
CIPlugIn Class Reference

[Cocoa](#) > [Graphics & Imaging](#)



2006-12-05



Apple Inc.
© 2006 Apple Computer, 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

CIPlugIn Class Reference 5

Overview 5

Tasks 5

 Loading Plug-ins 5

Class Methods 6

 loadAllPlugIns 6

 loadNonExecutablePlugIns 6

 loadPlugIn:allowNonExecutable: 6

Document Revision History 9

Index 11

CIPlugIn Class Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	Library/Frameworks/QuartzCore.framework
Availability	Mac OS X v10.4 and later
Declared in	CIPlugIn.h
Companion guides	Image Unit Tutorial Core Image Programming Guide
Related sample code	CIAnnotation CIVideoDemoGL

Overview

The `CIPlugIn` class loads image units. An image unit is an image processing bundle that contains one or more Core Image filters. The `.plugin` extension indicates one or more filters that are packaged as an image unit.

Tasks

Loading Plug-ins

+ [loadAllPlugIns](#) (page 6)

Scans directories for files that have the `.plugin` extension and then loads the image units.

+ [loadNonExecutablePlugIns](#) (page 6)

Scans directories for files that have the `.plugin` extension and then loads only those filters that are marked by the image unit as non-executable filters.

+ [loadPlugIn:allowNonExecutable:](#) (page 6)

Loads filters from an image unit that have the appropriate executable status.

Class Methods

loadAllPlugins

Scans directories for files that have the `.plugin` extension and then loads the image units.

```
+ (void)loadAllPlugins
```

Discussion

This method scans the following directories:

- `/Library/Graphics/Image Units`
- `~/Library/Graphics/Image Units`

Call this method once. If you call this method more than once, Core Image loads newly added image units, but image units (and the filters they contain) that are already loaded are not removed.

Availability

Mac OS X v10.4 and later.

Related Sample Code

CIAnnotation

CIVideoDemoGL

Declared In

CIPlugin.h

loadNonExecutablePlugins

Scans directories for files that have the `.plugin` extension and then loads only those filters that are marked by the image unit as non-executable filters.

```
+ (void)loadNonExecutablePlugins
```

Discussion

This call does not execute any of the code in the image unit, it simply loads the code. You need to call this method only once to load a specific image unit. The behavior of this method is not defined for multiple calls for the same image unit.

Availability

Mac OS X v10.4 and later.

Declared In

CIPlugin.h

loadPlugin:allowNonExecutable:

Loads filters from an image unit that have the appropriate executable status.

```
+ (void)loadPlugin:(NSURL *)url allowNonExecutable:(BOOL)allowNonExecutable
```

Parameters

url

The location of the image unit to load.

allowNonExecutable

TRUE to load only those filters that are marked by the image unit as non-executable filters.

Discussion

You need to call this method only once to load a specific image unit. The behavior of this method is not defined for multiple calls for the same image unit.

Availability

Mac OS X v10.4 and later.

Related Sample Code

CIAnnotation

Declared In

CIPlugin.h

Document Revision History

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

Date	Notes
2006-12-05	Added link to image unit tutorial.
2006-05-23	First publication of this content as a separate document.
	Added parameter descriptions and updated Class Description.

REVISION HISTORY

Document Revision History

Index

L

loadAllPlugIns **class method** [6](#)
loadNonExecutablePlugIns **class method** [6](#)
loadPlugIn:allowNonExecutable: **class method** [6](#)