
NSPasteboard Class Reference

[Cocoa](#) > [Interapplication Communication](#)



2008-10-15



Apple Inc.
© 2008 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, Quartz, and QuickDraw are trademarks of Apple Inc., registered in the United States and other countries.

Spotlight is a trademark of Apple Inc.

OpenGL is a registered trademark of Silicon Graphics, Inc.

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

NSPasteboard Class Reference 5

Overview	5
Tasks	5
Creating and Releasing a Pasteboard	5
Getting a Pasteboard's Name	6
Writing Data	6
Reading Data	7
Determining Types	7
Noticing Changed Ownership	7
Class Methods	7
generalPasteboard	7
pasteboardByFilteringData:ofType:	8
pasteboardByFilteringFile:	8
pasteboardByFilteringTypesInPasteboard:	9
pasteboardWithName:	9
pasteboardWithUniqueName	10
typesFilterableTo:	10
Instance Methods	11
addTypes:owner:	11
availableTypeFromArray:	12
changeCount	12
dataForType:	13
declareTypes:owner:	14
name	15
propertyListForType:	15
readFileContentsType:toFile:	16
readFileWrapper	16
releaseGlobally	17
setData:forType:	17
setPropertyList:forType:	18
setString:forType:	18
stringForType:	19
types	20
writeFileContents:	20
writeFileWrapper:	21
Delegate Methods	22
pasteboard:provideDataForType:	22
pasteboardChangedOwner:	22
Constants	23
Pasteboard Names	23
NSPasteboard Types for Standard Data	24

Document Revision History 27

Index 29

NSPasteboard Class Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.0 and later.
Declared in	NSPasteboard.h
Companion guides	Pasteboard Programming Topics for Cocoa Drag and Drop Programming Topics for Cocoa System Services
Related sample code	iSpend JDragNDrop Quartz Composer WWDC 2005 TextEdit Sketch-112 TextEditPlus

Overview

`NSPasteboard` objects transfer data to and from the pasteboard server. The server is shared by all running applications. It contains data that the user has cut or copied, as well as other data that one application wants to transfer to another. `NSPasteboard` objects are an application's sole interface to the server and to all pasteboard operations.

The drag pasteboard (`NSDragPboard`) is used to transfer data that is being dragged by the user.

An `NSPasteboard` object is also used to transfer data between applications and service providers listed in each application's Services menu.

Tasks

Creating and Releasing a Pasteboard

+ [generalPasteboard](#) (page 7)

Returns the general `NSPasteboard` object.

- + [pasteboardByFilteringData:ofType:](#) (page 8)
Creates and returns a new pasteboard with a unique name that supplies the specified data in as many types as possible given the available filter services.
- + [pasteboardByFilteringFile:](#) (page 8)
Creates and returns a new pasteboard with a unique name that supplies the specified file data in as many types as possible given the available filter services.
- + [pasteboardByFilteringTypesInPasteboard:](#) (page 9)
Creates and returns a new pasteboard with a unique name that supplies the specified pasteboard data in as many types as possible given the available filter services.
- + [pasteboardWithName:](#) (page 9)
Returns the pasteboard with the specified name.
- + [pasteboardWithUniqueName](#) (page 10)
Creates and returns a new pasteboard with a name that is guaranteed to be unique with respect to other pasteboards on the computer.
- + [typesFilterableTo:](#) (page 10)
Returns the data types that can be converted to the specified type using the available filter services.
- [releaseGlobally](#) (page 17)
Releases the receiver's resources in the pasteboard server.

Getting a Pasteboard's Name

- [name](#) (page 15)
Returns the receiver's name.

Writing Data

- [addTypes:owner:](#) (page 11)
Updates the list of declared data types in the receiver
- [declareTypes:owner:](#) (page 14)
Prepares the receiver for a change in its contents by declaring the new types of data it will contain and a new owner.
- [setData:forType:](#) (page 17)
Writes data of the specified type to the pasteboard server.
- [setPropertyList:forType:](#) (page 18)
Writes the specified property list to the pasteboard server.
- [setString:forType:](#) (page 18)
Writes the specified string to the pasteboard server.
- [writeFileContents:](#) (page 20)
Writes the contents of the specified file to the pasteboard.
- [writeFileWrapper:](#) (page 21)
Writes the serialized contents of the specified file wrapper to the pasteboard.

Reading Data

- [changeCount](#) (page 12)
Returns the receiver's change count.
- [dataForType:](#) (page 13)
Returns the data for the specified type.
- [propertyListForType:](#) (page 15)
Returns a property list object for the specified type.
- [readFileContentsType:toFile:](#) (page 16)
Reads data representing a file's contents from the receiver and writes it to the specified file.
- [readFileWrapper](#) (page 16)
Reads data representing a file's contents from the receiver and returns it as a file wrapper.
- [stringForType:](#) (page 19)
Returns the string for the specified type.
- [pasteboard:provideDataForType:](#) (page 22) *delegate method*
Implemented by an owner object (previously declared in a [declareTypes:owner:](#) (page 14) message) to provide promised data.

Determining Types

- [availableTypeFromArray:](#) (page 12)
Scans the specified types for a type that the receiver supports.
- [types](#) (page 20)
Returns an array of the receiver's supported data types.

Noticing Changed Ownership

- [pasteboardChangedOwner:](#) (page 22) *delegate method*
Notifies a prior owner of the specified pasteboard (and owners of representations on the pasteboard) that the pasteboard has changed owners.

Class Methods

generalPasteboard

Returns the general NSPasteboard object.

```
+ (NSPasteboard *)generalPasteboard
```

Return Value

The general pasteboard.

Discussion

Invokes [pasteboardWithName:](#) (page 9) to obtain the pasteboard.

Availability

Available in Mac OS X v10.0 and later.

Related Sample Code

Departments and Employees

iSpend

PDFKitLinker2

RGB Image

Sketch-112

Declared In

NSPasteboard.h

pasteboardByFilteringData:ofType:

Creates and returns a new pasteboard with a unique name that supplies the specified data in as many types as possible given the available filter services.

```
+ (NSPasteboard *)pasteboardByFilteringData:(NSData *)data ofType:(NSString *)type
```

Parameters

data

The data to be placed on the pasteboard.

type

The type of data in the *data* parameter.

Return Value

The new pasteboard object.

Discussion

The returned pasteboard also declares data of the supplied *type*.

No filter service is invoked until the data is actually requested, so invoking this method is reasonably inexpensive.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSPasteboard.h

pasteboardByFilteringFile:

Creates and returns a new pasteboard with a unique name that supplies the specified file data in as many types as possible given the available filter services.

```
+ (NSPasteboard *)pasteboardByFilteringFile:(NSString *)filename
```

Parameters

filename

The filename to put on the pasteboard.

Return Value

The new pasteboard object.

Discussion

No filter service is invoked until the data is actually requested, so invoking this method is reasonably inexpensive.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSPasteboard.h

pasteboardByFilteringTypesInPasteboard:

Creates and returns a new pasteboard with a unique name that supplies the specified pasteboard data in as many types as possible given the available filter services.

```
+ (NSPasteboard *)pasteboardByFilteringTypesInPasteboard:(NSPasteboard *)pasteboard
```

Parameters

pasteboard

The original pasteboard object.

Return Value

The new pasteboard object. This method returns the object in the *pasteboard* parameter if the pasteboard was returned by one of the `pasteboardByFiltering...` methods. This prevents a pasteboard from being expanded multiple times.

Discussion

This process can be thought of as expanding the pasteboard, because the new pasteboard generally contains more representations of the data than *pasteboard*.

This method only returns the original types and the types that can be created as a result of a single filter; the pasteboard does not have defined types that are the result of translation by multiple filters.

No filter service is invoked until the data is actually requested, so invoking this method is reasonably inexpensive.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSPasteboard.h

pasteboardWithName:

Returns the pasteboard with the specified name.

```
+ (NSPasteboard *)pasteboardWithName:(NSString *)name
```

Parameters*name*

The name of the pasteboard. The names of standard pasteboards are given in “[Pasteboard Names](#)” (page 23).

Return Value

The pasteboard associated with the given name, or a new `NSPasteboard` object if the application does not yet have a pasteboard object for the specified name.

Discussion

Other names can be assigned to create private pasteboards for other purposes.

Availability

Available in Mac OS X v10.0 and later.

Related Sample Code

CocoaDragAndDrop

PDFKitLinker2

Declared In

`NSPasteboard.h`

pasteboardWithUniqueName

Creates and returns a new pasteboard with a name that is guaranteed to be unique with respect to other pasteboards on the computer.

```
+ (NSPasteboard *)pasteboardWithUniqueName
```

Return Value

The new pasteboard object.

Discussion

This method is useful for applications that implement their own interprocess communication using pasteboards.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSPasteboard.h`

typesFilterableTo:

Returns the data types that can be converted to the specified type using the available filter services.

```
+ (NSArray *)typesFilterableTo:(NSString *)type
```

Parameters*type*

The target data type.

Return Value

An array of `NSString` objects containing the types that can be converted to the target data type.

Discussion

The array also contains the original type.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSPasteboard.h

Instance Methods

addTypes:owner:

Updates the list of declared data types in the receiver

```
- (NSInteger)addTypes:(NSArray *)newTypes owner:(id)newOwner
```

Parameters

newTypes

An array of `NSString` objects, each of which specifies a type of data that can be provided to the pasteboard.

newOwner

The object that provides the data for the specified types.

Return Value

The new change count, or 0 if there was an error adding the types.

Discussion

This method lets you declare additional types of data to the receiver. You can also use it to replace existing types added by a previous `declareTypes:owner:` (page 14) or `addTypes:owner: message`.

The *newTypes* parameter specifies the types of data you are promising to the pasteboard. The types should be ordered according to the preference of the source application, with the most preferred type coming first (typically, the richest representation). New types are added to the end of the list containing any existing types, if any.

If you specify a type that has already been declared, this method replaces the owner of that type with the value in *newOwner*. In addition, any data already written to the pasteboard for that type is removed.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [changeCount](#) (page 12)

Related Sample Code

CocoaDragAndDrop

RGB Image

RGB ValueTransformers

Declared In

NSPasteboard.h

availableTypeFromArray:

Scans the specified types for a type that the receiver supports.

```
- (NSString *)availableTypeFromArray:(NSArray *)types
```

Parameters

types

An array of `NSString` objects specifying the pasteboard types your application supports, in preferred order.

Return Value

The first pasteboard type in *types* that is available on the pasteboard, or `nil` if the receiver does not contain any of the types in *types*.

Discussion

You use this method to determine the best representation available on the pasteboard. For example, if your application supports RTFD, RTF, and string data, then you might invoke the method as follows:

```
NSArray *supportedTypes =
    [NSArray arrayWithObjects: NSRTFDpboardType, NSRTFPboardType,
    NSStringPboardType, nil];
NSString *bestType = [[NSPasteboard generalPasteboard]
    availableTypeFromArray:supportedTypes];
```

If the pasteboard contains RTF and string data, then `bestType` would contain `NSRTFPboardType`. If the pasteboard contains none of the types in `supportedTypes`, then `bestType` would be `nil`.

You must send a [types](#) (page 20) or `availableTypeFromArray:` message before reading any data from an `NSPasteboard` object. If you need to see if a type in the returned array matches a type string you have stored locally, use the `isEqualToString:` method to perform the comparison.

Availability

Available in Mac OS X v10.0 and later.

Related Sample Code

GLChildWindowDemo

MP3 Player

Sketch-112

Declared In

`NSPasteboard.h`

changeCount

Returns the receiver's change count.

```
- (NSInteger)changeCount
```

Return Value

The change count of the receiver.

Discussion

The change count starts at zero when a client creates the receiver and becomes the first owner. The change count increments for reasons other than ownership changes. Basically any modification of the pasteboard increments the change count. For example, when an owner converts a promise to actual data, the change count is incremented.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [declareTypes:owner:](#) (page 14)

Related Sample Code

Sketch-112

Declared In

NSPasteboard.h

dataForType:

Returns the data for the specified type.

```
- (NSData *)dataForType:(NSString *)dataType
```

Parameters

dataType

The type of data you want to read from the pasteboard. This value should be one of the types returned by [types](#) (page 20) or [availableTypeFromArray:](#) (page 12).

Return Value

A data object containing the data for the specified type, or `nil` if the contents of the pasteboard changed since they were last checked.

This method may also return `nil` if the pasteboard server cannot supply the data in time—for example, if the pasteboard's owner is slow in responding to a [pasteboard:provideDataForType:](#) (page 22) message and the interprocess communication times out.

Discussion

Errors other than a timeout raise a `NSPasteboardCommunicationException`.

If `nil` is returned, the application should put up a panel informing the user that it was unable to carry out the paste operation.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setData:forType:](#) (page 17)

Related Sample Code

Departments and Employees

iSpend

QTMetadataEditor

RGB Image

Sketch-112

Declared In

NSPasteboard.h

declareTypes:owner:

Prepares the receiver for a change in its contents by declaring the new types of data it will contain and a new owner.

```
- (NSInteger)declareTypes:(NSArray *)newTypes owner:(id)newOwner
```

Parameters*newTypes*

An array of `NSString` objects that specify the types of data that may be added to the new pasteboard. The types should be ordered according to the preference of the source application, with the most preferred type coming first (typically, the richest representation).

newOwner

The object responsible for writing data to the pasteboard, or `nil` if you provide data for all types immediately. If you specify a *newOwner* object, it must support all of the types declared in the *newTypes* parameter and must remain valid for as long as the data is *promised* on the pasteboard.

Return Value

The receiver's new change count.

Discussion

This is the first step in responding to a user's copy or cut command and must precede the messages that actually write the data. A `declareTypes:owner:` message essentially changes the contents of the receiver: It invalidates the current contents of the receiver and increments its change count.

You can write the data immediately after declaring the types or wait until it is required for a paste operation. If you wait, the object in the *newOwner* parameter receives a `pasteboard:provideDataForType:` (page 22) message requesting the data in a particular type when it is needed. You might choose to write data immediately for the most preferred type, but wait for the others to see whether they'll be requested.

A *newOwner* must remain valid for as long as the data is *promised* on the pasteboard. Once an object has provided all of the types it has declared it is no longer needed by the pasteboard—conceptually the “owner” owns the promises to provide data for declared types. See also `pasteboardChangedOwner:` (page 22).

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setString:forType:](#) (page 18)
- [addTypes:owner:](#) (page 11)
- [changeCount](#) (page 12)
- [pasteboardChangedOwner:](#) (page 22)

Related Sample Code

Departments and Employees

iSpend

PDFKitLinker2

Sketch-112
SpotlightFortunes

Declared In

NSPasteboard.h

name

Returns the receiver's name.

- (NSString *)name

Return Value

The name of the receiver.

Availability

Available in Mac OS X v10.0 and later.

See Also

+ [pasteboardWithName:](#) (page 9)

Declared In

NSPasteboard.h

propertyListForType:

Returns a property list object for the specified type.

- (id)propertyListForType:(NSString *)*dataType*

Parameters

dataType

The pasteboard data type containing the property-list data.

Return Value

The property-list object. This object consists of NSArray, NSData, NSDictionary, or NSString objects—or any combination thereof

Discussion

This method invokes the [dataForType:](#) (page 13) method.

Special Considerations

You must send [types](#) (page 20) or [availableTypeFromArray:](#) (page 12) before invoking `propertyListForType:`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setPropertyList:forType:](#) (page 18)

Related Sample Code

bMoviePalette

bMoviePaletteCocoa
 GLChildWindowDemo
 iSpend
 Sketch-112

Declared In

NSPasteboard.h

readFileContentsType:toFile:

Reads data representing a file's contents from the receiver and writes it to the specified file.

```
- (NSString *)readFileContentsType:(NSString *)type toFile:(NSString *)filename
```

Parameters

type

The pasteboard data type to read. You should generally specify a value for this parameter. If you specify `nil`, the filename extension (in combination with the `NSCreateFileContentsPboardType` function) is used to determine the type.

filename

The file to receive the pasteboard data.

Return Value

The name of the file into which the data was actually written.

Discussion

Data of any file contents type should only be read using this method. If data matching the specified type is not found on the pasteboard, data of type `NSFileContentsPboardType` is requested.

Special Considerations

You must send an [availableTypeFromArray:](#) (page 12) or [types](#) (page 20) message before invoking `readFileContentsType:toFile:.`

Availability

Available in Mac OS X v10.0 and later.

See Also

- [writeFileContents:](#) (page 20)

Declared In

NSPasteboard.h

readFileWrapper

Reads data representing a file's contents from the receiver and returns it as a file wrapper.

```
- (NSFileWrapper *)readFileWrapper
```

Return Value

A file wrapper containing the pasteboard data, or `nil` if the receiver contained no data of type `NSFileContentsPboardType`.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSPasteboard.h

releaseGlobally

Releases the receiver's resources in the pasteboard server.

```
- (void)releaseGlobally
```

Discussion

Because a pasteboard object is an autoreleased instance of `NSPasteboard`, it is not released by this method, and its retain count is not changed.

After this method is invoked, no other application can use the named pasteboard. A temporary, privately named pasteboard can be released this way when it is no longer needed, but a standard pasteboard should never be released globally.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSPasteboard.h

setData:forType:

Writes data of the specified type to the pasteboard server.

```
- (BOOL)setData:(NSData *)data forType:(NSString *)dataType
```

Parameters

data

The data to write to the pasteboard.

dataType

The type of data in the *data* parameter. The type must have been declared by a previous [declareTypes:owner:](#) (page 14) message.

Return Value

YES if the data was written successfully, otherwise NO if ownership of the pasteboard has changed. Any other error raises an `NSPasteboardCommunicationException`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [dataForType:](#) (page 13)
- [setPropertyList:forType:](#) (page 18)
- [setString:forType:](#) (page 18)

Related Sample Code

CocoaDragAndDrop

Departments and Employees
 PDFKitLinker2
 RGB Image
 Sketch-112

Declared In

NSPasteboard.h

setPropertyList:forType:

Writes the specified property list to the pasteboard server.

```
- (BOOL)setPropertyList:(id)propertyList forType:(NSString *)dataType
```

Parameters

propertyList

The property list data to write to the pasteboard.

dataType

The type of property-list data in the *propertyList* parameter. The type must have been declared by a previous `declareTypes:owner:` (page 14) message.

Return Value

YES if the data was written successfully, otherwise NO if ownership of the pasteboard has changed. Any other error raises an `NSPasteboardCommunicationException`.

Discussion

This method invokes `setData:forType:` (page 17) with a serialized property list parameter.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [propertyListForType:](#) (page 15)
- [setString:forType:](#) (page 18)

Related Sample Code

EnhancedAudioBurn
 iSpend
 OpenGLCompositorLab
 Quartz Composer WWDC 2005 TextEdit
 TextEditPlus

Declared In

NSPasteboard.h

setString:forType:

Writes the specified string to the pasteboard server.

```
- (BOOL)setString:(NSString *)string forType:(NSString *)dataType
```

Parameters*string*

The string to write to the pasteboard.

*dataType*The type of string data. The type must have been declared by a previous [declareTypes:owner:](#) (page 14) message.**Return Value**YES if the data was written successfully, otherwise NO if ownership of the pasteboard has changed. Any other error raises an `NSPasteboardCommunicationException`.**Discussion**This method invokes [setData:forType:](#) (page 17) to perform the write.**Availability**

Available in Mac OS X v10.0 and later.

See Also

- [stringForType:](#) (page 19)
- [setData:forType:](#) (page 17)
- [setPropertyList:forType:](#) (page 18)

Related Sample Code

Calculator

Departments and Employees

EnhancedDataBurn

QTMetadataEditor

SpotlightFortunes

Declared In

NSPasteboard.h

stringForType:

Returns the string for the specified type.

- (NSString *)stringForType:(NSString *)dataType

Parameters*dataType*

The pasteboard data type to read.

Return ValueA string containing the pasteboard data, or `nil` if the data could not be read.**Discussion**This method invokes `dataForType:` to obtain the string. If the string cannot be obtained, `stringForType:` returns `nil`. See [dataForType:](#) (page 13) for a description of what will cause `nil` to be returned.**Special Considerations**You must send [types](#) (page 20) or [availableTypeFromArray:](#) (page 12) before invoking `stringForType:`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setString:forType:](#) (page 18)

Related Sample Code

Calculator

iSpend

SpotlightFortunes

Declared In

NSPasteboard.h

types

Returns an array of the receiver's supported data types.

- (NSArray *)types

Return Value

An array of `NSString` objects containing the types of data declared for the current contents of the receiver. The returned types are listed in the order they were declared.

Discussion

You must send a `types` or [availableTypeFromArray:](#) (page 12) message before reading any data from an `NSPasteboard` object. If you need to see if a type in the returned array matches a type string you have stored locally, use the `isEqualToString:` method to perform the comparison.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [declareTypes:owner:](#) (page 14)

- [dataForType:](#) (page 13)

Related Sample Code

iSpend

Quartz Composer WWDC 2005 TextEdit

TextEditPlus

Declared In

NSPasteboard.h

writeFileContents:

Writes the contents of the specified file to the pasteboard.

- (BOOL)writeFileContents:(NSString *)filename

Parameters*filename*

The name of the file to write to the pasteboard.

Return Value

YES if the data was successfully written, otherwise NO.

Discussion

Writes the contents of the file *filename* to the receiver and declares the data to be of type `NSFileContentsPboardType` and also of a type appropriate for the file's extension (as returned by the `NSCreateFileContentsPboardType` function when passed the files extension), if it has one.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [readFileContentsType:toFile:](#) (page 16)

Declared In

`NSPasteboard.h`

writeFileWrapper:

Writes the serialized contents of the specified file wrapper to the pasteboard.

```
- (BOOL)writeFileWrapper:(NSFileWrapper *)wrapper
```

Parameters*wrapper*

The file wrapper to write to the pasteboard.

Return Value

YES if the data was successfully written, otherwise NO.

Discussion

Writes the serialized contents of the file wrapper *wrapper* to the receiver and declares the data to be of type `NSFileContentsPboardType` and also of a type appropriate for the file's extension (as returned by the `NSCreateFileContentsPboardType` function when passed the files extension), if it has one. If *wrapper* does not have a preferred filename, this method raises an exception.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSPasteboard.h`

Delegate Methods

pasteboard:provideDataForType:

Implemented by an owner object (previously declared in a [declareTypes:owner:](#) (page 14) message) to provide promised data.

```
- (void)pasteboard:(NSPasteboard *)sender provideDataForType:(NSString *)type
```

Parameters

sender

The pasteboard that requires the specified data for a paste operation.

type

The type of data the owner object must provide.

Discussion

The requested data should be written to *sender* using the [setData:forType:](#) (page 17), [setPropertyList:forType:](#) (page 18), or [setString:forType:](#) (page 18) method. The `pasteboard:provideDataForType:` messages may also be sent to the owner when the application is shut down through an application's `terminate:` method. This is the method that is invoked in response to a Quit command. Thus the user can copy something to the pasteboard, quit the application, and still paste the data that was copied. A `pasteboard:provideDataForType:` message is sent only if the specified type of data has not already been supplied to the pasteboard. Instead of writing all data types when the cut or copy operation is done, an application can choose to implement this method to provide the data for certain types only when they are requested.

If an application writes data to the pasteboard in the richest, and therefore most preferred, type at the time of a cut or copy operation, its `pasteboard:provideDataForType:` method can simply read that data from the pasteboard, convert it to the requested type, and write it back to the pasteboard as the new type.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSPasteboard.h`

pasteboardChangedOwner:

Notifies a prior owner of the specified pasteboard (and owners of representations on the pasteboard) that the pasteboard has changed owners.

```
- (void)pasteboardChangedOwner:(NSPasteboard *)sender
```

Parameters

sender

The pasteboard object whose owner changed.

Discussion

This method is optional and need only be implemented by pasteboard owners that need to know when they have lost ownership.

The owner is not able to read the contents of the pasteboard when responding to this method. The owner should be prepared to receive this method at any time, even from within the `declareTypes:owner:` (page 14) method used to declare ownership.

Once an owner has provided all of its data for declared types, it will not receive a `pasteboardChangedOwner:` message. If, therefore, you are maintaining an object just for the purpose of providing data lazily, rather than relying solely on receipt of a `pasteboardChangedOwner:` message you need to keep track of what types were promised and what types have been provided. When all the types have been provided, you may release the owner. See also `declareTypes:owner:` (page 14).

Availability

Available in Mac OS X v10.0 and later.

See Also

- [changeCount](#) (page 12)
- [declareTypes:owner:](#) (page 14)

Declared In

`NSPasteboard.h`

Constants

Pasteboard Names

The `NSPasteboard` class defines the following named pasteboards.

```
NSString *NSGeneralPboard;
NSString *NSFontPboard;
NSString *NSRulerPboard;
NSString *NSFindPboard;
NSString *NSDragPboard;
```

Constants

`NSGeneralPboard`

The pasteboard that's used for ordinary cut, copy, and paste operations.

This pasteboard holds the contents of the last selection that's been cut or copied.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

`NSFontPboard`

The pasteboard that holds font and character information and supports Copy Font and Paste Font commands that may be implemented in a text editor.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

`NSRulerPboard`

The pasteboard that holds information about paragraph formats in support of the Copy Ruler and Paste Ruler commands that may be implemented in a text editor.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSFindPboard

The pasteboard that holds information about the current state of the active application's find panel.

This information permits users to enter a search string into the find panel, then switch to another application to conduct another search.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSDragPboard

The pasteboard that stores data to be moved as the result of a drag operation.

For additional information on working with the drag pasteboard, see *Drag and Drop Programming Topics for Cocoa*.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

Declared In

`AppKit/NSPasteboard.h`

NSPasteboard Types for Standard Data

The `NSPasteboard` class uses the following common pasteboard data types.

```
NSString *NSStringPboardType;
NSString *NSFileNamesPboardType;
NSString *NSPostScriptPboardType;
NSString *NSTIFFPboardType;
NSString *NSRTFPboardType;
NSString *NSTabularTextPboardType;
NSString *NSFontPboardType;
NSString *NSRulerPboardType;
NSString *NSFileContentsPboardType;
NSString *NSColorPboardType;
NSString *NSRTFDPboardType;
NSString *NSHTMLPboardType;
NSString *NSPICTPboardType;
NSString *NSURLPboardType;
NSString *NSPDFPboardType;
NSString *NSVCardPboardType;
NSString *NSFilesPromisePboardType;
NSString *NSMultipleTextSelectionPboardType;
```

Constants

NSColorPboardType

NSColor **data**.

To write a color to a pasteboard you use `writeToPasteboard:(NSColor)`; to get a color from a pasteboard you use `colorFromPasteboard:(NSColor)`.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSFileContentsPboardType

A representation of a file's contents.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSFileNamesPboardType

An array of `NSString` objects designating one or more filenames.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSFontPboardType

Font and character information.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSHTMLPboardType

HTML (which an `NSTextView` object can read from, but not write to).

Available in Mac OS X v10.1 and later.

Declared in `NSPasteboard.h`.

NSPDFPboardType

PDF data.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSPICTPboardType

QuickDraw picture data.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSPostScriptPboardType

Encapsulated PostScript (EPS) code.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSRulerPboardType

Paragraph formatting information.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSRTFPboardType

Rich Text Format (RTF) data.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSRTFDPboardType

RTFD formatted file contents

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

NSStringPboardType

`NSString` data.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

`NSTabularTextPboardType`

An `NSString` object containing tab-separated fields of text.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

`NSTIFFPboardType`

Tag Image File Format (TIFF) data.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

`NSURLPboardType`

NSURL data for one file or resource.

To write an URL to a pasteboard you use `writeToPasteboard:(NSURL)`; to get an URL from a pasteboard you use `URLFromPasteboard:(NSURL)`.

Available in Mac OS X v10.0 and later.

Declared in `NSPasteboard.h`.

`NSVCardPboardType`

VCard data

Available in Mac OS X v10.2 and later.

Declared in `NSPasteboard.h`.

`NSFilesPromisePboardType`

Promised files.

For information on promised files, see *Dragging Files in Drag and Drop Programming Topics for Cocoa*.

Available in Mac OS X v10.2 and later.

Declared in `NSPasteboard.h`.

`NSInkTextPboardType`

Ink text data.

For information on ink text objects, see *Using Ink Services in Your Application*.

Available in Mac OS X v10.4 and later.

Declared in `NSPasteboard.h`.

`NSMultipleTextSelectionPboardType`

Multiple text selection.

Available in Mac OS X v10.5 and later.

Declared in `NSPasteboard.h`.

Discussion

See also `NSSoundPboardType` (`NSSound`).

Declared In

`AppKit/NSPasteboard.h`

Document Revision History

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

Date	Notes
2008-10-15	Clarified behavior of <code>pasteboardChangedOwner:</code> .
2008-02-08	Enhanced discussions of color and URL pasteboard types.
2007-01-04	Updated to include new API introduced in Mac OS X v10.5.
2006-05-23	Enhanced description of <code>availableTypeFromArray:</code> .
	First publication of this content as a separate document.

REVISION HISTORY

Document Revision History

Index

A

`addTypes:owner:` [instance method 11](#)
`availableTypeFromArray:` [instance method 12](#)

C

`changeCount` [instance method 12](#)

D

`dataForType:` [instance method 13](#)
`declareTypes:owner:` [instance method 14](#)

G

`generalPasteboard` [class method 7](#)

N

`name` [instance method 15](#)
`NSColorPboardType` [constant 24](#)
`NSDragPboard` [constant 24](#)
`NSFileContentsPboardType` [constant 24](#)
`NSFileNamesPboardType` [constant 25](#)
`NSFilesPromisePboardType` [constant 26](#)
`NSFindPboard` [constant 24](#)
`NSFontPboard` [constant 23](#)
`NSFontPboardType` [constant 25](#)
`NSGeneralPboard` [constant 23](#)
`NSHTMLPboardType` [constant 25](#)
`NSInkTextPboardType` [constant 26](#)
`NSMultipleTextSelectionPboardType` [constant 26](#)
`NSPasteboard Types for Standard Data` [24](#)
`NSPDFPboardType` [constant 25](#)

`NSPICTPboardType` [constant 25](#)
`NSPostScriptPboardType` [constant 25](#)
`NSRTFDPboardType` [constant 25](#)
`NSRTFPboardType` [constant 25](#)
`NSRulerPboard` [constant 23](#)
`NSRulerPboardType` [constant 25](#)
`NSStringPboardType` [constant 25](#)
`NSTabularTextPboardType` [constant 26](#)
`NSTIFFPboardType` [constant 26](#)
`NSURLPboardType` [constant 26](#)
`NSVCardPboardType` [constant 26](#)

P

`Pasteboard Names` [23](#)
`pasteboardByFilteringData:ofType:` [class method 8](#)
`pasteboardByFilteringFile:` [class method 8](#)
`pasteboardByFilteringTypesInPasteboard:` [class method 9](#)
`pasteboardChangedOwner: <NSObject> delegate` [method 22](#)
`pasteboard:provideDataForType: <NSObject> delegate` [method 22](#)
`pasteboardWithName:` [class method 9](#)
`pasteboardWithUniqueName` [class method 10](#)
`propertyListForType:` [instance method 15](#)

R

`readFileContentsType:toFile:` [instance method 16](#)
`readFileWrapper` [instance method 16](#)
`releaseGlobally` [instance method 17](#)

S

`setData:forType:` [instance method 17](#)
`setPropertyList:forType:` [instance method 18](#)

setString:forType: [instance method 18](#)
stringForType: [instance method 19](#)

T

types [instance method 20](#)
typesFilterableTo: [class method 10](#)

W

writeFileContents: [instance method 20](#)
writeFileWrapper: [instance method 21](#)