

---

# AVPlayerLayer Class Reference

Audio & Video



2010-05-16



Apple Inc.  
© 2010 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, iPhone, and Objective-C are trademarks of Apple Inc., registered in the United States and other countries.

iOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license.

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

---

## **AVPlayerLayer Class Reference 5**

---

Overview	5
Tasks	5
Miscellaneous	5
Properties	6
player	6
readyForDisplay	6
videoGravity	6
Class Methods	7
playerLayerWithPlayer:	7

---

## **Document Revision History 9**

---



# AVPlayerLayer Class Reference

---

<b>Inherits from</b>	CALayer : NSObject
<b>Conforms to</b>	NSCoding (CALayer) CAMediaTiming (CALayer) NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/AVFoundation.framework
<b>Availability</b>	Available in iOS 4.0 and later.
<b>Declared in</b>	AVPlayerLayer.h

## Overview

`AVPlayerLayer` is a subclass of `CALayer` to which an `AVPlayer` object can direct its visual output.

You can create arbitrary numbers of player layers with the same `AVPlayer` object. You can create a layer as illustrated in the following code fragment:

```
AVPlayer *player = <#A configured AVPlayer object#>;

CALayer *superlayer = <#Get a CALayer#>;
AVPlayerLayer *playerLayer = [AVPlayerLayer playerLayerWithPlayer:player];
[superlayer addSublayer:playerLayer];
```

The value for the `contents` key of a player layer is opaque and effectively read-only.

During playback, `AVPlayer` may compensate for temporal drift between its visual output and its audible output to one or more independently-clocked audio output devices by adjusting the timing of its associated player layers. The effects of these adjustments are usually very small; however, clients that wish to remain entirely unaffected by such adjustments may wish to place other layers for which timing is important into independently timed subtrees of their layer trees.

## Tasks

### Miscellaneous

[player](#) (page 6) *property*

The player for which the player layer displays visual output.

+ [playerLayerWithPlayer:](#) (page 7)

Returns a player layer to display the visual output of a specified player.

[readyForDisplay](#) (page 6) *property*

Indicates whether the player is ready to be displayed. (read-only)

[videoGravity](#) (page 6) *property*

Specifies how the video is displayed within a player layer's bounds.

## Properties

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

### player

The player for which the player layer displays visual output.

```
@property(n nonatomic, retain) AVPlayer *player
```

#### Discussion

#### Availability

Available in iOS 4.0 and later.

#### Declared In

AVPlayerLayer.h

### readyForDisplay

Indicates whether the player is ready to be displayed. (read-only)

```
@property(n nonatomic, readonly, getter=isReadyForDisplay) BOOL readyForDisplay
```

#### Discussion

#### Availability

Available in iOS 4.0 and later.

#### Declared In

AVPlayerLayer.h

### videoGravity

Specifies how the video is displayed within a player layer's bounds.

```
@property(copy) NSString *videoGravity
```

#### Discussion

The default is [AVPlayerLayerVideoGravityResizeAspect](#) (page ?).

This property is animatable.

**Availability**

Available in iOS 4.0 and later.

**Declared In**

AVPlayerLayer.h

## Class Methods

### **playerLayerWithPlayer:**

Returns a player layer to display the visual output of a specified player.

```
+ (AVPlayerLayer *)playerLayerWithPlayer:(AVPlayer *)player
```

**Parameters**

*player*

The player for which the player layer displays visual output.

**Return Value**

A player layer configured to display the visual output of *player*.

**Discussion**

**Availability**

Available in iOS 4.0 and later.

**Declared In**

AVPlayerLayer.h





# Document Revision History

---

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

Date	Notes
2010-05-16	First version of a document that describes a Core Animation layer to which an AVPlayer object can direct its visual output.

## REVISION HISTORY

### Document Revision History