
NSDistributedNotificationCenter Class Reference

Interapplication Communication





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

NSDistributedNotificationCenter Class Reference 5

Class at a Glance	5
Overview	6
Tasks	6
Getting Distributed Notification Centers	6
Managing Observers	7
Posting Notifications	7
Suspending and Resuming Notification Delivery	7
Class Methods	7
defaultCenter	7
notificationCenterForType:	8
Instance Methods	8
addObserver:selector:name:object:	8
addObserver:selector:name:object:suspensionBehavior:	9
postNotificationName:object:	10
postNotificationName:object:userInfo:	10
postNotificationName:object:userInfo:deliverImmediately:	11
postNotificationName:object:userInfo:options:	12
removeObserver:name:object:	12
setSuspended:	13
suspended	13
Constants	14
Notification Center Type	14
Notification Posting Behavior	14
NSNotificationSuspensionBehavior	15

Document Revision History 17

Index 19

NSDistributedNotificationCenter Class Reference

Inherits from	NSNotificationCenter : NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/Foundation.framework
Availability	Available in Mac OS X v10.0 and later.
Companion guide	Notification Programming Topics for Cocoa
Declared in	NSDistributedNotificationCenter.h
Related sample code	From A View to A Movie

Class at a Glance

The `NSDistributedNotificationCenter` class provides a way to send notifications to objects in other tasks. It takes `NSNotification` objects and broadcasts them to any objects in other tasks that have registered for the notification with their task's default distributed notification center.

Principal Attributes

- **Notification dispatch table.** See “Class at a Glance” > “Principal Attributes” in *NSNotificationCenter Class Reference* for information about the dispatch table.

In addition to the notification name and sender, dispatch table entries for distributed notification centers specify when the notification center delivers notifications to its observers. See the [postNotificationName:object:userInfo:deliverImmediately:](#) (page 11) method, “[Suspending and Resuming Notification Delivery](#)” (page 7), and [NSNotificationSuspensionBehavior](#) (page 15) for details.

Commonly Used Methods

[defaultCenter](#) (page 7)

Accesses the default distributed notification center.

[addObserver:selector:name:object:suspensionBehavior:](#) (page 9)

Registers an object to receive a notification with a specified behavior when notification delivery is suspended.

[postNotificationName:object:userInfo:deliverImmediately:](#) (page 11)
Creates and posts a notification.

[removeObserver:name:object:](#) (page 12)
Specifies that an object no longer wants to receive certain notifications.

Overview

The `NSDistributedNotificationCenter` class implements a notification center that can distribute notifications asynchronously to tasks other than the one in which the notification was posted. An instance of this class are known as a **distributed notification center**.

Each task has a default distributed notification center that you access with the `defaultCenter` (page 7) class method. There may be different types of distributed notification centers. Currently there is a single type—`NSLocalNotificationCenterType`. This type of distributed notification center handles notifications that can be sent between tasks on a single computer. For communication between tasks on different computers, use *Distributed Objects Programming Topics*.

Posting a *distributed notification* is an expensive operation. The notification gets sent to a system-wide server that distributes it to all the tasks that have objects registered for distributed notifications. The latency between posting the notification and the notification's arrival in another task is unbounded. In fact, when too many notifications are posted and the server's queue fills up, notifications may be dropped.

Distributed notifications are delivered via a task's run loop. A task must be running a run loop in one of the "common" modes, such as `NSDefaultRunLoopMode`, to receive a distributed notification. For multithreaded applications running in Mac OS X v10.3 and later, distributed notifications are always delivered to the main thread. For multithreaded applications running in Mac OS X v10.2.8 and earlier, notifications are delivered to the thread that first used the distributed notifications API, which in most cases is the main thread.

Note: `NSDistributedNotificationCenter` objects should not be used to send notifications between threads within the same task. Use *Distributed Objects Programming Topics* or the `NSObject` method `performSelectorOnMainThread:withObject:waitUntilDone:`, instead. You can also setup an `NSPort` object to receive and distribute messages from other threads.

Tasks

Getting Distributed Notification Centers

- + [defaultCenter](#) (page 7)
Returns the default distributed notification center, representing the local notification center for the computer.
- + [notificationCenterForType:](#) (page 8)
Returns the distributed notification center for a particular notification center type.

Managing Observers

- [addObserver:selector:name:object:](#) (page 8)
Adds an entry to the receiver's dispatch table with an observer, a notification selector and optional criteria: notification name and sender.
- [addObserver:selector:name:object:suspensionBehavior:](#) (page 9)
Adds an entry to the receiver's dispatch table with a specific observer and suspended-notifications behavior, and optional notification name and sender.
- [removeObserver:name:object:](#) (page 12)
Removes matching entries from the receiver's dispatch table.

Posting Notifications

- [postNotificationName:object:](#) (page 10)
Creates a notification, and posts it to the receiver.
- [postNotificationName:object:userInfo:](#) (page 10)
Creates a notification with information, and posts it to the receiver.
- [postNotificationName:object:userInfo:deliverImmediately:](#) (page 11)
Creates a notification with information and an immediate-delivery specifier, and posts it to the receiver.
- [postNotificationName:object:userInfo:options:](#) (page 12)
Creates a notification with information, and posts it to the receiver.

Suspending and Resuming Notification Delivery

- [suspended](#) (page 13)
Returns a Boolean value that indicates whether notification delivery is suspended.
- [setSuspended:](#) (page 13)
Suspends or resumes notification delivery.

Class Methods

defaultCenter

Returns the default distributed notification center, representing the local notification center for the computer.

```
+ (id)defaultCenter
```

Return Value

Default distributed notification center for the computer.

Discussion

This method calls [notificationCenterForType:](#) (page 8) with an argument of `NSLocalNotificationCenterType`.

Availability

Available in Mac OS X v10.0 and later.

Related Sample Code

From A View to A Movie

Declared In

NSDistributedNotificationCenter.h

notificationCenterForType:

Returns the distributed notification center for a particular notification center type.

```
+ (NSDistributedNotificationCenter *)notificationCenterForType:(NSString *)notificationCenterType
```

Parameters

notificationCenterType

Notification center type being inquired about.

Return Value

Distributed notification center for *notificationCenterType*.

Discussion

Currently only one type, `NSLocalNotificationCenterType`, is supported.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSDistributedNotificationCenter.h

Instance Methods

addObserver:selector:name:object:

Adds an entry to the receiver's dispatch table with an observer, a notification selector and optional criteria: notification name and sender.

```
- (void)addObserver:(id)notificationObserver selector:(SEL)notificationSelector name:(NSString *)notificationName object:(NSString *)notificationSender
```

Parameters

notificationObserver

Object registering as an observer. Must not be `nil`.

notificationSelector

Selector that specifies the message the receiver sends *notificationObserver* to notify it of the notification posting. Must not be `0`.

notificationName

The name of the notification for which to register the observer; that is, only notifications with this name are delivered to the observer. When `nil`, the notification center doesn't use a notification's name to decide whether to deliver it to the observer.

notificationSender

The object whose notifications the observer wants to receive; that is, only notifications sent by this sender are delivered to the observer. When `nil`, the notification center doesn't use a notification's sender to decide whether to deliver it to the observer.

Discussion

This method calls `addObserver:selector:name:object:suspensionBehavior:` (page 9) with `suspensionBehavior:NSNotificationSuspensionBehaviorCoalesce` (described in “Constants” (page 14)).

Availability

Available in Mac OS X v10.0 and later.

Related Sample Code

From A View to A Movie

Declared In

NSDistributedNotificationCenter.h

addObserver:selector:name:object:suspensionBehavior:

Adds an entry to the receiver's dispatch table with a specific observer and suspended-notifications behavior, and optional notification name and sender.

```
- (void)addObserver:(id)notificationObserver selector:(SEL)notificationSelector
    name:(NSString *)notificationName object:(NSString *)notificationSender
    suspensionBehavior:(NSNotificationSuspensionBehavior)suspendedDeliveryBehavior
```

Parameters*notificationObserver*

Object registering as an observer. Must not be `nil`.

notificationSelector

Selector that specifies the message the receiver sends *notificationObserver* to notify it of the notification posting. Must not be `0`.

notificationName

The name of the notification for which to register the observer; that is, only notifications with this name are delivered to the observer. When `nil`, the notification center doesn't use a notification's name to decide whether to deliver it to the observer.

notificationSender

The object whose notifications the observer wants to receive; that is, only notifications sent by this sender are delivered to the observer. When `nil`, the notification center doesn't use a notification's sender to decide whether to deliver it to the observer.

suspendedDeliveryBehavior

Notification posting behavior when notification delivery is suspended.

Discussion

The receiver does not retain *notificationObserver*. Therefore, you should always send `removeObserver:` or `removeObserver:name:object:` (page 12) to the receiver before releasing *notificationObserver*.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [postNotificationName:object:userInfo:deliverImmediately:](#) (page 11)

Declared In

NSDistributedNotificationCenter.h

postNotificationName:object:

Creates a notification, and posts it to the receiver.

```
- (void)postNotificationName:(NSString *)notificationName object:(NSString *)notificationSender
```

Parameters

notificationName

Name of the notification to post. Must not be nil.

notificationSender

Sender of the notification. May be nil.

Discussion

This method invokes [postNotificationName:object:userInfo:deliverImmediately:](#) (page 11) with `userInfo:nil` `deliverImmediately:NO`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [postNotificationName:object:userInfo:](#) (page 10)
- [postNotificationName:object:userInfo:deliverImmediately:](#) (page 11)
- [postNotificationName:object:userInfo:options:](#) (page 12)

Declared In

NSDistributedNotificationCenter.h

postNotificationName:object:userInfo:

Creates a notification with information, and posts it to the receiver.

```
- (void)postNotificationName:(NSString *)notificationName object:(NSString *)notificationSender userInfo:(NSDictionary *)notificationInfo
```

Parameters

notificationName

Name of the notification to post. Must not be nil.

notificationSender

Sender of the notification. May be nil.

notificationInfo

Dictionary containing additional information. May be nil.

Discussion

This method invokes `postNotificationName:object:userInfo:deliverImmediately:` (page 11) with `deliverImmediately:NO`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- `postNotificationName:object:` (page 10)
- `postNotificationName:object:userInfo:deliverImmediately:` (page 11)
- `postNotificationName:object:userInfo:options:` (page 12)

Declared In

NSDistributedNotificationCenter.h

postNotificationName:object:userInfo:deliverImmediately:

Creates a notification with information and an immediate-delivery specifier, and posts it to the receiver.

```
- (void)postNotificationName:(NSString *)notificationName object:(NSString
    *)notificationSender userInfo:(NSDictionary *)userInfo
    deliverImmediately:(BOOL)deliverImmediately
```

Parameters

notificationName

Name of the notification to post. Must not be `nil`.

notificationSender

Sender of the notification. May be `nil`.

userInfo

Dictionary containing additional information. May be `nil`.

deliverImmediately

Specifies when to deliver the notification. When `NO`, the receiver delivers notifications to their observers according to the suspended-notification behavior specified in the corresponding dispatch table entry. When `YES`, the receiver delivers the notification immediately to its observers.

Discussion

This is the preferred method for posting notifications.

The *notificationInfo* dictionary is serialized as a property list, so it can be passed to another task. In the receiving task, it is deserialized back into a dictionary. This serialization imposes some restrictions on the objects that can be placed in the *notificationInfo* dictionary. See XML Property Lists for details.

Availability

Available in Mac OS X v10.0 and later.

See Also

- `postNotificationName:object:` (page 10)
- `postNotificationName:object:userInfo:` (page 10)
- `postNotificationName:object:userInfo:options:` (page 12)
- `encodeRootObject:` (NSArchiver)
- + `unarchiveObjectWithData:` (NSUnarchiver)

Declared In

NSDistributedNotificationCenter.h

postNotificationName:object:userInfo:options:

Creates a notification with information, and posts it to the receiver.

```
- (void)postNotificationName:(NSString *)notificationName object:(NSString *)notificationSender userInfo:(NSDictionary *)userInfo options:(NSUInteger)notificationOptions
```

Parameters*notificationName*Name of the notification to post. Must not be `nil`.*notificationSender*Sender of the notification. May be `nil`.*userInfo*Dictionary containing additional information. May be `nil`.*notificationOptions*Specifies how the notification is posted to the task and when to deliver it to its observers. See [“Notification Posting Behavior”](#) (page 14) for details.**Discussion**

The *userInfo* dictionary is serialized as a property list, so it can be passed to another task. In the receiving task, it is deserialized back into a dictionary. This serialization imposes some restrictions on the objects that can be placed in the *notificationInfo* dictionary. See XML Property Lists for details.

Availability

Available in Mac OS X v10.3 and later.

See Also

- [postNotificationName:object:](#) (page 10)
- [postNotificationName:object:userInfo:](#) (page 10)
- [postNotificationName:object:userInfo:deliverImmediately:](#) (page 11)

Declared In

NSDistributedNotificationCenter.h

removeObserver:name:object:

Removes matching entries from the receiver’s dispatch table.

```
- (void)removeObserver:(id)notificationObserver name:(NSString *)notificationName object:(NSString *)notificationSender
```

Parameters*notificationObserver*Observer to remove from the dispatch table. Specify an observer to remove only entries for this observer. When `nil`, the receiver does not use notification observers as criteria for removal.

notificationName

Name of the notification to remove from dispatch table. Specify a notification name to remove only entries that specify this notification name. When `nil`, the receiver does not use notification names as criteria for removal.

notificationSender

Sender to remove from the dispatch table. Specify a notification sender to remove only entries that specify this sender. When `nil`, the receiver does not use notification senders as criteria for removal.

Discussion

Be sure to invoke this method with `notificationName:nil notificationSender:nil` (or `removeObserver:`) before deallocating the observer object.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSDistributedNotificationCenter.h

setSuspended:

Suspends or resumes notification delivery.

- (void)setSuspended:(BOOL)suspended

Parameters

suspended

YES suspends notification delivery, NO resumes it.

Discussion

See [NSNotificationSuspensionBehavior](#) (page 15) for details on how the receiver delivers notifications to their observers when normal notification delivery is suspended.

The `NSApplication` class automatically suspends distributed notification delivery when the application is not active. Applications based on the Application Kit framework should let AppKit manage the suspension of notification delivery. Foundation-only programs may have occasional need to use this method.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [addObserver:selector:name:object:suspensionBehavior:](#) (page 9)
- [postNotificationName:object:userInfo:deliverImmediately:](#) (page 11)
- [suspended](#) (page 13)

Declared In

NSDistributedNotificationCenter.h

suspended

Returns a Boolean value that indicates whether notification delivery is suspended.

- (BOOL)suspended

Return Value

YES when notification delivery is suspended, otherwise NO.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

NSDistributedNotificationCenter.h

Constants

Notification Center Type

This constant specifies the notification center type.

```
FOUNDATION_EXPORT NSString * const NSLocalNotificationCenterType;
```

Constants

NSLocalNotificationCenterType

Distributes notifications to all tasks on the sender's computer.

Available in Mac OS X v10.0 and later.

Declared in NSDistributedNotificationCenter.h.

Declared In

NSDistributedNotificationCenter.h

Notification Posting Behavior

These constants specify the behavior of notifications posted using the [postNotificationName:object:userInfo:options:](#) (page 12) method.

```
enum {
    NSNotificationDeliverImmediately = (1 << 0),
    NSNotificationPostToAllSessions = (1 << 1)
};
```

Constants

NSNotificationDeliverImmediately

When set, the notification is delivered immediately to all observers, regardless of their suspension behavior or suspension state. When not set, allows the normal suspension behavior of notification observers to take place.

Available in Mac OS X v10.3 and later.

Declared in NSDistributedNotificationCenter.h.

`NSNotificationPostToAllSessions`

When set, the notification is posted to all sessions. When not set, the notification is sent only to applications within the same login session as the posting task.

Available in Mac OS X v10.3 and later.

Declared in `NSDistributedNotificationCenter.h`.

Declared In

`NSDistributedNotificationCenter.h`

NSNotificationSuspensionBehavior

These constants specify the types of notification delivery suspension behaviors.

```
typedef enum {
    NSNotificationSuspensionBehaviorDrop = 1,
    NSNotificationSuspensionBehaviorCoalesce = 2,
    NSNotificationSuspensionBehaviorHold = 3,
    NSNotificationSuspensionBehaviorDeliverImmediately = 4
} NSNotificationSuspensionBehavior;
```

Constants`NSNotificationSuspensionBehaviorDrop`

The server does not queue any notifications with this name and object until [setSuspended:](#) (page 13) with an argument of `NO` is called.

Available in Mac OS X v10.0 and later.

Declared in `NSDistributedNotificationCenter.h`.

`NSNotificationSuspensionBehaviorCoalesce`

The server only queues the last notification of the specified name and object; earlier notifications are dropped. In cover methods for which suspension behavior is not an explicit argument, `NSNotificationSuspensionBehaviorCoalesce` is the default.

Available in Mac OS X v10.0 and later.

Declared in `NSDistributedNotificationCenter.h`.

`NSNotificationSuspensionBehaviorHold`

The server holds all matching notifications until the queue has been filled (queue size determined by the server), at which point the server may flush queued notifications.

Available in Mac OS X v10.0 and later.

Declared in `NSDistributedNotificationCenter.h`.

`NSNotificationSuspensionBehaviorDeliverImmediately`

The server delivers notifications matching this registration irrespective of whether [setSuspended:](#) (page 13) with an argument of `YES` has been called. When a notification with this suspension behavior is matched, it has the effect of first flushing any queued notifications. The effect is as if [setSuspended:](#) (page 13) with an argument of `NO` were first called if the application is suspended, followed by the notification in question being delivered, followed by a transition back to the previous suspended or unsuspended state.

Available in Mac OS X v10.0 and later.

Declared in `NSDistributedNotificationCenter.h`.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSDistributedNotificationCenter.h

Document Revision History

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

Date	Notes
2007-04-01	Made minor corrections and editorial improvements.
	Moved main discussion about the notification dispatch table to <i>NSNotification Class Reference</i> .
2006-06-28	Updated the Constants section to conform to reference consistency guidelines.
2006-05-23	Made minor code layout and editorial changes.
	First publication of this content as a separate document.

REVISION HISTORY

Document Revision History

Index

A

addObserver:selector:name:object: **instance method 8**
addObserver:selector:name:object:
suspensionBehavior: **instance method 9**

D

defaultCenter **class method 7**

N

Notification Center Type **14**
Notification Posting Behavior **14**
notificationCenterForType: **class method 8**
NSLocalNotificationCenterType **constant 14**
NSNotificationDeliverImmediately **constant 14**
NSNotificationPostToAllSessions **constant 15**
NSNotificationSuspensionBehavior **data type 15**
NSNotificationSuspensionBehaviorCoalesce
constant 15
NSNotificationSuspensionBehaviorDeliverImmediately
constant 15
NSNotificationSuspensionBehaviorDrop **constant 15**
NSNotificationSuspensionBehaviorHold **constant 15**

P

postNotificationName:object: **instance method 10**
postNotificationName:object:userInfo: **instance method 10**
postNotificationName:object:userInfo:
deliverImmediately: **instance method 11**

postNotificationName:object:userInfo:options:
instance method 12

R

removeObserver:name:object: **instance method 12**

S

setSuspended: **instance method 13**
suspended **instance method 13**