NSHost Class Reference

Cocoa > Networking

2007-03-24

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

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

NSHost Class Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/Foundation.framework
Availability	Available in Mac OS X v10.0 and later.
Companion guide	Interacting with the Operating System
Declared in	NSHost.h
Related sample code	Core Data HTML Store NameAndAddress

Overview

The `NSHost` class provides methods to access the network name and address information for a host. Instances of the `NSHost` class represent individual **hosts** on a network. Use `NSHost` objects to get the current host's name and address and to look up other hosts by name or by address.

To create an `NSHost` object, use the `currentHost` (page 7), `hostWithAddress:` (page 7), or `hostWithName:` (page 8) class methods (don't use `alloc` and `init`). These methods use available network administration services (such as NetInfo or the Domain Name Service) to discover all names and addresses for the host requested. They don't attempt to contact the host itself, however. This approach avoids untimely delays due to a host being unavailable, but it may result in incomplete information about the host.

An `NSHost` object contains all of the network addresses and names discovered for a given host by the network administration services. Each `NSHost` object typically contains one unique address, but it may have more than one name. If an `NSHost` object has more than one name, the additional names are variations on the same name, typically the basic host name plus the fully qualified domain name. For example, with a host name "`sales`" in the domain "`anycorp.com`", an `NSHost` object can hold both the names "`sales`" and "`sales.anycorp.com`".

The `NSHost` class maintains a cache of the `NSHost` objects it creates so that requests for an existing `NSHost` object return that object instead of creating a new one. Use the `setHostCacheEnabled:` (page 9) method to turn the cache off, forcing lookup of hosts as they're requested. You can also use the `flushHostCache` (page 7) method to clear the cache of its entries so that subsequent requests look up the host information and create new instances.

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

Tasks

Creating Hosts

`+ currentHost` (page 7)
Returns an `NSHost` object representing the host the process is running on.

`+ hostWithAddress:` (page 7)
Returns the `NSHost` with the Internet address *address*.

`+ hostWithName:` (page 8)
Returns a host with a specific name.

Getting Host Information

`- address` (page 9)
Returns one of the network addresses of the receiver.

`- addresses` (page 10)
Returns all the network addresses of the receiver.

`- name` (page 11)
Returns one of the hostnames of the receiver.

`- names` (page 11)
Returns all the hostnames of the receiver.

Comparing Hosts

`- isEqualToHost:` (page 10)
Indicates whether the receiver represents the same host as another `NSHost` object.

Managing the Host Cache

`+ isHostCacheEnabled` (page 9)
Indicates whether caching is turned on or off.

`+ setHostCacheEnabled:` (page 9)
Specifies whether the receiver is to cache instances as it creates them to avoid creating duplicate instances.

`+ flushHostCache` (page 7)
Releases the cache of existing `NSHost` objects so subsequent requests for `NSHost` objects create new ones.

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

Class Methods

currentHost

Returns an `NSHost` object representing the host the process is running on.

```
+ (NSHost *)currentHost
```

Return Value
`NSHost` object for the process's host.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ hostWithAddress:` (page 7)
`+ hostWithName:` (page 8)

Related Sample Code
Core Data HTML Store
NameAndAddress

Declared In
`NSHost.h`

flushHostCache

Releases the cache of existing `NSHost` objects so subsequent requests for `NSHost` objects create new ones.

```
+ (void)flushHostCache
```

Discussion
`NSHost` objects that were retained before this method was invoked remain valid.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ isHostCacheEnabled` (page 9)
`+ setHostCacheEnabled:` (page 9)

Declared In
`NSHost.h`

hostWithAddress:

Returns the `NSHost` with the Internet address *address*.

```
+ (NSHost *)hostWithAddress:(NSString *)address
```


2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

Parameters

address

Network address to look up. For example, `@"127.0.0.1"` or `@"fe80::1"`.

Return Value

Host for *address*.

Discussion

If caching is turned on and the cache already contains an `NSHost` object with *address*, returns that object. Otherwise, this method creates an instance and returns it.

Availability

Available in Mac OS X v10.0 and later.

See Also

`+ hostWithName:` (page 8)

`+ setHostCacheEnabled:` (page 9)

Related Sample Code

NameAndAddress

Declared In

`NSHost.h`

hostWithName:

Returns a host with a specific name.

```
+ (NSHost *)hostWithName:(NSString *)hostname
```

Parameters

hostname

Name of the host to look up. Can be either a simple hostname, such as `@"sales"`, or a fully qualified domain name, such as `@"sales.anycorp.com"`.

Return Value

Host named *hostname*.

Discussion

If caching is turned on and the cache already contains an `NSHost` object with *name*, returns that object. Otherwise, this method creates a new instance and returns it.

Availability

Available in Mac OS X v10.0 and later.

See Also

`+ hostWithAddress:` (page 7)

`+ setHostCacheEnabled:` (page 9)

Related Sample Code

NameAndAddress

Declared In

`NSHost.h`

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

isHostCacheEnabled

Indicates whether caching is turned on or off.

```
+ (BOOL)isHostCacheEnabled
```

Return Value
`YES` when caching is turned on; `NO` otherwise.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ setHostCacheEnabled:` (page 9)
`+ flushHostCache` (page 7)

Declared In
`NSHost.h`

setHostCacheEnabled:

Specifies whether the receiver is to cache instances as it creates them to avoid creating duplicate instances.

```
+ (void)setHostCacheEnabled:(BOOL)cacheOn
```

Parameters
cacheOn
`YES` to turn on caching. `NO` to turn of caching.

Discussion
Caching is turned on by default.

This method doesn't flush the cache. If you turn caching off and then back on, new requests for hosts use what was in the cache at the time caching was turned off. However, `NSHost` objects created while caching is turned off aren't entered into the cache.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ isHostCacheEnabled` (page 9)
`+ flushHostCache` (page 7)

Declared In
`NSHost.h`

Instance Methods

address

Returns one of the network addresses of the receiver.

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

```
- (NSString *)address
```

Return Value
One of the network address for the receiver. For example, `@"192.42.172.1"` or `@"fe80::1"`.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `addresses` (page 10)
- `name` (page 11)

Related Sample Code
NameAndAddress

Declared In
`NSHost.h`

addresses

Returns all the network addresses of the receiver.

```
- (NSArray *)addresses
```

Return Value
All the network addresses of the receiver.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `address` (page 9)
- `names` (page 11)

Declared In
`NSHost.h`

isEqualToHost:

Indicates whether the receiver represents the same host as another `NSHost` object.

```
- (BOOL)isEqualToHost:(NSHost *)host
```

Parameters
host
Host to compare the receiver to.

Return Value
`YES` when the receiver and *host* share at least one network address; `NO` otherwise.

Availability
Available in Mac OS X v10.0 and later.

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

See Also
- `addresses` (page 10)

Declared In
`NSHost.h`

name

Returns one of the hostnames of the receiver.

```
- (NSString *)name
```

Return Value
One of the hostnames of the receiver. Can be either a simple hostname, such as `@"sales"`, or a fully qualified domain name, such as `@"sales.anycorp.com"`.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `address` (page 9)
- `names` (page 11)

Related Sample Code
Core Data HTML Store
NameAndAddress

Declared In
`NSHost.h`

names

Returns all the hostnames of the receiver.

```
- (NSArray *)names
```

Return Value
All the hostnames of the receiver.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `addresses` (page 10)
- `name` (page 11)

Declared In
`NSHost.h`

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

Document Revision History

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

Date	Notes
2007-03-24	Made editorial improvements.
2006-05-23	First publication of this content as a separate document.

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

Document Revision History

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.

Index

2007-03-24 | © 2007 Apple Inc. All Rights Reserved.