

---

# CAOpenGLLayer Class Reference

[Graphics & Imaging](#) > Quartz



2008-03-11



Apple Inc.  
© 2008 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, Mac, Mac OS, Objective-C, and Quartz are trademarks of Apple Inc., registered in the United States and other countries.

OpenGL is a registered trademark of Silicon Graphics, Inc.

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

---

## **CAOpenGLLayer Class Reference 5**

---

Overview 5

Tasks 5

    Drawing the Content 5

    Managing the Pixel Format 6

    Managing the Rendering Context 6

Properties 6

    asynchronous 6

Instance Methods 7

    canDrawInCGLContext:pixelFormat:forLayerTime:displayTime: 7

    copyCGLContextForPixelFormat: 7

    copyCGLPixelFormatForDisplayMask: 8

    drawInCGLContext:pixelFormat:forLayerTime:displayTime: 8

    isAsynchronous 9

    releaseCGLContext: 9

    releaseCGLPixelFormat: 9

---

## **Document Revision History 11**

---

## **Index 13**

---



# CAOpenGLLayer Class Reference

---

<b>Inherits from</b>	CALayer : NSObject
<b>Conforms to</b>	NSCoding (CALayer) CAMediaTiming (CALayer) NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/QuartzCore.framework
<b>Availability</b>	Available in Mac OS X v10.5 and later.
<b>Declared in</b>	CAOpenGLLayer.h
<b>Companion guides</b>	Core Animation Programming Guide Core Animation Cookbook
<b>Related sample code</b>	CALayerEssentials

## Overview

CAOpenGLLayer provides a layer suitable for rendering OpenGL content.

To provide OpenGL content you subclass CAOpenGLLayer and override [drawInCGLContext:pixelFormat:forLayerTime:displayTime:](#) (page 8). You can specify that the OpenGL content is static by setting the [asynchronous](#) (page 6) property to NO.

## Tasks

### Drawing the Content

- [asynchronous](#) (page 6) *property*  
Determines when the contents of the layer are updated.
- [isAsynchronous](#) (page 9)  
A synthesized accessor for the [asynchronous](#) (page 6) property.
- [canDrawInCGLContext:pixelFormat:forLayerTime:displayTime:](#) (page 7)  
Returns whether the receiver should draw OpenGL content for the specified time.
- [drawInCGLContext:pixelFormat:forLayerTime:displayTime:](#) (page 8)  
Draws the OpenGL content for the specified time.

## Managing the Pixel Format

- [copyCGLPixelFormatForDisplayMask:](#) (page 8)  
Returns the OpenGL pixel format suitable for rendering to the set of displays specified by the display mask.
- [releaseCGLPixelFormat:](#) (page 9)  
Releases the specified OpenGL pixel format object.

## Managing the Rendering Context

- [copyCGLContextForPixelFormat:](#) (page 7)  
Returns the rendering context the receiver requires for the specified pixel format.
- [releaseCGLContext:](#) (page 9)  
Releases the specified rendering context.

## Properties

For more about Objective-C properties, see “Properties” in *The Objective-C 2.0 Programming Language*.

### asynchronous

Determines when the contents of the layer are updated.

```
@property BOOL asynchronous
```

#### Discussion

If NO, the contents of the layer are updated only in response to receiving a `setNeedsDisplay` message. When YES, the receiver’s `canDrawInCGLContext:pixelFormat:forLayerTime:displayTime:` (page 7) is called periodically to determine if the OpenGL content should be updated.

#### Availability

Available in Mac OS X v10.5 and later.

#### See Also

- [isAsynchronous](#) (page 9)

#### Related Sample Code

CALayerEssentials

#### Declared In

CAOpenGLLayer.h

## Instance Methods

### canDrawInCGLContext:pixelFormat:forLayerTime:displayTime:

Returns whether the receiver should draw OpenGL content for the specified time.

```
- (BOOL)canDrawInCGLContext:(CGLContextObj)glContext
  pixelFormat:(CGLPixelFormatObj)pixelFormat
  forLayerTime:(CTimeInterval)timeInterval
  displayTime:(const CVTimeStamp *)timeStamp
```

#### Parameters

*glContext*

The CGLContextObj in to which the OpenGL content would be drawn.

*pixelFormat*

The pixel format used when the *glContext* was created.

*timeInterval*

The current layer time.

*timeStamp*

The display timestamp associated with *timeInterval*. Can be null.

#### Return Value

YES if the receiver should render OpenGL content, NO otherwise.

#### Discussion

This method is called before attempting to render the frame for the layer time specified by *timeInterval*. If the method returns NO, the frame is skipped. The default implementation always returns YES.

#### Availability

Available in Mac OS X v10.5 and later.

#### Declared In

CAOpenGLLayer.h

### copyCGLContextForPixelFormat:

Returns the rendering context the receiver requires for the specified pixel format.

```
- (CGLContextObj)copyCGLContextForPixelFormat:(CGLPixelFormatObj)pixelFormat
```

#### Parameters

*pixelFormat*

The pixel format for the rendering context.

#### Return Value

A new CGLContext with renderers for *pixelFormat*.

#### Discussion

This method is called when a rendering context is needed by the receiver. The default implementation allocates a new context with a null share context.

You should not call this method directly, it is intended to be overridden by subclasses.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CAOpenGLLayer.h

**copyCGLPixelFormatForDisplayMask:**

Returns the OpenGL pixel format suitable for rendering to the set of displays specified by the display mask.

```
- (CGLPixelFormatObj)copyCGLPixelFormatForDisplayMask:(uint32_t)mask
```

**Parameters**

*mask*

The display mask the OpenGL content will be rendered on.

**Discussion**

This method is called when a pixel format object is needed for the receiver. The default implementation returns a 32bpp fixed point pixel format, with the `NoRecovery` and `Accelerated` flags set.

You should not call this method directly, it is intended to be overridden by subclasses.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CAOpenGLLayer.h

**drawInCGLContext:pixelFormat:forLayerTime:displayTime:**

Draws the OpenGL content for the specified time.

```
- (void)drawInCGLContext:(CGLContextObj)glContext
    pixelFormat:(CGLPixelFormatObj)pixelFormat
    forLayerTime:(CFTimeInterval)timeInterval
    displayTime:(const CVTimeStamp *)timeStamp
```

**Parameters**

*glContext*

The rendering context in to which the OpenGL content should be rendered.

*pixelFormat*

The pixel format used when the *glContext* was created.

*timeInterval*

The current layer time.

*timeStamp*

The display timestamp associated with *timeInterval*. Can be null.

**Discussion**

This method is called when a new frame needs to be generated for the layer time specified by *timeInterval*. The viewport of *glContext* is set correctly for the size of the layer. No other state is defined. If the method enables OpenGL features, it should disable them before returning.



The default implementation of the method flushes the context.

#### Availability

Available in Mac OS X v10.5 and later.

#### Declared In

CAOpenGLLayer.h

## isAsynchronous

A synthesized accessor for the [asynchronous](#) (page 6) property.

- (BOOL)isAsynchronous

#### See Also

[@property asynchronous](#) (page 6)

## releaseCGLContext:

Releases the specified rendering context.

- (void)releaseCGLContext:(CGLContextObj)*glContext*

#### Parameters

*glContext*

The rendering context to release.

#### Discussion

This method is called when the OpenGL context that was previously returned by [copyCGLContextForPixelFormat:](#) (page 7) is no longer needed.

You should not call this method directly, it is intended to be overridden by subclasses.

#### Availability

Available in Mac OS X v10.5 and later.

#### Declared In

CAOpenGLLayer.h

## releaseCGLPixelFormat:

Releases the specified OpenGL pixel format object.

- (void)releaseCGLPixelFormat:(CGLPixelFormatObj)*pixelFormat*

#### Parameters

*pixelFormat*

The pixel format object to release.

#### Discussion

This method is called when the OpenGL pixel format that was previously returned by [copyCGLContextForPixelFormat:](#) (page 7).

You should not call this method directly, it is intended to be overridden by subclasses.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CAOpenGLLayer.h

# Document Revision History

---

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

Date	Notes
2008-03-11	Clarified that <code>copyCGLPixelFormatForDisplayMask</code> , <code>copyCGLContextForPixelFormat</code> , <code>releaseCGLContext</code> , and <code>releaseCGLPixelFormat</code> should only be implemented, never called.
2007-07-24	New document that describes the class that provides a context for rendering OpenGL in a layer.

## REVISION HISTORY

### Document Revision History

# Index

---

## A

---

asynchronous [instance property 6](#)

## C

---

canDrawInCGLContext:pixelFormat:forLayerTime:  
displayTime: [instance method 7](#)

copyCGLContextForPixelFormat: [instance method 7](#)

copyCGLPixelFormatForDisplayMask: [instance method 8](#)

## D

---

drawInCGLContext:pixelFormat:forLayerTime:  
displayTime: [instance method 8](#)

## I

---

isAsynchronous [instance method 9](#)

## R

---

releaseCGLContext: [instance method 9](#)

releaseCGLPixelFormat: [instance method 9](#)