CGBitmapContext Reference

Graphics & Imaging > Quartz

2007-06-28

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

CGBitmapContext Reference

Derived From:	`CGContextRef`
Framework:	ApplicationServices/ApplicationServices.h
Companion guide	Quartz 2D Programming Guide
Declared in	CGBitmapContext.h

Overview

The CGBitmapContext header file defines functions that create and operate on a Quartz bitmap graphics context. A bitmap graphics context is a type of `CGContextRef` that you can use for drawing bits to memory. The functions in this reference operate only on Quartz bitmap graphics contexts created using the functionCGBitmapContextCreate (page 6).

The number of components for each pixel in a bitmap graphics context is specified by a color space (defined by a `CGColorSpaceRef`, which includes RGB, grayscale, and CMYK, and which also may specify a destination color profile). The bitmap graphics context specifies whether the bitmap should contain an alpha channel, and how the bitmap is generated.

Functions by Task

Creating Bitmap Contexts

`CGBitmapContextCreate` (page 6)
: Creates a bitmap graphics context.

`CGBitmapContextCreateImage` (page 7)
: Creates and returns a Quartz image from the pixel data in a bitmap graphics context.

Getting Information About Bitmap Contexts

These functions return the values of attributes specified when a bitmap context is created.

`CGBitmapContextGetBitmapInfo` (page 8)
: Obtains the bitmap information associated with a bitmap graphics context.

`CGBitmapContextGetAlphaInfo` (page 8)
: Returns the alpha information associated with the context, which indicates how a bitmap context handles the alpha component.

`CGBitmapContextGetBitsPerComponent` (page 9)
: Returns the bits per component of a bitmap context.

`CGBitmapContextGetBitsPerPixel` (page 9)
: Returns the bits per pixel of a bitmap context.

`CGBitmapContextGetBytesPerRow` (page 10)
: Returns the bytes per row of a bitmap context.

`CGBitmapContextGetColorSpace` (page 10)
: Returns the color space of a bitmap context.

`CGBitmapContextGetData` (page 10)
: Returns a pointer to the image data associated with a bitmap context.

`CGBitmapContextGetHeight` (page 11)
: Returns the height in pixels of a bitmap context.

`CGBitmapContextGetWidth` (page 11)
: Returns the width in pixels of a bitmap context.

Functions

CGBitmapContextCreate

Creates a bitmap graphics context.

```
CGContextRef CGBitmapContextCreate (
   void *data,
   size_t width,
   size_t height,
   size_t bitsPerComponent,
   size_t bytesPerRow,
   CGColorSpaceRef colorspace,
   CGBitmapInfo bitmapInfo
);
```

Parameters

data

A pointer to the destination in memory where the drawing is to be rendered. The size of this memory block should be at least `(bytesPerRow*height)` bytes.

Starting in Mac OS X v10.3, you can pass `NULL` if you don't care where the data is stored. This frees you from managing your own memory, which reduces memory leak issues. Quartz has more flexibility when it manages data storage for you. For example, it's possible for Quartz to use OpenGL for rendering if it takes care of the memory.

width

The width, in pixels, of the required bitmap.

height

The height, in pixels, of the required bitmap.

bitsPerComponent

The number of bits to use for each component of a pixel in memory. For example, for a 32-bit pixel format and an RGB color space, you would specify a value of 8 bits per component. For more information about supported pixel formats, see *Quartz 2D Programming Guide*.

bytesPerRow

The number of bytes of memory to use per row of the bitmap.

colorspace

The color space to use for the bitmap context. Note that indexed color spaces are not supported for bitmap graphics contexts.

bitmapInfo

A `CGBitmapInfo` constant that specifies whether the bitmap should contain an alpha channel and its relative location in a pixel, along with whether the components are floating-point or integer values. (See *CGImage Reference* for a description `CGBitmapInfo` constants.) In *Quartz 2D Programming Guide*, see "Creating a Bitmap Graphics Context" (in the *Graphics Contexts* chapter) for the color space, bits per pixel, bits per pixel component, and bitmap information constant combinations that you can use when creating a bitmap context with `CGBitmapContextCreate`.

Return Value

A new bitmap context, or `NULL` if a context could not be created. You are responsible for releasing this object using `CGContextRelease`.

Discussion

When you call this function, Quartz creates a bitmap drawing environment—that is, a bitmap context—to your specifications. When you draw into this context, Quartz renders your drawing as bitmapped data in the specified block of memory.

The pixel format for a new bitmap context is determined by three parameters—the number of bits per component, the color space, and an alpha option (expressed as a `CGBitmapInfo` constant). The alpha value determines the opacity of a pixel when it is drawn.

Availability

Available in Mac OS X version 10.0 and later.

Related Sample Code

CarbonSketch

Declared In

`CGBitmapContext.h`

CGBitmapContextCreateImage

Creates and returns a Quartz image from the pixel data in a bitmap graphics context.

```
CGImageRef CGBitmapContextCreateImage (
   CGContextRef c
);
```

Parameters

c

A bitmap graphics context.

Return Value

A CGImage object that contains a snapshot of the bitmap graphics context or `NULL` if the image is not created.

Discussion

The CGImage object returned by this function is created by a copy operation. Subsequent changes to the bitmap graphics context do not affect the contents of the returned image. In some cases the copy operation actually follows copy-on-write semantics, so that the actual physical copy of the bits occur only if the underlying data in the bitmap graphics context is modified. As a consequence, you may want to use the resulting image and release it before you perform additional drawing into the bitmap graphics context. In this way, you can avoid the actual physical copy of the data.

Availability
Available in Mac OS X v10.4 and later.

Declared In
`CGBitmapContext.h`

CGBitmapContextGetAlphaInfo

Returns the alpha information associated with the context, which indicates how a bitmap context handles the alpha component.

```
CGImageAlphaInfo CGBitmapContextGetAlphaInfo (
   CGContextRef c
);
```

Parameters

`context`

A bitmap context.

Return Value
A bitmap information constant. If the specified context is not a bitmap context, `kCGImageAlphaNone` is returned. See `CGImageAlphaInfo` (renamed to `CGBitmapInfo` in Mac OS X v10.4) for more information about values.

Discussion
Every bitmap context contains an attribute that specifies whether the bitmap contains an alpha component, and how it is generated. The alpha component determines the opacity of a pixel when it is drawn.

Availability
Available in Mac OS X version 10.2 and later.

Declared In
`CGBitmapContext.h`

CGBitmapContextGetBitmapInfo

Obtains the bitmap information associated with a bitmap graphics context.

```
CGBitmapInfo CGBitmapContextGetBitmapInfo (
   CGContextRef c
);
```

Parameters

`c`

A bitmap graphics context.

Return Value
The bitmap info of the bitmap graphics context or `0` if `c` is not a bitmap graphics context. See *CGImage Reference* for a description of the `CGBitmapInfo` constants that can be returned.

Discussion

The `CGBitmapInfo` data returned by the function specifies whether the bitmap contains an alpha channel and how the alpha channel is generated, along with whether the components are floating-point or integer.

Availability

Available in Mac OS X v10.4 and later.

Declared In

`CGBitmapContext.h`

CGBitmapContextGetBitsPerComponent

Returns the bits per component of a bitmap context.

```
size_t CGBitmapContextGetBitsPerComponent (
   CGContextRef c
);
```

Parameters

`context`

The bitmap context to examine.

Return Value

The number of bits per component in the specified context, or `0` if the context is not a bitmap context.

Availability

Available in Mac OS X version 10.2 and later.

Declared In

`CGBitmapContext.h`

CGBitmapContextGetBitsPerPixel

Returns the bits per pixel of a bitmap context.

```
size_t CGBitmapContextGetBitsPerPixel (
   CGContextRef c
);
```

Parameters

`context`

The bitmap context to examine.

Return Value

The number of bits per pixel in the specified context, or `0` if the context is not a bitmap context.

Availability

Available in Mac OS X version 10.2 and later.

Declared In

`CGBitmapContext.h`

CGBitmapContextGetBytesPerRow

Returns the bytes per row of a bitmap context.

```
size_t CGBitmapContextGetBytesPerRow (
   CGContextRef c
);
```

Parameters

context

The bitmap context to examine.

Return Value

The number of bytes per row of the specified context, or `0` if the context is not a bitmap context.

Availability

Available in Mac OS X version 10.2 and later.

Declared In

`CGBitmapContext.h`

CGBitmapContextGetColorSpace

Returns the color space of a bitmap context.

```
CGColorSpaceRef CGBitmapContextGetColorSpace (
   CGContextRef c
);
```

Parameters

context

The bitmap context to examine.

Return Value

The color space of the specified context, or `NULL` if the context is not a bitmap context. You are responsible for retaining and releasing this object as necessary.

Availability

Available in Mac OS X version 10.2 and later.

Declared In

`CGBitmapContext.h`

CGBitmapContextGetData

Returns a pointer to the image data associated with a bitmap context.

```
void * CGBitmapContextGetData (
   CGContextRef c
);
```

Parameters

context

The bitmap context to examine.

Return Value
A pointer to the specified bitmap context's image data, or `NULL` if the context is not a bitmap context.

Availability
Available in Mac OS X version 10.2 and later.

Related Sample Code
CarbonSketch

Declared In
`CGBitmapContext.h`

CGBitmapContextGetHeight

Returns the height in pixels of a bitmap context.

```
size_t CGBitmapContextGetHeight (
   CGContextRef c
);
```

Parameters

context
The bitmap context to examine.

Return Value
The height in pixels of the specified context, or `0` if the context is not a bitmap context.

Availability
Available in Mac OS X version 10.2 and later.

Declared In
`CGBitmapContext.h`

CGBitmapContextGetWidth

Returns the width in pixels of a bitmap context.

```
size_t CGBitmapContextGetWidth (
   CGContextRef c
);
```

Parameters

context
The bitmap context to examine.

Return Value
The width in pixels of the specified context, or `0` if the context is not a bitmap context.

Availability
Available in Mac OS X version 10.2 and later.

Declared In
`CGBitmapContext.h`

Document Revision History

This table describes the changes to *CGBitmapContext Reference*.

Date	Notes
2007-06-28	Updated for Mac OS X v10.5.
	Revised the documentation for the function CGBitmapContextCreate (page 6).
2006-05-23	Revised the introduction.
2006-01-10	Changed CGImageBitmapInfo to CGBitmapInfo.
2005-07-07	Added information to CGBitmapContextCreate about indexed colorspaces.
2005-06-04	Added a link from CGBitmapContextCreate to Quartz 2D Programming Guide.
2005-04-29	Updated for Mac OS X v10.4.
	Added the functions CGBitmapContextGetBitmapInfo (page 8) and CGBitmapContextCreateImage (page 7).
	In the function CGBitmapContextCreate (page 6), changed the name of the alphaInfo parameter to bitmapInfo, and the data type for this parameter from CGAlphaInfo to CGBitmapInfo.
	Removed outdated pixel format table; added cross-references to conceptual documentation.
2004-02-26	First version of this document. An earlier version of this information appeared in *Quartz 2D Reference*.

Index