

---

# NSPrintOperation Class Reference

[Cocoa > Printing](#)



2007-03-02



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, and Mac OS 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

## NSPrintOperation Class Reference 5

---

Overview	5
Tasks	6
Creating an NSPrintOperation	6
Setting the Current NSPrintOperation for This Thread	6
Determining the Type of Operation	6
Modifying the NSPrintInfo Object	7
Getting the NSView Object	7
Running a Print Operation	7
Modifying the User Interface	7
Managing the Drawing Context	8
Managing Page Information	8
Managing Printing-Related Threads	8
Deprecated Methods	8
Class Methods	9
currentOperation	9
EPSOperationWithView:insideRect:toData:	9
EPSOperationWithView:insideRect:toData:printInfo:	10
EPSOperationWithView:insideRect:toPath:printInfo:	11
PDFOperationWithView:insideRect:toData:	11
PDFOperationWithView:insideRect:toData:printInfo:	12
PDFOperationWithView:insideRect:toPath:printInfo:	13
printOperationWithView:	13
printOperationWithView:printInfo:	14
setCurrentOperation:	15
Instance Methods	15
canSpawnSeparateThread	15
cleanUpOperation	16
context	16
createContext	16
currentPage	17
deliverResult	17
destroyContext	17
isCopyingOperation	18
jobTitle	18
pageOrder	18
pageRange	19
printInfo	19
printPanel	20
runOperation	20
runOperationModalForWindow:delegate:didRunSelector:contextInfo:	21

- setCanSpawnSeparateThread: 21
- setJobTitle: 22
- setPageOrder: 23
- setPrintInfo: 23
- setPrintPanel: 23
- setShowsPrintPanel: 24
- setShowsProgressPanel: 24
- showsPrintPanel 25
- showsProgressPanel 25
- view 26
- Constants 26
  - NSPrintingPageOrder 26
  - Exception Name 27

**Appendix A      [Deprecated NSPrintOperation Methods 29](#)**

---

- Deprecated in Mac OS X v10.4 and later 29
  - setShowPanels: 29
  - showPanels 30
- Deprecated in Mac OS X v10.5 30
  - accessoryView 30
  - jobStyleHint 31
  - setAccessoryView: 31
  - setJobStyleHint: 32

**[Document Revision History 33](#)**

---

**[Index 35](#)**

---

# NSPrintOperation Class Reference

---

<b>Inherits from</b>	NSObject
<b>Conforms to</b>	NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/AppKit.framework
<b>Availability</b>	Available in Mac OS X v10.0 and later.
<b>Companion guide</b>	Printing Programming Topics for Cocoa
<b>Declared in</b>	NSPrintOperation.h
<b>Related sample code</b>	QTKitPlayer Quartz Composer WWDC 2005 TextEdit Quartz2DBasics Sketch-112 TextEditPlus

## Overview

An `NSPrintOperation` object controls operations that generate Encapsulated PostScript (EPS) code, Portable Document Format (PDF) code, or print jobs. An `NSPrintOperation` object works in conjunction with two other objects: an `NSPrintInfo` object, which specifies how the code should be generated, and an `NSView` object, which generates the actual code.

It is important to note that the majority of methods in `NSPrintOperation` copy the instance of `NSPrintInfo` passed into them. Future changes to that print info are not reflected in the print info retained by the current `NSPrintOperation` object. All changes should be made to the print info before passing to the methods of this class. The only method in `NSPrintOperation` which does not copy the `NSPrintInfo` instance is [setPrintInfo:](#) (page 23).

**Note:** You should not subclass `NSPrintOperation`. Methods that return an `NSPrintOperation` object return an instance of a concrete subclass whose implementation is private.

## Tasks

### Creating an NSPrintOperation

- + [EPSOperationWithView:insideRect:toData:](#) (page 9)  
Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view.
- + [EPSOperationWithView:insideRect:toData:printInfo:](#) (page 10)  
Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view using the specified print settings.
- + [EPSOperationWithView:insideRect:toPath:printInfo:](#) (page 11)  
Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view and write the resulting data to the specified file.
- + [PDFOperationWithView:insideRect:toData:](#) (page 11)  
Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view.
- + [PDFOperationWithView:insideRect:toData:printInfo:](#) (page 12)  
Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view using the specified print settings.
- + [PDFOperationWithView:insideRect:toPath:printInfo:](#) (page 13)  
Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view and write the resulting data to the specified file.
- + [printOperationWithView:](#) (page 13)  
Creates and returns an `NSPrintOperation` object ready to control the printing of the specified view.
- + [printOperationWithView:printInfo:](#) (page 14)  
Creates and returns an `NSPrintOperation` object ready to control the printing of the specified view using custom print settings.

### Setting the Current NSPrintOperation for This Thread

- + [currentOperation](#) (page 9)  
Returns the current print operation for this thread.
- + [setCurrentOperation:](#) (page 15)  
Sets the current print operation for this thread.

### Determining the Type of Operation

- [isCopyingOperation](#) (page 18)  
Returns a Boolean value indicating whether the receiver is an EPS or PDF copy operation.

## Modifying the NSPrintInfo Object

- [printInfo](#) (page 19)  
Returns the receiver's `NSPrintInfo` object.
- [setPrintInfo:](#) (page 23)  
Sets the receiver's `NSPrintInfo` object.

## Getting the NSView Object

- [view](#) (page 26)  
Returns the view object that generates the actual data for the print operation.

## Running a Print Operation

- [runOperation](#) (page 20)  
Runs the print operation on the current thread.
- [runOperationModalForWindow:delegate:didRunSelector:contextInfo:](#) (page 21)  
Runs the print operation, calling your custom delegate method upon completion.
- [cleanUpOperation](#) (page 16)  
Called at the end of a print operation to remove the receiver as the current operation.
- [deliverResult](#) (page 17)  
Delivers the results of the print operation to the intended destination.

## Modifying the User Interface

- [showsPrintPanel](#) (page 25)  
Returns a Boolean value indicating whether a print panel is displayed during the operation,
- [setShowsPrintPanel:](#) (page 24)  
Sets whether the receiver displays a print panel for this operation.
- [showsProgressPanel](#) (page 25)  
Returns a Boolean value indicating whether a progress panel is displayed during the operation.
- [setShowsProgressPanel:](#) (page 24)  
Sets whether the receiver displays a progress panel for this operation.
- [jobTitle](#) (page 18)  
Returns the title of the print job.
- [setJobTitle:](#) (page 22)  
Assigns a custom title to the print job.
- [printPanel](#) (page 20)  
Returns the `NSPrintPanel` object used when running the operation.
- [setPrintPanel:](#) (page 23)  
Sets the `NSPrintPanel` object to be used during the operation.

## Managing the Drawing Context

- [context](#) (page 16)  
Returns the graphics context object used for generating output.
- [createContext](#) (page 16)  
Creates the graphics context object used for drawing during the operation.
- [destroyContext](#) (page 17)  
Destroys the receiver's graphics context.

## Managing Page Information

- [currentPage](#) (page 17)  
Returns the current page number being printed.
- [pageRange](#) (page 19)  
Returns the range of pages associated with the print operation.
- [pageOrder](#) (page 18)  
Returns the print order for the pages.
- [setPageOrder:](#) (page 23)  
Sets the print order for the pages of the operation.

## Managing Printing-Related Threads

- [canSpawnSeparateThread](#) (page 15)  
Returns a Boolean value indicating whether the receiver is allowed to spawn a separate printing thread.
- [setCanSpawnSeparateThread:](#) (page 21)  
Sets whether the receiver is allowed to spawn a separate printing thread.

## Deprecated Methods

- [setShowPanels:](#) (page 29) **Deprecated in Mac OS X v10.4 and later**  
Sets whether the print operation should display a print panel. (**Deprecated.** Use [setShowsPrintPanel:](#) (page 24) and [setShowsProgressPanel:](#) (page 24) instead.)
- [showPanels](#) (page 30) **Deprecated in Mac OS X v10.4 and later**  
Returns a Boolean value that indicates whether the print panel is to be displayed. (**Deprecated.** Use [showsPrintPanel](#) (page 25) and [showsProgressPanel](#) (page 25) instead.)
- [accessoryView](#) (page 30) **Deprecated in Mac OS X v10.5**  
Returns the accessory view used by the receiver's print panel. (**Deprecated.** Use the `accessoryControllers` method of `NSPrintPanel` instead.)
- [jobStyleHint](#) (page 31) **Deprecated in Mac OS X v10.5**  
Returns the type of content that the print job is printing. (**Deprecated.** Use the `jobStyleHint` method of `NSPrintPanel` instead.)



- [setAccessoryView:](#) (page 31) **Deprecated in Mac OS X v10.5**  
Sets the custom accessory view to be displayed by the receiver's print panel. (**Deprecated.** Use the `addAccessoryController:` method of `NSPrintPanel` instead.)
- [setJobStyleHint:](#) (page 32) **Deprecated in Mac OS X v10.5**  
Sets the type of content that the print job is printing. (**Deprecated.** Use the `setJobStyleHint:` method of `NSPrintPanel` instead.)

## Class Methods

### currentOperation

Returns the current print operation for this thread.

```
+ (NSPrintOperation *)currentOperation
```

#### Return Value

The print operation object, or `nil` if there is no current operation.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

+ [setCurrentOperation:](#) (page 15)

#### Related Sample Code

Quartz2DBasics

#### Declared In

`NSPrintOperation.h`

### EPSOperationWithView:insideRect:toData:

Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view.

```
+ (NSPrintOperation *)EPSOperationWithView:(NSView *)aView insideRect:(NSRect)rect
  toData:(NSMutableData *)data
```

#### Parameters

*aView*

The view containing the data to be turned into EPS data.

*rect*

The portion of the view (specified in points in the view's coordinate space) to be rendered as EPS data.

*data*

An empty `NSMutableData` object. After the job is run, this object contains the EPS data.

#### Return Value

The new `NSPrintOperation` object. You must run the operation to generate the EPS data.

**Discussion**

The new `NSPrintOperation` object uses the default `NSPrintInfo` object. This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- + [EPSOperationWithView:insideRect:toData:printInfo:](#) (page 10)
- + [EPSOperationWithView:insideRect:toPath:printInfo:](#) (page 11)
- [runOperation](#) (page 20)

**Declared In**

`NSPrintOperation.h`

**EPSOperationWithView:insideRect:toData:printInfo:**

Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view using the specified print settings.

```
+ (NSPrintOperation *)EPSOperationWithView:(NSView *)aView insideRect:(NSRect)rect
  toData:(NSMutableData *)data printInfo:(NSPrintInfo *)aPrintInfo
```

**Parameters**

*aView*

The view containing the data to be turned into EPS data.

*rect*

The portion of the view (specified in points in the view's coordinate space) to be rendered as EPS data.

*data*

An empty `NSMutableData` object. After the job is run, this object contains the EPS data.

*aPrintInfo*

The print settings to use when generating the EPS data.

**Return Value**

The new `NSPrintOperation` object. You must run the operation to generate the EPS data.

**Discussion**

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- + [EPSOperationWithView:insideRect:toData:](#) (page 9)
- + [EPSOperationWithView:insideRect:toPath:printInfo:](#) (page 11)
- [runOperation](#) (page 20)

**Declared In**

`NSPrintOperation.h`

## EPSOperationWithView:insideRect:toPath:printInfo:

Creates and returns a new `NSPrintOperation` object ready to control the copying of EPS graphics from the specified view and write the resulting data to the specified file.

```
+ (NSPrintOperation *)EPSOperationWithView:(NSView *)aView insideRect:(NSRect)rect
  toPath:(NSString *)path printInfo:(NSPrintInfo *)aPrintInfo
```

### Parameters

*aView*

The view containing the data to be turned into EPS data.

*rect*

The portion of the view (specified in points in the view's coordinate space) to be rendered as EPS data.

*path*

The path to a file. After the job is run, this file contains the EPS data.

*aPrintInfo*

The print settings to use when generating the EPS data.

### Return Value

The new `NSPrintOperation` object. You must run the operation to generate the EPS data.

### Discussion

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- + [EPSOperationWithView:insideRect:toData:](#) (page 9)
- + [EPSOperationWithView:insideRect:toData:printInfo:](#) (page 10)
- [runOperation](#) (page 20)

### Declared In

`NSPrintOperation.h`

## PDFOperationWithView:insideRect:toData:

Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view.

```
+ (NSPrintOperation *)PDFOperationWithView:(NSView *)aView insideRect:(NSRect)rect
  toData:(NSMutableData *)data
```

### Parameters

*aView*

The view containing the data to be turned into PDF data.

*rect*

The portion of the view (specified in points in the view's coordinate space) to be rendered as PDF data.

*data*

An empty `NSMutableData` object. After the job is run, this object contains the PDF data.

#### Return Value

The new `NSPrintOperation` object. You must run the operation to generate the PDF data.

#### Discussion

The new `NSPrintOperation` object uses the default `NSPrintInfo` object. This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

+ [PDFOperationWithView:insideRect:toData:printInfo:](#) (page 12)

+ [PDFOperationWithView:insideRect:toPath:printInfo:](#) (page 13)

- [runOperation](#) (page 20)

#### Declared In

`NSPrintOperation.h`

## PDFOperationWithView:insideRect:toData:printInfo:

Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view using the specified print settings.

```
+ (NSPrintOperation *)PDFOperationWithView:(NSView *)aView insideRect:(NSRect)rect
  toData:(NSMutableData *)data printInfo:(NSPrintInfo *)aPrintInfo
```

#### Parameters

*aView*

The view containing the data to be turned into PDF data.

*rect*

The portion of the view (specified in points in the view's coordinate space) to be rendered as PDF data.

*data*

An empty `NSMutableData` object. After the job is run, this object contains the PDF data.

*aPrintInfo*

The print settings to use when generating the PDF data.

#### Return Value

The new `NSPrintOperation` object. You must run the operation to generate the PDF data.

#### Discussion

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

+ [PDFOperationWithView:insideRect:toData:](#) (page 11)

- + [PDFOperationWithView:insideRect:toPath:printInfo:](#) (page 13)
- [runOperation](#) (page 20)

**Declared In**

NSPrintOperation.h

**PDFOperationWithView:insideRect:toPath:printInfo:**

Creates and returns a new `NSPrintOperation` object ready to control the copying of PDF graphics from the specified view and write the resulting data to the specified file.

```
+ (NSPrintOperation *)PDFOperationWithView:(NSView *)aView insideRect:(NSRect)rect
  toPath:(NSString *)path printInfo:(NSPrintInfo *)aPrintInfo
```

**Parameters***aView*

The view containing the data to be turned into PDF data.

*rect*

The portion of the view (specified in points in the view's coordinate space) to be rendered as PDF data.

*path*

The path to a file. After the job is run, this file contains the PDF data.

*aPrintInfo*

The print settings to use when generating the PDF data.

**Return Value**

The new `NSPrintOperation` object. You must run the operation to generate the PDF data.

**Discussion**

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- + [PDFOperationWithView:insideRect:toData:](#) (page 11)
- + [PDFOperationWithView:insideRect:toData:printInfo:](#) (page 12)
- [runOperation](#) (page 20)

**Declared In**

NSPrintOperation.h

**printOperationWithView:**

Creates and returns an `NSPrintOperation` object ready to control the printing of the specified view.

```
+ (NSPrintOperation *)printOperationWithView:(NSView *)aView
```

**Parameters***aView*

The view whose contents you want to print.

**Return Value**

The new `NSPrintOperation` object. You must run the operation to print the view.

**Discussion**

The new `NSPrintOperation` object uses the settings stored in the shared `NSPrintInfo` object. This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [printOperationWithView:printInfo:](#) (page 14)

- [runOperation](#) (page 20)

**Related Sample Code**

Quartz2DBasics

ToolbarSample

**Declared In**

`NSPrintOperation.h`

**printOperationWithView:printInfo:**

Creates and returns an `NSPrintOperation` object ready to control the printing of the specified view using custom print settings.

```
+ (NSPrintOperation *)printOperationWithView:(NSView *)aView printInfo:(NSPrintInfo *)aPrintInfo
```

**Parameters***aView*

The view whose contents you want to print.

*aPrintInfo*

The print settings to use when printing the view.

**Return Value**

The new `NSPrintOperation` object. You must run the operation to print the view.

**Discussion**

This method raises an `NSPrintOperationExistsException` if there is already a print operation in progress; otherwise the returned object is made the current print operation for this thread.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [printOperationWithView:](#) (page 13)

- [runOperation](#) (page 20)

**Related Sample Code**

QTAudioExtractionPanel

QTKitPlayer

Quartz Composer WWDC 2005 TextEdit

Sketch-112

TextEditPlus

**Declared In**

NSPrintOperation.h

**setCurrentOperation:**

Sets the current print operation for this thread.

+ (void)setCurrentOperation:(NSPrintOperation \*)*operation***Parameters***operation*The print operation to make current. You may specify `nil` to clear the current print operation.**Availability**

Available in Mac OS X v10.0 and later.

**See Also**+ [currentOperation](#) (page 9)**Declared In**

NSPrintOperation.h

## Instance Methods

**canSpawnSeparateThread**

Returns a Boolean value indicating whether the receiver is allowed to spawn a separate printing thread.

- (BOOL)canSpawnSeparateThread

**Return Value**

YES if the receiver is allowed to spawn a separate thread; otherwise, NO.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [setCanSpawnSeparateThread:](#) (page 21)**Declared In**

NSPrintOperation.h

## cleanUpOperation

Called at the end of a print operation to remove the receiver as the current operation.

- (void)cleanUpOperation

### Discussion

You typically do not invoke this method directly.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [runOperation](#) (page 20)

### Declared In

NSPrintOperation.h

## context

Returns the graphics context object used for generating output.

- (NSGraphicsContext \*)context

### Return Value

The graphics context object used for drawing during the operation.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [createContext](#) (page 16)

- [destroyContext](#) (page 17)

### Declared In

NSPrintOperation.h

## createContext

Creates the graphics context object used for drawing during the operation.

- (NSGraphicsContext \*)createContext

### Return Value

The graphics context object used for drawing. This object is created using the settings from the receiver's `NSPrintInfo` object.

### Discussion

Do not invoke this method directly—it is invoked before any output is generated.

### Availability

Available in Mac OS X v10.0 and later.



**See Also**

- [context](#) (page 16)
- [destroyContext](#) (page 17)

**Declared In**

NSPrintOperation.h

## currentPage

Returns the current page number being printed.

- (NSInteger)currentPage

**Return Value**

The current page being printed.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [pageOrder](#) (page 18)
- [setPageOrder:](#) (page 23)

**Declared In**

NSPrintOperation.h

## deliverResult

Delivers the results of the print operation to the intended destination.

- (BOOL)deliverResult

**Return Value**

YES if the results were successfully delivered; otherwise, NO.

**Discussion**

This method may be called to deliver the results to the printer spool or preview application. Do not invoke this method directly—it is invoked automatically when the print operation is done.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [runOperation](#) (page 20)

**Declared In**

NSPrintOperation.h

## destroyContext

Destroys the receiver's graphics context.

- (void)destroyContext

#### Discussion

Do not invoke this method directly—it is invoked at the end of a print operation.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

- [context](#) (page 16)
- [createContext](#) (page 16)

#### Declared In

NSPrintOperation.h

## isCopyingOperation

Returns a Boolean value indicating whether the receiver is an EPS or PDF copy operation.

- (BOOL)isCopyingOperation

#### Return Value

YES if the receiver is an EPS or PDF copy operation; otherwise, NO.

#### Availability

Available in Mac OS X v10.0 and later.

#### Declared In

NSPrintOperation.h

## jobTitle

Returns the title of the print job.

- (NSString \*)jobTitle

#### Return Value

A string containing the print job title. If set, this value overrides the title returned by the printing view.

#### Availability

Available in Mac OS X v10.5 and later.

#### See Also

- [setJobTitle:](#) (page 22)
- [printJobTitle](#) (NSView)

#### Declared In

NSPrintOperation.h

## pageOrder

Returns the print order for the pages.

- (NSPrintingPageOrder)pageOrder

**Return Value**

The print order. For a list of possible values, see “Constants” (page 26).

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [currentPage](#) (page 17)

**Declared In**

NSPrintOperation.h

## pageRange

Returns the range of pages associated with the print operation.

- (NSRange)pageRange

**Return Value**

The range of page numbers. Page numbers are one-based values where the index of page one is 1, the index of page two is 2, and so on. Depending on the information returned by the printing view, the starting page number may not be 1. Also, if the number of pages being printed is not known, the page count may be set to NSIntegerMax.

**Availability**

Available in Mac OS X v10.5 and later.

**See Also**

- [knowsPageRange:](#) (NSView)

**Declared In**

NSPrintOperation.h

## printInfo

Returns the receiver’s NSPrintInfo object.

- (NSPrintInfo \*)printInfo

**Return Value**

The print settings of the receiver.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [setPrintInfo:](#) (page 23)

**Related Sample Code**

Quartz2DBasics

Sketch-112

**Declared In**

NSPrintOperation.h

**printPanel**Returns the `NSPrintPanel` object used when running the operation.- (`NSPrintPanel *`)`printPanel`**Return Value**

The print panel object for the operation.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [accessoryView](#) (page 30)
- [setAccessoryView:](#) (page 31)
- [setPrintPanel:](#) (page 23)
- [setShowPanels:](#) (page 29)
- [showPanels](#) (page 30)

**Declared In**

NSPrintOperation.h

**runOperation**

Runs the print operation on the current thread.

- (`BOOL`)`runOperation`**Return Value**

YES if the operation was successful; otherwise, NO.

**Discussion**

The operation runs to completion in the current thread, blocking the application. A separate thread is not spawned, even if [canSpawnSeparateThread](#) (page 15) is YES. Use [runOperationModalForWindow:delegate:didRunSelector:contextInfo:](#) (page 21) to use document-modal sheets and to allow a separate thread to perform the operation.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [cleanupOperation](#) (page 16)
- [deliverResult](#) (page 17)

**Related Sample Code**

Quartz Composer WWDC 2005 TextEdit

Quartz2DBasics

TextEditPlus

**Declared In**

NSPrintOperation.h

**runOperationModalForWindow:delegate:didRunSelector:contextInfo:**

Runs the print operation, calling your custom delegate method upon completion.

```
- (void)runOperationModalForWindow:(NSWindow *)docWindow delegate:(id)delegate
    didRunSelector:(SEL)didRunSelector contextInfo:(void *)contextInfo
```

**Parameters***docWindow*

The document window to receive a print progress sheet.

*delegate*

The printing delegate object. Messages are sent to this object.

*didRunSelector*

The delegate method called after the completion of the print operation.

*contextInfo*A pointer to any data you want passed to the method in the *didRunSelector* parameter.**Discussion**The method specified by the *didRunSelector* parameter must have the following signature:

```
- (void)printOperationDidRun:(NSPrintOperation *)printOperation
    success:(BOOL)success contextInfo:(void *)contextInfo
```

The value of *success* is YES if the print operation ran to completion without cancellation or error, and NO otherwise.

If you send [setCanSpawnSeparateThread:](#) (page 21) to an `NSPrintOperation` object with an argument of YES, then the delegate specified in a subsequent invocation of [runOperationModalForWindow:delegate:didRunSelector:contextInfo:](#) (page 21) may be messaged in that spawned, non-main thread.

**Availability**

Available in Mac OS X v10.0 and later.

**Related Sample Code**

Quartz Composer WWDC 2005 TextEdit

TextEditPlus

ToolbarSample

**Declared In**

NSPrintOperation.h

**setCanSpawnSeparateThread:**

Sets whether the receiver is allowed to spawn a separate printing thread.

```
- (void)setCanSpawnSeparateThread:(BOOL)canSpawnSeparateThread
```

**Parameters***canSpawnSeparateThread*

YES if the receiver is allowed to spawn a separate thread; otherwise, NO.

**Discussion**

If *canSpawnSeparateThread* is YES, an `NSThread` object is detached when the print panel is dismissed (or immediately, if the panel is not to be displayed). The new thread performs the print operation, so that control can return to your application. A thread is detached only if the print operation is run using the [runOperationModalForWindow:delegate:didRunSelector:contextInfo:](#) (page 21) method. If *canSpawnSeparateThread* is NO, the operation runs on the current thread, blocking the application until the operation completes.

If you send [setCanSpawnSeparateThread:](#) (page 21) to an `NSPrintOperation` object with an argument of YES, then the delegate specified in a subsequent invocation of [runOperationModalForWindow:delegate:didRunSelector:contextInfo:](#) (page 21) may be messaged in that spawned, non-main thread.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [canSpawnSeparateThread](#) (page 15)**Related Sample Code**

Sketch-112

**Declared In**`NSPrintOperation.h`**setJobTitle:**

Assigns a custom title to the print job.

- (void)setJobTitle:(NSString \*)*jobTitle***Parameters***jobTitle*

The print job title. The receiver makes its own copy of the specified string.

**Discussion**

Assigning a title with this method overrides the job title provided by the printing view's `printJobTitle` method. Specifying `nil` for the *jobTitle* parameter causes the receiver to once again take its title from the printing view.

**Availability**

Available in Mac OS X v10.5 and later.

**See Also**

- [jobTitle](#) (page 18)  
- [printJobTitle \(NSView\)](#)

**Declared In**`NSPrintOperation.h`

**setPageOrder:**

Sets the print order for the pages of the operation.

- (void)setPageOrder:(NSPrintingPageOrder)*order*

**Parameters**

*order*

The print order. For a list of possible values, see “[Constants](#)” (page 26).

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [currentPage](#) (page 17)
- [pageOrder](#) (page 18)

**Declared In**

NSPrintOperation.h

**setPrintInfo:**

Sets the receiver's NSPrintInfo object.

- (void)setPrintInfo:(NSPrintInfo \*)*aPrintInfo*

**Parameters**

*aPrintInfo*

The new print settings for the receiver.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [printInfo](#) (page 19)

**Declared In**

NSPrintOperation.h

**setPrintPanel:**

Sets the NSPrintPanel object to be used during the operation.

- (void)setPrintPanel:(NSPrintPanel \*)*panel*

**Parameters**

*panel*

The print panel object to use for the operation.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [accessoryView](#) (page 30)

- [printPanel](#) (page 20)
- [setAccessoryView:](#) (page 31)
- [setShowPanels:](#) (page 29)
- [showPanels](#) (page 30)

**Declared In**

NSPrintOperation.h

**setShowsPrintPanel:**

Sets whether the receiver displays a print panel for this operation.

```
- (void)setShowsPrintPanel:(BOOL)flag
```

**Parameters***flag*

YES if you want to display a print panel; otherwise, NO.

**Discussion**

This method does not affect the display of a progress panel; that operation is controlled by the [setShowsProgressPanel:](#) (page 24) method.

Operations that generate EPS or PDF data do not display a progress panel, regardless of the value in the *flag* parameter.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

- [setShowsProgressPanel:](#) (page 24)
- [showsPrintPanel](#) (page 25)

**Declared In**

NSPrintOperation.h

**setShowsProgressPanel:**

Sets whether the receiver displays a progress panel for this operation.

```
- (void)setShowsProgressPanel:(BOOL)flag
```

**Parameters***flag*

YES if you want to display a progress panel; otherwise, NO.

**Discussion**

This method does not affect the display of a print panel; that operation is controlled by the [setShowsPrintPanel:](#) (page 24) method.

Operations that generate EPS or PDF data do not display a progress panel, regardless of the value in the *flag* parameter.



#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [setShowsPrintPanel](#): (page 24)
- [showsProgressPanel](#) (page 25)

#### Declared In

NSPrintOperation.h

## showsPrintPanel

Returns a Boolean value indicating whether a print panel is displayed during the operation,

- (BOOL)showsPrintPanel

#### Return Value

YES if the operation displays a print panel; otherwise, NO.

#### Discussion

Operations that generate EPS or PDF data do not display a print panel (instance of `NSPrintPanel`), regardless of the value returned by this method.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [setShowsPrintPanel](#): (page 24)

#### Declared In

NSPrintOperation.h

## showsProgressPanel

Returns a Boolean value indicating whether a progress panel is displayed during the operation.

- (BOOL)showsProgressPanel

#### Return Value

YES if the operation displays a progress panel; otherwise, NO.

#### Discussion

Operations that generate EPS or PDF data do not display a progress panel, regardless of the value returned by this method.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [setShowsProgressPanel](#): (page 24)

#### Declared In

NSPrintOperation.h

**view**

Returns the view object that generates the actual data for the print operation.

- (NSView \*)view

**Return Value**

The view object that generates the data.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSPrintOperation.h

## Constants

**NSPrintingPageOrder**

These constants specify the page order.

```
typedef enum _NSPrintingPageOrder {
    NSDescendingPageOrder = (-1),
    NSSpecialPageOrder    = 0,
    NSAscendingPageOrder  = 1,
    NSUnknownPageOrder    = 2
} NSPrintingPageOrder;
```

**Constants**

NSAscendingPageOrder

Ascending (back to front) page order.

Available in Mac OS X v10.0 and later.

Declared in NSPrintOperation.h.

NSDescendingPageOrder

Descending (front to back) page order.

Available in Mac OS X v10.0 and later.

Declared in NSPrintOperation.h.

NSSpecialPageOrder

The spooler does not rearrange pages—they are printed in the order received by the spooler.

Available in Mac OS X v10.0 and later.

Declared in NSPrintOperation.h.

NSUnknownPageOrder

No page order specified.

Available in Mac OS X v10.0 and later.

Declared in NSPrintOperation.h.

**Discussion**

These constants are used by [pageOrder](#) (page 18) and [setPageOrder:](#) (page 23).

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSPrintOperation.h

**Exception Name**

This is the name of an exception that can be raised by NSPrintOperation.

```
NSString *NSPrintOperationExistsException;
```

**Constants**

NSPrintOperationExistsException

The name of an exception raised when there is already a print operation in process.

The methods that raise this exception are the `EPSOperation...` and `printOperation...`

Available in Mac OS X v10.0 and later.

Declared in NSPrintOperation.h.

**Declared In**

NSPrintOperation.h



# Deprecated NSPrintOperation Methods

---

A method identified as deprecated has been superseded and may become unsupported in the future.

## Deprecated in Mac OS X v10.4 and later

### setShowPanels:

Sets whether the print operation should display a print panel. (Deprecated in Mac OS X v10.4 and later. Use [setShowsPrintPanel:](#) (page 24) and [setShowsProgressPanel:](#) (page 24) instead.)

- (void)setShowPanels:(BOOL)flag

#### Parameters

*flag*

YES if the print operation should display a print panel; otherwise, NO.

#### Discussion

This method also affects whether a progress panel is presented while the operation runs. If an EPS or PDF copy operation is being performed, neither panel is displayed, regardless of the value of *flag*.

#### Availability

Deprecated in Mac OS X v10.4 and later.

#### See Also

- [accessoryView](#) (page 30)
- [printPanel](#) (page 20)
- [setAccessoryView:](#) (page 31)
- [setPrintPanel:](#) (page 23)
- [showPanels](#) (page 30)

#### Related Sample Code

QTAudioExtractionPanel  
 QTKitImport  
 QTKitPlayer  
 Quartz Composer WWDC 2005 TextEdit  
 TextEditPlus

#### Declared In

NSPrintOperation.h

## showPanels

Returns a Boolean value that indicates whether the print panel is to be displayed. (Deprecated in Mac OS X v10.4 and later. Use [showsPrintPanel](#) (page 25) and [showsProgressPanel](#) (page 25) instead.)

- (BOOL)showPanels

### Return Value

YES if the print panel is to be displayed; otherwise, NO.

### Availability

Deprecated in Mac OS X v10.4 and later.

### See Also

- [accessoryView](#) (page 30)
- [printPanel](#) (page 20)
- [setAccessoryView:](#) (page 31)
- [setPrintPanel:](#) (page 23)
- [setShowPanels:](#) (page 29)

### Declared In

NSPrintOperation.h

## Deprecated in Mac OS X v10.5

## accessoryView

Returns the accessory view used by the receiver's print panel. (Deprecated in Mac OS X v10.5. Use the [accessoryControllers](#) method of [NSPrintPanel](#) instead.)

- (NSView \*)accessoryView

### Return Value

The custom accessory view.

### Discussion

You use the [setAccessoryView:](#) (page 31) method to customize the default [NSPrintPanel](#) object without having to subclass [NSPrintPanel](#) or specify your own print panel object.

### Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.5.

### See Also

- [printPanel](#) (page 20)
- [setPrintPanel:](#) (page 23)
- [setShowPanels:](#) (page 29)
- [showPanels](#) (page 30)

### Declared In

NSPrintOperation.h

## jobStyleHint

Returns the type of content that the print job is printing. (Deprecated in Mac OS X v10.5. Use the `jobStyleHint` method of `NSPrintPanel` instead.)

- (NSString \*)jobStyleHint

### Return Value

The content description, or `nil` if no job style hint has been set.

### Availability

Available in Mac OS X v10.2 and later.

Deprecated in Mac OS X v10.5.

### See Also

- [setJobStyleHint:](#) (page 32)

### Declared In

`NSPrintOperation.h`

## setAccessoryView:

Sets the custom accessory view to be displayed by the receiver's print panel. (Deprecated in Mac OS X v10.5. Use the `addAccessoryController:` method of `NSPrintPanel` instead.)

- (void)setAccessoryView:(NSView \*)aView

### Parameters

*aView*

The view to display in the print panel. You can use this view to specify additional print options.

### Discussion

You can use this method to avoid subclassing `NSPrintPanel` or specifying your own print panel object. The print panel is automatically resized (as needed) to accommodate the accessory view when it is selected.

### Availability

Available in Mac OS X v10.0 and later.

Deprecated in Mac OS X v10.5.

### See Also

- [accessoryView](#) (page 30)

- [printPanel](#) (page 20)

- [setPrintPanel:](#) (page 23)

- [setShowPanels:](#) (page 29)

- [showPanels](#) (page 30)

### Declared In

`NSPrintOperation.h`

## setJobStyleHint:

Sets the type of content that the print job is printing. (Deprecated in Mac OS X v10.5. Use the `setJobStyleHint:` method of `NSPrintPanel` instead.)

```
- (void)setJobStyleHint:(NSString *)hint
```

### Parameters

*hint*

A supported job style hint. Valid values for this parameter are described in the “Constants” section of the `NSPrintPanel` class. If this value is `nil`, the standard interface is used.

### Discussion

This controls the set of items that appear in the Presets menu of the simplified Print panel interface presented by this operation, if it presents one.

### Availability

Available in Mac OS X v10.2 and later.

Deprecated in Mac OS X v10.5.

### See Also

- [jobStyleHint](#) (page 31)

### Declared In

`NSPrintOperation.h`



# Document Revision History

---

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

Date	Notes
2007-03-02	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History

# Index

---

## A

---

accessoryView **instance method** [30](#)

## C

---

canSpawnSeparateThread **instance method** [15](#)

cleanupOperation **instance method** [16](#)

context **instance method** [16](#)

createContext **instance method** [16](#)

currentOperation **class method** [9](#)

currentPage **instance method** [17](#)

## D

---

deliverResult **instance method** [17](#)

destroyContext **instance method** [17](#)

## E

---

EPSOperationWithView:insideRect:toData: **class method** [9](#)

EPSOperationWithView:insideRect:toData:printInfo: **class method** [10](#)

EPSOperationWithView:insideRect:toPath:printInfo: **class method** [11](#)

Exception Name [27](#)

## I

---

isCopyingOperation **instance method** [18](#)

## J

---

jobStyleHint **instance method** [31](#)

jobTitle **instance method** [18](#)

## N

---

NSAscendingPageOrder **constant** [26](#)

NSDescendingPageOrder **constant** [26](#)

NSPrintingPageOrder **data type** [26](#)

NSPrintOperationExistsException **constant** [27](#)

NSSpecialPageOrder **constant** [26](#)

NSUnknownPageOrder **constant** [26](#)

## P

---

pageOrder **instance method** [18](#)

pageRange **instance method** [19](#)

PDFOperationWithView:insideRect:toData: **class method** [11](#)

PDFOperationWithView:insideRect:toData:printInfo: **class method** [12](#)

PDFOperationWithView:insideRect:toPath:printInfo: **class method** [13](#)

printInfo **instance method** [19](#)

printOperationWithView: **class method** [13](#)

printOperationWithView:printInfo: **class method** [14](#)

printPanel **instance method** [20](#)

## R

---

runOperation **instance method** [20](#)

runOperationModalForWindow:delegate:didRunSelector:contextInfo: **instance method** [21](#)

## S

---

setAccessoryView: **instance method** [31](#)  
setCanSpawnSeparateThread: **instance method** [21](#)  
setCurrentOperation: **class method** [15](#)  
setJobStyleHint: **instance method** [32](#)  
setJobTitle: **instance method** [22](#)  
setPageOrder: **instance method** [23](#)  
setPrintInfo: **instance method** [23](#)  
setPrintPanel: **instance method** [23](#)  
setShowPanels: **instance method** [29](#)  
setShowsPrintPanel: **instance method** [24](#)  
setShowsProgressPanel: **instance method** [24](#)  
showPanels **instance method** [30](#)  
showsPrintPanel **instance method** [25](#)  
showsProgressPanel **instance method** [25](#)

## V

---

view **instance method** [26](#)