
QCComposition Class Reference

[Cocoa > Graphics & Imaging](#)



2007-06-26



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, Mac OS, and Quartz 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

QCComposition Class Reference 5

Overview	5
Tasks	6
Creating a Composition	6
Getting Information About a Composition	6
Getting Port Keys	6
Class Methods	6
compositionWithData:	6
compositionWithFile:	7
Instance Methods	7
attributes	7
identifier	7
inputKeys	8
outputKeys	8
protocols	8
Constants	9
Attribute Keys	9
Composition Categories	10
Standard Protocol Input Keys	10
Standard Protocol Output Keys	12
Standard Protocols	13

Document Revision History 15

Index 17

QCComposition Class Reference

Inherits from	NSObject
Conforms to	NSCopying NSObject (NSObject)
Framework	/System/Library/Frameworks/Quartz.framework/Frameworks/QuartzComposer.framework
Availability	Available in Mac OS X v10.5 and later.
Companion guide	Quartz Composer Programming Guide
Declared in	QCComposition.h QCCompositionRepository.h

Overview

The `QCComposition` class represents a Quartz Composer composition that either:

- comes from the system-wide composition repository (`/Library/Compositions` and `~/Library/Compositions`) where it can be accessed by any application through the methods of the `QCCompositionRepository` class
- is created from an arbitrary source (typically a file on disk) using one of its methods

This class cannot be subclassed.

A `QCComposition` object has the following information associated with it and that you can obtain by using the appropriate method of the `QCComposition` class:

- Attributes include the name and description of the composition, copyright information, and whether or not its provided by Mac OS X (built-in).
- The protocols that the composition conforms to. A **composition protocol** defines a set of required and optional input parameters and output results.

Many methods of the `QCRenderer`, `QCCompositionLayer`, and `QCView` classes take a `QCComposition` object as a parameter.

Tasks

Creating a Composition

- + `compositionWithFile:` (page 7)
Returns an autoreleased composition object initialized with a Quartz Composer composition file.
- + `compositionWithData:` (page 6)
Returns an autoreleased composition object initialized with the contents of a Quartz Composer composition file.

Getting Information About a Composition

- `attributes` (page 7)
Returns the attributes of the composition.
- `protocols` (page 8)
Returns the list of protocols to which the composition conforms.
- `identifier` (page 7)
Returns the unique and persistent identifier for the composition from the composition repository.

Getting Port Keys

- `inputKeys` (page 8)
Returns an array listing the keys that identify the input ports of the root patch of the composition.
- `outputKeys` (page 8)
Returns an array listing the keys that identify the output ports of the root patch of the composition.

Class Methods

compositionWithData:

Returns an autoreleased composition object initialized with the contents of a Quartz Composer composition file.

```
+ (QCComposition*) compositionWithData:(NSData*)data;
```

Parameters

data

The contents of a file created with the Quartz Composer developer tool.

Return Value

A Quartz Composer composition object or `nil` if there is an error.

Availability

Available in Mac OS X v10.5 and later.

Declared In

QCComposition.h

compositionWithFile:

Returns an autoreleased composition object initialized with a Quartz Composer composition file.

+ (QCComposition*) compositionWithFile:(NSString*)path;

Parameters*path*

A path to a file created with the Quartz Composer developer tool (.qtz extension).

Return ValueA Quartz Composer composition object or `nil` if there is an error.**Availability**

Available in Mac OS X v10.5 and later.

Declared In

QCComposition.h

Instance Methods

attributes

Returns the attributes of the composition.

- (NSDictionary*) attributes

Return ValueA dictionary of composition attributes. See “[Attribute Keys](#)” (page 9) for the attributes that can be returned.**Availability**

Available in Mac OS X v10.5 and later.

Declared In

QCComposition.h

identifier

Returns the unique and persistent identifier for the composition from the composition repository.

- (NSString*) identifier

Return ValueThe unique identifier for the composition if it comes from the composition repository; `nil` otherwise.**Availability**

Available in Mac OS X v10.5 and later.

Declared In

QCCompositionRepository.h

inputKeys

Returns an array listing the keys that identify the input ports of the root patch of the composition.

- (NSArray*) inputKeys

Return Value

An array of input keys.

Availability

Available in Mac OS X v10.5 and later.

Declared In

QCComposition.h

outputKeys

Returns an array listing the keys that identify the output ports of the root patch of the composition.

- (NSArray*) outputKeys

Return Value

An array of output keys.

Availability

Available in Mac OS X v10.5 and later.

Declared In

QCComposition.h

protocols

Returns the list of protocols to which the composition conforms.

- (NSArray*) protocols

Return Value

A list of protocols. See [“Standard Protocols”](#) (page 13).

Availability

Available in Mac OS X v10.5 and later.

Declared In

QCComposition.h

Constants

Attribute Keys

Attributes of a composition.

```
extern NSString* const QCCompositionAttributeNameKey;
extern NSString* const QCCompositionAttributeDescriptionKey;
extern NSString* const QCCompositionAttributeCopyrightKey;
extern NSString* const QCCompositionAttributeBuiltInKey;
extern NSString* const QCCompositionAttributeTimeDependentKey;
extern NSString* const QCCompositionAttributeHasConsumersKey;
extern NSString* const QCCompositionAttributeCategoryKey;
```

Constants

`QCCompositionAttributeNameKey`

The key for the composition name. The associated value is an `NSString` object.

Available in Mac OS X v10.4 and later.

Declared in `QCComposition.h`.

`QCCompositionAttributeDescriptionKey`

The key for the composition description. The associated value is an `NSString` object.

Available in Mac OS X v10.4 and later.

Declared in `QCComposition.h`.

`QCCompositionAttributeCopyrightKey`

The key for composition copyright information. The associated value is an `NSString` object.

Available in Mac OS X v10.4 and later.

Declared in `QCComposition.h`.

`QCCompositionAttributeBuiltInKey`

The key for the composition origin. The associated value is an `NSNumber` object that contains a Boolean value. YES indicates the composition is built-in (provided by Mac OS X).

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionAttributeTimeDependentKey`

The key for the composition time dependency. The associated value is an `NSNumber` object that contains a Boolean value. YES indicates that the composition is time dependent.

`QCCompositionAttributeHasConsumersKey`

The key for a composition that has consumer patches. The associated value is an `NSNumber` object that contains a Boolean value. YES indicates that the composition has consumers.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionAttributeCategoryKey`

The composition category. The associated value is a category constant. See [“Composition Categories”](#) (page 10).

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

Declared In

QCComposition.h

Composition Categories

Categories for compositions.

```
extern NSString* const QCCompositionCategoryDistortion;
extern NSString* const QCCompositionCategoryStylize;
extern NSString* const QCCompositionCategoryUtility;
```

Constants

QCCompositionCategoryDistortion
A composition that produces a distortion effect.
 Available in Mac OS X v10.5 and later.
 Declared in QCComposition.h.

QCCompositionCategoryStylize
A composition that produces a stylize effect.
 Available in Mac OS X v10.5 and later.
 Declared in QCComposition.h.

QCCompositionCategoryUtility
A utility composition.
 Available in Mac OS X v10.5 and later.
 Declared in QCComposition.h.

Declared In

QCComposition.h

Standard Protocol Input Keys

Input ports of a composition.

```
extern NSString* const QCCompositionInputImageKey;
extern NSString* const QCCompositionInputSourceImageKey;
extern NSString* const QCCompositionInputDestinationImageKey;
extern NSString* const QCCompositionInputRSSFeedURLKey;
extern NSString* const QCCompositionInputRSSArticleDurationKey;
extern NSString* const QCCompositionInputPreviewModeKey;
extern NSString* const QCCompositionInputXKey;
extern NSString* const QCCompositionInputYKey;
extern NSString* const QCCompositionInputScreenImageKey;
extern NSString* const QCCompositionInputAudioPeakKey;
extern NSString* const QCCompositionInputAudioSpectrumKey;
extern NSString* const QCCompositionInputTrackPositionKey;
extern NSString* const QCCompositionInputTrackInfoKey;
extern NSString* const QCCompositionInputTrackSignalKey;
extern NSString* const QCCompositionInputPrimaryColorKey;
extern NSString* const QCCompositionInputSecondaryColorKey;
extern NSString* const QCCompositionInputPaceKey;
```

Constants

`QCCompositionInputImageKey`

An image input port whose key is `inputImage`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputSourceImageKey`

An image input port whose key is `inputSourceImage`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputDestinationImageKey`

An image input port whose key is `inputDestinationImage`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputRSSFeedURLKey`

A string input port whose key is `inputRSSFeedURL`. This port must be passed an http or feed scheme URL.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputRSSArticleDurationKey`

A number input port whose key is `inputRSSArticleDuration`. The value must be expressed in seconds.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputPreviewModeKey`

A Boolean input port whose key is `inputPreviewMode`. When the value of this input port is set to `TRUE`, the composition that provides this port must be able to run in a low-quality mode that produces a preview of the composition.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputXKey`

A number input port whose key is `inputX`. The value must be normalized to the image width with the origin on the left.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputYKey`

A number input port whose key is `inputY`. The value must be normalized to the image height with the origin at the bottom.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputScreenImageKey`

An image input port whose key is `inputScreenImage`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputAudioPeakKey`

A number input port whose key is `inputAudioPeak`. The value must be in the `[0, 1]` range as a mono signal with no decay applied.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputAudioSpectrumKey`

A structure input port whose key is `inputAudioSpectrum`. The structure must contain 16 values in the `[0, 1]` range representing 16 spectrum bands of the mono signal from low to high frequencies with no decay applied.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputTrackPositionKey`

A number input port whose key is `inputTrackPosition`. The value must be expressed in seconds.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputTrackInfoKey`

A structure input port whose key is `inputTrackInfo`. The structure contains optional entries, such as "name", "artist", "album", "duration", "artwork", and so on.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputTrackSignalKey`

A Boolean input port whose key is `inputTrackSignal`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputPrimaryColorKey`

A color input port whose key is `inputPrimaryColor`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputSecondaryColorKey`

A color input port whose key is `inputSecondaryColor`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionInputPaceKey`

A number input port whose key is `inputPace`. The value must be in the `[0, 1]` range.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

Declared In

`QCComposition.h`

Standard Protocol Output Keys

Output ports of a composition.

```
extern NSString* const QCCompositionOutputImageKey;
extern NSString* const QCCompositionOutputWebPageURLKey;
```

Constants

`QCCompositionOutputImageKey`

An image output port whose key is `outputImage`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionOutputWebPageURLKey`

A string output port whose key is `outputWebPageURL`.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

Declared In

`QCComposition.h`

Standard Protocols

Protocols for a composition.

```
extern NSString* const QCCompositionProtocolGraphicAnimation;
extern NSString* const QCCompositionProtocolGraphicTransition;
extern NSString* const QCCompositionProtocolImageFilter;
extern NSString* const QCCompositionProtocolImageCompositor;
extern NSString* const QCCompositionProtocolImageTransition;
extern NSString* const QCCompositionProtocolScreenSaverRSS;
```

Constants

`QCCompositionProtocolGraphicAnimation`

A composition that renders a generic graphical animation. It has the option to use [QCCompositionInputPrimaryColorKey](#) (page 12) for the primary color of the animation, [QCCompositionInputSecondaryColorKey](#) (page 12) for the secondary color of the animation, [QCCompositionInputPaceKey](#) (page 12) for the global pace of the animation, and [QCCompositionInputPreviewModeKey](#) (page 11) to indicate if the animation should run in lower-quality for preview purposes.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionProtocolGraphicTransition`

A composition that performs a transition between two images, using a transition time in range of 0 to 1. A conforming composition must use the input keys

[QCCompositionInputSourceImageKey](#) (page 11) for the starting image and

[QCCompositionInputDestinationImageKey](#) (page 11) for the image to transition to. The composition can optionally use [QCCompositionInputPreviewModeKey](#) (page 11) to indicate if the animation should run in lower-quality for preview purposes.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionProtocolImageFilter`

A composition that applies an effect to a source image. A conforming composition must use the input key `QCCompositionInputImageKey` (page 11) for the source image and `QCCompositionOutputImageKey` (page 13) for the output image. The composition can optionally use `QCCompositionInputXKey` (page 11) to specify the X position of the center point of the effect, `QCCompositionInputYKey` (page 11) to specify the Y position of the center point of the effect, and `QCCompositionInputPreviewModeKey` (page 11) to indicate if the animation should run in lower-quality for preview purposes.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionProtocolScreenSaver`

A composition that can be used as a screen saver. The composition has the option to use `QCCompositionInputScreenImageKey` (page 11) for a screenshot image of the screen that the screen saver runs on, `QCCompositionInputPreviewModeKey` (page 11) to indicate if the animation should run in lower-quality for preview purposes, and `QCCompositionOutputWebPageURLKey` (page 13) for a URL to open in the default web browser when screen saver exits (only allowed if screen saver password is disabled).

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionProtocolImageTransition`

A composition that performs a transition between two images, using a parametric time value to drives the transition from start (at time 0) to end (at time 1). A conforming composition must use the input keys `QCCompositionInputImageKey` (page 11) for the starting image and `QCCompositionInputDestinationImageKey` (page 11) for the ending image. The composition can optionally use `QCCompositionInputPreviewModeKey` (page 11) to indicate if the animation should run in lower-quality for preview purposes.

`QCCompositionProtocolRSSVisualizer`

A composition that acts as a visualizer for an RSS feed. A conforming composition must use the input key `QCCompositionInputRSSFeedURLKey` (page 11) for the URL to use for the RSS feed. It can optionally use `QCCompositionInputRSSArticleDurationKey` (page 11) to specify the duration of each feed article.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

`QCCompositionProtocolMusicVisualizer`

A composition that acts as a visualizer for music. A conforming composition must use the input key `QCCompositionInputAudioPeakKey` (page 12) for the instantaneous audio peak and the `QCCompositionInputAudioSpectrumKey` (page 12) for the instantaneous audio spectrum. It can optionally use the `QCCompositionInputTrackInfoKey` (page 12) to indicate it receives information about the current track and the `QCCompositionInputTrackSignalKey` (page 12) to indicate the start of a new track.

Available in Mac OS X v10.5 and later.

Declared in `QCComposition.h`.

Declared In

`QCComposition.h`

Document Revision History

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

Date	Notes
2007-06-26	New document that describes the class that represents a Quartz Composer composition.

REVISION HISTORY

Document Revision History

Index

A

Attribute Keys [9](#)

`attributes` **instance method** [7](#)

C

Composition Categories [10](#)

`compositionWithData:` **class method** [6](#)

`compositionWithFile:` **class method** [7](#)

I

`identifier` **instance method** [7](#)

`inputKeys` **instance method** [8](#)

O

`outputKeys` **instance method** [8](#)

P

`protocols` **instance method** [8](#)

Q

`QCCompositionAttributeBuiltInKey` **constant** [9](#)

`QCCompositionAttributeCategoryKey` **constant** [9](#)

`QCCompositionAttributeCopyrightKey` **constant** [9](#)

`QCCompositionAttributeDescriptionKey` **constant** [9](#)

`QCCompositionAttributeHasConsumersKey` **constant** [9](#)

`QCCompositionAttributeNameKey` **constant** [9](#)

`QCCompositionAttributeTimeDependentKey` **constant** [9](#)

`QCCompositionCategoryDistortion` **constant** [10](#)

`QCCompositionCategoryStylize` **constant** [10](#)

`QCCompositionCategoryUtility` **constant** [10](#)

`QCCompositionInputAudioPeakKey` **constant** [12](#)

`QCCompositionInputAudioSpectrumKey` **constant** [12](#)

`QCCompositionInputDestinationImageKey` **constant** [11](#)

`QCCompositionInputImageKey` **constant** [11](#)

`QCCompositionInputPaceKey` **constant** [12](#)

`QCCompositionInputPreviewModeKey` **constant** [11](#)

`QCCompositionInputPrimaryColorKey` **constant** [12](#)

`QCCompositionInputRSSArticleDurationKey` **constant** [11](#)

`QCCompositionInputRSSFeedURLKey` **constant** [11](#)

`QCCompositionInputScreenImageKey` **constant** [11](#)

`QCCompositionInputSecondaryColorKey` **constant** [12](#)

`QCCompositionInputSourceImageKey` **constant** [11](#)

`QCCompositionInputTrackInfoKey` **constant** [12](#)

`QCCompositionInputTrackPositionKey` **constant** [12](#)

`QCCompositionInputTrackSignalKey` **constant** [12](#)

`QCCompositionInputXKey` **constant** [11](#)

`QCCompositionInputYKey` **constant** [11](#)

`QCCompositionOutputImageKey` **constant** [13](#)

`QCCompositionOutputWebPageURLKey` **constant** [13](#)

`QCCompositionProtocolGraphicAnimation` **constant** [13](#)

`QCCompositionProtocolGraphicTransition` **constant** [13](#)

`QCCompositionProtocolImageFilter` **constant** [14](#)

`QCCompositionProtocolImageTransition` **constant** [14](#)

`QCCompositionProtocolMusicVisualizer` **constant** [14](#)

`QCCompositionProtocolRSSVisualizer` **constant** [14](#)

`QCCompositionProtocolScreenSaver` **constant** [14](#)

S

- Standard Protocol Input Keys [10](#)
- Standard Protocol Output Keys [12](#)
- Standard Protocols [13](#)