
A C-style array of window depths. The returned values cannot be used directly. You must pass each one to a function such as `NSBitsPerPixelFromDepth` or `NSColorSpaceFromDepth` to obtain a concrete value for the desired screen.

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

Declared In
`NSScreen.h`

userSpaceScaleFactor

Returns the scaling factor from user space to device space on the screen represented by the receiver.

```
- (CGFloat)userSpaceScaleFactor
```

Return Value

The scaling factor, measured in pixels per point, where a point is always equal to 1/72 of an inch. For example, a scaling factor of 2.0 indicates the display has a resolution 2 pixels per point or 144 pixels-per-inch.

Availability

Available in Mac OS X v10.4 and later.

Declared In

`NSScreen.h`

visibleFrame

Returns the current location and dimensions of the visible screen.

```
- (NSRect)visibleFrame
```

Return Value

The rectangle defining the portion of the screen in which it is currently safe to draw your application content.

Discussion

The returned rectangle is always based on the current user-interface settings and does not include the area currently occupied by the dock and menu bar. Because it is based on the current user -interface settings, the returned rectangle can change between calls and should not be cached.

> **Note:** Even when dock hiding is enabled, the rectangle returned by this method may be smaller than the full screen. The system uses a small boundary area to determine when it should display the dock.

Availability

Available in Mac OS X v10.0 and later.

See Also

`- frame` (page 9)

Related Sample Code

CocoaDVDPlayer

Link Snoop

MyPhoto

PDFKitLinker2

QTQuartzPlayer

Declared In

`NSScreen.h`

Document Revision History

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

Date	Notes
2009-01-06	Updated the information regarding what constitutes the main screen.
	Removed the Constants section because all the constants but one are already documented in *NSWindow Class Reference*. The remaining constant is now documented inline with the `deviceDescription` (page 8) method.
2007-01-30	Updated for Mac OS X v10.5
2006-05-23	First publication of this content as a separate document.

Index

D

deepestScreen class method 6
depth instance method 8
deviceDescription instance method 8

F

frame instance method 9

M

mainScreen class method 7

S

screens class method 7
supportedWindowDepths instance method 9

U

userSpaceScaleFactor instance method 10

V

visibleFrame instance method 10