Quick Look Framework Reference

Data Management: File Management

2010-04-19

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Simultaneously published in the United States and Canada.

Contents

Introduction

Framework	/System/Library/Frameworks/QuickLook.framework
Header file directories	/System/Library/Frameworks/QuickLook.framework/Headers
Declared in	QLPreviewController.h QLPreviewItem.h

Use the Quick Look framework to provide previews of items that are in formats you don't handle—such as iWork or Microsoft Office. This framework affords you more control over the preview process than you get from the `UIDocumentInteractionController` class—including choosing whether the preview is displayed in the context of a navigation controller or modally (full screen). The primary class in this framework is `QLPreviewController`, which provides a specialized view for previewing an item. It relies on a delegate for mediating preview actions, and a data source for providing the preview items.

If you want to take advantage of the options-menu feature, which lets a user pick an application to open a previewed item with, use the `UIDocumentInteractionController` class instead.

Introduction

Classes

PART I

Classes

QLPreviewController Class Reference

Inherits from	UIViewController : UIResponder : NSObject
Conforms to	UIDocumentInteractionControllerDelegate NSCoding (UIViewController) NSObject (NSObject)
Framework	/System/Library/Frameworks/QuickLook.framework
Availability	Available in iOS 4.0 and later.
Declared in	

Overview

A `QLPreviewController` object, or Quick Look preview controller, provides a specialized view for previewing an item. To display a Quick Look preview controller you have two options: You can push it into view using a `UINavigationController` object, or can present it modally, full screen, using the `presentModalViewController:animated:` method of its parent class, `UIViewController`.

A displayed preview includes a title taken from the last path component of the item URL. You can override this by implementing a `previewItemTitle` (page 24) accessor for the preview item.

A Quick Look preview controller can display previews for the following items:

- iWork documents
- Microsoft Office documents (Office '97 and newer)
- Rich Text Format (RTF) documents
- PDF files
- Images
- Text files whose uniform type identifier (UTI) conforms to the `public.text` type (see *Uniform Type Identifiers Reference*)
- Comma-separated value (csv) files

To use a Quick Look preview controller, you must provide a data source object. The data source provides preview items to the controller and tells it how many items to include in a preview navigation list. If there is more than one item in the list, a modally-presented (that is, full-screen) controller displays navigation arrows to let the user switch among the items. For a Quick Look preview controller pushed using a navigation controller, you can provide buttons in the navigation bar for moving through the navigation list.

For details on providing items to a preview controller, refer to *QLPreviewControllerDataSource Protocol Reference* and *QLPreviewItem Protocol Reference*.

Tasks

Configuring a Quick Look Preview Controller

`dataSource` (page 11) *property*
The Quick Look preview controller's data source.

`delegate` (page 11) *property*
The Quick Look preview controller's delegate object.

Managing Item Previews

`+ canPreviewItem:` (page 12)
Indicates whether or not the Quick Look preview controller can display an item.

`currentPreviewItem` (page 10) *property*
The item currently displayed in the Quick Look preview controller. (read-only)

`currentPreviewItemIndex` (page 11) *property*
The index, within the preview item navigation list, of the item currently displayed in the Quick Look preview controller.

`- refreshCurrentPreviewItem` (page 12)
Asks the Quick Look preview controller to recompute the display of the current preview item.

`- reloadData` (page 12)
Asks the Preview Controller to reload its data from its data source.

Properties

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

currentPreviewItem

The item currently displayed in the Quick Look preview controller. (read-only)

```
@property (readonly) id<QLPreviewItem> currentPreviewItem
```

Discussion
If no item is being displayed, this property's value is `nil`.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewController.h`

currentPreviewItemIndex

The index, within the preview item navigation list, of the item currently displayed in the Quick Look preview controller.

```
@property NSInteger currentPreviewItemIndex
```

Discussion
You can change which item is displayed, among those in a navigation list, by setting this property's value. If no item is being displayed, this property's value is `NSNotFound`.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewController.h`

dataSource

The Quick Look preview controller's data source.

```
@property (assign) id<QLPreviewControllerDataSource> dataSource
```

Discussion
To use a Quick Look preview controller, you must implement a data source. The data source is responsible for providing items for display by the controller, and for telling it how many items to include in the preview navigation list. It is described in*QLPreviewControllerDataSource Protocol Reference*.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewController.h`

delegate

The Quick Look preview controller's delegate object.

```
@property (assign) id delegate
```

Discussion
The delegate determines whether or not to open URLs that the user taps in a preview. See *QLPreviewControllerDelegate Protocol Reference*.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewController.h`

Class Methods

canPreviewItem:

Indicates whether or not the Quick Look preview controller can display an item.

```
+ (BOOL) canPreviewItem: (id < QLPreviewItem >) item
```

Parameters

item

An item to be previewed.

Return Value

Returns `YES` if the Quick Look preview controller can display the specified preview item.

Discussion

If an item cannot be displayed, but you still attempt to display it, a Quick Look preview controller displays a generic error. Always check if an item can be displayed before choosing whether or not to display it.

Refer to this document's Overview section (QLPreviewController (page 9)) for the types of items that a Quick Look preview controller can display.

Availability

Available in iOS 4.0 and later.

Declared In

`QLPreviewController.h`

Instance Methods

refreshCurrentPreviewItem

Asks the Quick Look preview controller to recompute the display of the current preview item.

```
- (void) refreshCurrentPreviewItem
```

Discussion

The display is recomputed whether or not the current preview item has changed.

Availability

Available in iOS 4.0 and later.

Declared In

`QLPreviewController.h`

reloadData

Asks the Preview Controller to reload its data from its data source.

```
- (void) reloadData
```

Discussion
The display is recomputed only if the current preview item has changed.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewController.h`

Protocols

PART II

Protocols

QLPreviewControllerDataSource Protocol Reference

Framework /System/Library/Frameworks/QuickLook.framework

Availability Available in iOS 4.0 and later.

Declared in

Overview

The data source for a `QLPreviewController` (Quick Look preview controller) object must adopt this protocol to enable it to provide preview items to the controller.

This protocol also lets the data source tell the controller how many items to include in a preview item navigation list.

This protocol's methods are required.

Tasks

Providing Data to a Quick Look Preview Controller

- `numberOfPreviewItemsInPreviewController:` (page 17) *required method*
  Invoked when the Quick Look preview controller needs to know the number of preview items to include in the preview navigation list. (required)
- `previewController:previewItemAtIndex:` (page 18) *required method*
  Invoked when the Quick Look preview controller needs the preview item for a specified index position. (required)

Instance Methods

numberOfPreviewItemsInPreviewController:

Invoked when the Quick Look preview controller needs to know the number of preview items to include in the preview navigation list. (required)

```
- (NSInteger) numberOfPreviewItemsInPreviewController: (QLPreviewController *)
    controller;
```

Parameters

controller

The Quick Look preview controller that is requesting the number of preview items.

Return Value

The number of items that the Quick Look preview controller should include in its preview navigation list.

Discussion

If you push a Quick Look preview controller into view using a `UINavigationController` object, it is up to you to provide a means for the user to navigate among the items in a navigation list. For example, you could add buttons to the navigation bar, using them to set the `currentPreviewItemIndex` property to indicate the item you want to display.

If you display a preview controller modally (full screen), the controller includes navigation arrows if there is more than one item in the navigation list.

Availability

Available in iOS 4.0 and later.

Declared In

`QLPreviewController.h`

previewController:previewItemAtIndex:

Invoked when the Quick Look preview controller needs the preview item for a specified index position. (required)

```
- (id <QLPreviewItem>) previewController: (QLPreviewController *) controller
    previewItemAtIndex: (NSInteger) index;
```

Parameters

controller

The Quick Look preview controller that is requesting a preview item.

index

The index position, within the preview navigation list, of the item to preview.

Return Value

The preview item to display. The item must be an `NSURL` object, or a custom object that conforms to the `QLPreviewItem` protocol.

Availability

Available in iOS 4.0 and later.

Declared In

`QLPreviewController.h`

QLPreviewControllerDelegate Protocol Reference

Conforms to	NSObject
Framework	/System/Library/Frameworks/QuickLook.framework
Availability	Available in iOS 4.0 and later.
Declared in	

Overview

Implement the methods of this protocol in the delegate of a `QLPreviewController` (Quick Look preview controller) object to:

- Specify whether or not your application opens a URL that the user taps in a preview
- Respond to the opening or closing of a preview

The delegate of a `QLPreviewController` object must adopt this protocol. The methods described here are optional but expected.

Tasks

Responding to Preview Requests

`- previewControllerWillDismiss:` (page 21)
Invoked before the preview controller is closed.

`- previewControllerDidDismiss:` (page 20)
Invoked after the preview controller is closed.

Responding to User Actions

`- previewController:shouldOpenURL:forPreviewItem:` (page 20)
Invoked by the Quick Look preview controller before trying to open a URL.

Instance Methods

previewController:shouldOpenURL:forPreviewItem:

Invoked by the Quick Look preview controller before trying to open a URL.

```
- (BOOL) previewController: (QLPreviewController *) controller shouldOpenURL: (NSURL
    *) url forPreviewItem: (id <QLPreviewItem>) item;
```

Parameters

controller

The Quick Look preview controller that is asking the delegate to handle a user tap on a URL.

url

The URL, from the displayed preview, that the user tapped.

item

The item being displayed in the preview.

Return Value
A Boolean value indicating whether or not the URL indicated in the *url* parameter should be opened.

Discussion
This method comes into play when the user taps a URL link in a preview. If you return `YES`, the Quick Look preview controller invokes the `openURL:` method on the `UIApplication` object, sending it the value of the *url* parameter. If you return `NO`, the `openURL:` method is not invoked.

If you do not implement this method, it defaults to returning `YES`.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewController.h`

previewControllerDidDismiss:

Invoked after the preview controller is closed.

```
- (void) previewControllerDidDismiss: (QLPreviewController *) controller;
```

Parameters

controller

The Quick Look preview controller that just closed.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewController.h`

previewControllerWillDismiss:

Invoked before the preview controller is closed.

```
- (void) previewControllerWillDismiss: (QLPreviewController *) controller;
```

Parameters

controller

The Quick Look preview controller that is about to close.

Availability

Available in iOS 4.0 and later.

Declared In

QLPreviewController.h

QLPreviewItem Protocol Reference

Conforms to	NSObject
Framework	/System/Library/Frameworks/QuickLook.framework
Availability	Available in iOS 4.0 and later.
Declared in	

Overview

A Quick Look preview item must adopt the `QLPreviewItem` protocol to enable it to be displayed by a `QLPreviewController` (Quick Look preview controller) object.

The methods in this protocol are also declared as a category on the `NSURL` class. As a result, you can use `NSURL` objects directly as preview items—provided that you want to use the default titles of those items. A default title is the last path component of an item's URL. If you want to supply your own preview item titles, create your own preview item objects that adopt this protocol.

Tasks

Providing an Item URL

`previewItemURL` (page 24) *required property*
The URL of the item to preview. (required)

Providing an Item Title

`previewItemTitle` (page 24) *property*
The preview item's title.

Properties

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

previewItemTitle

The preview item's title.

```
@property (readonly) NSString *previewItemTitle;
```

Discussion
This property is used by a Quick Look preview controller to get an item's title. In typical use, you would implement a getter method in your preview item class to provide this value.

If not `nil`, this value is used as the preview item title, replacing the default item title. The default title is the last path component of an item's URL.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewItem.h`

previewItemURL

The URL of the item to preview. (required)

```
@property (readonly) NSURL *previewItemURL;
```

Discussion
This property is used by a Quick Look preview controller to get an item's URL. In typical use, you would implement a getter method in your preview item class to provide this value.

The value of this property must be a file-type URL.

If the item is not available for preview, this property's getter method should return `nil`. In this case, the Quick Look preview controller displays a "loading" view.

Availability
Available in iOS 4.0 and later.

Declared In
`QLPreviewItem.h`

Document Revision History

This table describes the changes to *Quick Look Framework Reference*.

Date	Notes
2010-04-19	New document that describes the API for presenting item previews in iOS.

Document Revision History