

---

# NSTimeZone Class Reference

[Cocoa > Data Management](#)



2008-02-08



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, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

iPhone and Numbers are trademarks of Apple Inc.

Times is a registered trademark of Heidelberger Druckmaschinen AG, available from Linotype Library GmbH.

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

## **NSTimeZone Class Reference 5**

---

Overview	5
Adopted Protocols	6
Tasks	6
Creating and Initializing Time Zone Objects	6
Working with System Time Zones	6
Getting Time Zone Information	7
Getting Information About a Specific Time Zone	7
Comparing Time Zones	7
Describing a Time Zone	7
Getting Information About Daylight Saving	7
Class Methods	8
abbreviationDictionary	8
defaultTimeZone	8
knownTimeZoneNames	9
localTimeZone	9
resetSystemTimeZone	10
setDefaultTimeZone:	10
systemTimeZone	10
timeZoneForSecondsFromGMT:	11
timeZoneWithAbbreviation:	11
timeZoneWithName:	12
timeZoneWithName:data:	12
Instance Methods	13
abbreviation	13
abbreviationForDate:	13
data	14
daylightSavingTimeOffset	14
daylightSavingTimeOffsetForDate:	15
description	15
initWithName:	15
initWithName:data:	16
isDaylightSavingTime	16
isDaylightSavingTimeForDate:	17
isEqualToTimeZone:	17
localizedName:locale:	17
name	18
nextDaylightSavingTimeTransition	18
nextDaylightSavingTimeTransitionAfterDate:	19
secondsFromGMT	19
secondsFromGMTForDate:	19

- Constants 20
  - NSTimeZoneNameStyle 20
  - Time Zone Name Styles 20
- Notifications 21
  - NSSystemTimeZoneDidChangeNotification 21

**Document Revision History 23**

---

**Index 25**

---

# NSTimeZone Class Reference

---

<b>Inherits from</b>	NSObject
<b>Conforms to</b>	NSCoding NSCopying NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/Foundation.framework
<b>Availability</b>	Available in Mac OS X v10.0 and later.
<b>Companion guide</b>	Date and Time Programming Guide for Cocoa
<b>Declared in</b>	NSTimeZone.h

## Overview

`NSTimeZone` is an abstract class that defines the behavior of time zone objects. Time zone objects represent geopolitical regions. Consequently, these objects have names for these regions. Time zone objects also represent a temporal offset, either plus or minus, from Greenwich Mean Time (GMT) and an abbreviation (such as PST for Pacific Standard Time).

`NSTimeZone` provides several class methods to get time zone objects: [`timeZoneWithName:`](#) (page 12), [`timeZoneWithName:data:`](#) (page 12), [`timeZoneWithAbbreviation:`](#) (page 11), and [`timeZoneForSecondsFromGMT:`](#) (page 11). The class also permits you to set the default time zone *within your application* ([`setDefaultTimeZone:`](#) (page 10)). You can access this default time zone at any time with the [`defaultTimeZone`](#) (page 8) class method, and with the [`localTimeZone`](#) (page 9) class method, you can get a relative time zone object that decodes itself to become the default time zone for any locale in which it finds itself.

Cocoa does not provide any API to change the time zone of the computer, or of other applications.

Some `NSDate` methods return date objects that are automatically bound to time zone objects. These date objects use the functionality of `NSTimeZone` to adjust dates for the proper locale. Unless you specify otherwise, objects returned from `NSDate` are bound to the default time zone for the current locale.

Note that, strictly, time zone database entries such as “America/Los\_Angeles” are IDs not names. An example of a time zone name is “Pacific Daylight Time”. Although many `NSTimeZone` method names include the word “name”, they refer to IDs.

`NSTimeZone` is “toll-free bridged” with its Core Foundation counterpart, [\*CFTimeZone Reference\*](#). This means that the Core Foundation type is interchangeable in function or method calls with the bridged Foundation object. Therefore, in a method where you see an `NSTimeZone *` parameter, you can pass a `CFTimeZoneRef`,

and in a function where you see a `CTimeZoneRef` parameter, you can pass an `NSTimeZone` instance (you cast one type to the other to suppress compiler warnings). See [Interchangeable Data Types](#) for more information on toll-free bridging.

## Adopted Protocols

### NSCoding

- `encodeWithCoder:`
- `initWithCoder:`

### NSCopying

- `copyWithZone:`

## Tasks

### Creating and Initializing Time Zone Objects

- + [timeZoneWithAbbreviation:](#) (page 11)  
Returns the time zone object identified by a given abbreviation.
- + [timeZoneWithName:](#) (page 12)  
Returns the time zone object identified by a given ID.
- + [timeZoneWithName:data:](#) (page 12)  
Returns the time zone with a given ID whose data has been initialized using given data,
- + [timeZoneForSecondsFromGMT:](#) (page 11)  
Returns a time zone object offset from Greenwich Mean Time by a given number of seconds.
- [initWithName:](#) (page 15)  
Returns a time zone initialized with a given ID.
- [initWithName:data:](#) (page 16)  
Initializes a time zone with a given ID and time zone data.

### Working with System Time Zones

- + [localTimeZone](#) (page 9)  
Returns an object that forwards all messages to the default time zone for the current application.
- + [defaultTimeZone](#) (page 8)  
Returns the default time zone for the current application.
- + [setDefaultTimeZone:](#) (page 10)  
Sets the default time zone for the current application to a given time zone.
- + [resetSystemTimeZone](#) (page 10)  
Resets the system time zone object cached by the application, if any.

- + [systemTimeZone](#) (page 10)  
Returns the time zone currently used by the system.

## Getting Time Zone Information

- + [abbreviationDictionary](#) (page 8)  
Returns a dictionary holding the mappings of time zone abbreviations to time zone names.
- + [knownTimeZoneNames](#) (page 9)  
Returns an array of strings listing the IDs of all the time zones known to the system.

## Getting Information About a Specific Time Zone

- [abbreviation](#) (page 13)  
Returns the abbreviation for the receiver.
- [abbreviationForDate:](#) (page 13)  
Returns the abbreviation for the receiver at a given date.
- [name](#) (page 18)  
Returns the geopolitical region ID that identifies the receiver.
- [secondsFromGMT](#) (page 19)  
Returns the current difference in seconds between the receiver and Greenwich Mean Time.
- [secondsFromGMTForDate:](#) (page 19)  
Returns the difference in seconds between the receiver and Greenwich Mean Time at a given date.
- [data](#) (page 14)  
Returns the data that stores the information used by the receiver.

## Comparing Time Zones

- [isEqualToTimeZone:](#) (page 17)  
Returns a Boolean value that indicates whether the receiver has the same name and data as another given time zone.

## Describing a Time Zone

- [description](#) (page 15)  
Returns the description of the receiver.
- [localizedName:locale:](#) (page 17)  
Returns the name of the receiver localized for a given locale.

## Getting Information About Daylight Saving

- [isDaylightSavingTime](#) (page 16)  
Returns a Boolean value that indicates whether the receiver is currently using daylight saving time.

- [daylightSavingTimeOffset](#) (page 14)  
Returns the current daylight saving time offset of the receiver.
- [isDaylightSavingTimeForDate:](#) (page 17)  
Returns a Boolean value that indicates whether the receiver uses daylight savings time at a given date.
- [daylightSavingTimeOffsetForDate:](#) (page 15)  
Returns the daylight saving time offset for a given date.
- [nextDaylightSavingTimeTransition](#) (page 18)  
Returns the date of the next daylight saving time transition for the receiver.
- [nextDaylightSavingTimeTransitionAfterDate:](#) (page 19)  
Returns the next daylight saving time transition after a given date.

## Class Methods

### abbreviationDictionary

Returns a dictionary holding the mappings of time zone abbreviations to time zone names.

```
+ (NSDictionary *)abbreviationDictionary
```

#### Return Value

A dictionary holding the mappings of time zone abbreviations to time zone names.

#### Discussion

Note that more than one time zone may have the same abbreviation—for example, US/Pacific and Canada/Pacific both use the abbreviation “PST.” In these cases, `abbreviationDictionary` chooses a single name to map the abbreviation to.

#### Availability

Available in Mac OS X v10.0 and later.

#### Declared In

`NSTimeZone.h`

### defaultTimeZone

Returns the default time zone for the current application.

```
+ (NSTimeZone *)defaultTimeZone
```

#### Return Value

The default time zone for the current application. If no default time zone has been set, this method invokes [systemTimeZone](#) (page 10) and returns the system time zone.

#### Discussion

The default time zone is the one that the application is running with, which you can change (so you can make the application run as if it were in a different time zone).



If you get the default time zone and hold onto the returned object, it does not change if a subsequent invocation of `setDefaultTimeZone:` (page 10) changes the default time zone—you still have the specific time zone you originally got. Contrast this behavior with the object returned by `localTimeZone` (page 9).

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

- + [localTimeZone](#) (page 9)
- + [setDefaultTimeZone:](#) (page 10)
- + [systemTimeZone](#) (page 10)

#### Declared In

NSTimeZone.h

## knownTimeZoneNames

Returns an array of strings listing the IDs of all the time zones known to the system.

```
+ (NSArray *)knownTimeZoneNames
```

#### Return Value

An array of strings listing the IDs of all the time zones known to the system.

#### Availability

Available in Mac OS X v10.0 and later.

#### Declared In

NSTimeZone.h

## localTimeZone

Returns an object that forwards all messages to the default time zone for the current application.

```
+ (NSTimeZone *)localTimeZone
```

#### Return Value

An object that forwards all messages to the default time zone for the current application.

#### Discussion

The local time zone represents the current state of the default time zone at all times. If you get the *default* time zone (using `defaultTimeZone` (page 8)) and hold onto the returned object, it does not change if a subsequent invocation of `setDefaultTimeZone:` (page 10) changes the default time zone—you still have the specific time zone you originally got. The *local* time zone adds a level of indirection, it acts as if it were the current default time zone whenever you invoke a method on it.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

- + [defaultTimeZone](#) (page 8)
- + [setDefaultTimeZone:](#) (page 10)

**Declared In**

NSTimeZone.h

**resetSystemTimeZone**

Resets the system time zone object cached by the application, if any.

```
+ (void)resetSystemTimeZone
```

**Discussion**

If the application has cached the system time zone, this method clears that cached object. If you subsequently invoke [systemTimeZone](#) (page 10), NSTimeZone will attempt to redetermine the system time zone and a new object will be created and cached (see [systemTimeZone](#) (page 10)).

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [systemTimeZone](#) (page 10)

**Declared In**

NSTimeZone.h

**setDefaultTimeZone:**

Sets the default time zone for the current application to a given time zone.

```
+ (void)setDefaultTimeZone:(NSTimeZone *)aTimeZone
```

**Parameters**

*aTimeZone*

The new default time zone for the current application.

**Discussion**

There can be only one default time zone, so by setting a new default time zone, you lose the previous one.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [defaultTimeZone](#) (page 8)

+ [localTimeZone](#) (page 9)

**Declared In**

NSTimeZone.h

**systemTimeZone**

Returns the time zone currently used by the system.

```
+ (NSTimeZone *)systemTimeZone
```

**Return Value**

The time zone currently used by the system. If the current time zone cannot be determined, returns the GMT time zone.

**Special Considerations**

If you get the system time zone, it is cached by the application and does not change if the user subsequently changes the system time zone. The next time you invoke `systemTimeZone`, you get back the same time zone you originally got. You have to invoke [resetSystemTimeZone](#) (page 10) to clear the cached object.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [resetSystemTimeZone](#) (page 10)

**Declared In**

NSTimeZone.h

**timeZoneForSecondsFromGMT:**

Returns a time zone object offset from Greenwich Mean Time by a given number of seconds.

```
+ (id)timeZoneForSecondsFromGMT:(NSInteger)seconds
```

**Parameters**

*seconds*

The number of seconds by which the new time zone is offset from GMT.

**Return Value**

A time zone object offset from Greenwich Mean Time by *seconds*.

**Discussion**

The name of the new time zone is GMT +/- the offset, in hours and minutes. Time zones created with this method never have daylight savings, and the offset is constant no matter the date.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ [timeZoneWithAbbreviation:](#) (page 11)

+ [timeZoneWithName:](#) (page 12)

**Declared In**

NSTimeZone.h

**timeZoneWithAbbreviation:**

Returns the time zone object identified by a given abbreviation.

```
+ (id)timeZoneWithAbbreviation:(NSString *)abbreviation
```

**Parameters***abbreviation*

An abbreviation for a time zone.

**Return Value**

The time zone object identified by *abbreviation* determined by resolving the abbreviation to a name using the abbreviation dictionary and then returning the time zone for that name. Returns `nil` if there is no match for *abbreviation*.

**Discussion**

In general, you are discouraged from using abbreviations except for unique instances such as “UTC” or “GMT”. Time Zone abbreviations are not standardized and so a given abbreviation may have multiple meanings—for example, “EST” refers to Eastern Time in both the United States and Australia

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- + [abbreviationDictionary](#) (page 8)
- + [timeZoneForSecondsFromGMT:](#) (page 11)
- + [timeZoneWithName:](#) (page 12)

**Declared In**

NSTimeZone.h

**timeZoneWithName:**

Returns the time zone object identified by a given ID.

```
+ (id)timeZoneWithName:(NSString *)aTimeZoneName
```

**Parameters***aName*

The ID for the time zone.

**Return Value**

The time zone in the information directory with a name matching *aName*. Returns `nil` if there is no match for the name.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- + [timeZoneForSecondsFromGMT:](#) (page 11)
- + [timeZoneWithAbbreviation:](#) (page 11)
- + [knownTimeZoneNames](#) (page 9)

**Declared In**

NSTimeZone.h

**timeZoneWithName:data:**

Returns the time zone with a given ID whose data has been initialized using given data,

```
+ (id)timeZoneWithName:(NSString *)aTimeZoneName data:(NSData *)data
```

**Parameters**

*aTimeZoneName*

The ID for the time zone.

*data*

The data from the time-zone files located at `/usr/share/zoneinfo`.

**Return Value**

The time zone with the ID *aTimeZoneName* whose data has been initialized using the contents of *data*.

**Discussion**

You should not call this method directly—use `timeZoneWithName:` (page 12) to get the time zone object for a given name.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

+ `timeZoneWithName:` (page 12)

**Declared In**

NSTimeZone.h

## Instance Methods

### abbreviation

Returns the abbreviation for the receiver.

```
- (NSString *)abbreviation
```

**Return Value**

The abbreviation for the receiver, such as “EDT” (Eastern Daylight Time).

**Discussion**

Invokes `abbreviationForDate:` (page 13) with the current date as the argument.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

### abbreviationForDate:

Returns the abbreviation for the receiver at a given date.

```
- (NSString *)abbreviationForDate:(NSDate *)aDate
```

**Parameters***aDate*

The date for which to get the abbreviation for the receiver.

**Return Value**

The abbreviation for the receiver at *aDate*.

**Discussion**

Note that the abbreviation may be different at different dates. For example, during daylight savings time the US/Eastern time zone has an abbreviation of “EDT.” At other times, its abbreviation is “EST.”

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

**data**

Returns the data that stores the information used by the receiver.

- (NSData \*)data

**Return Value**

The data that stores the information used by the receiver.

**Discussion**

This data should be treated as an opaque object.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

**daylightSavingTimeOffset**

Returns the current daylight saving time offset of the receiver.

- (NSTimeInterval)daylightSavingTimeOffset

**Return Value**

The daylight current saving time offset of the receiver.

**Availability**

Available in Mac OS X v10.5 and later.

**See Also**

- [isDaylightSavingTime](#) (page 16)
- [isDaylightSavingTimeForDate:](#) (page 17)
- [daylightSavingTimeOffsetForDate:](#) (page 15)

**Declared In**

NSTimeZone.h

## daylightSavingTimeOffsetForDate:

Returns the daylight saving time offset for a given date.

```
- (NSTimeInterval)daylightSavingTimeOffsetForDate:(NSDate *)aDate
```

### Parameters

*aDate*

A date.

### Return Value

The daylight saving time offset for *aDate*.

### Availability

Available in Mac OS X v10.5 and later.

### See Also

- [isDaylightSavingTime](#) (page 16)
- [daylightSavingTimeOffset](#) (page 14)
- [isDaylightSavingTimeForDate:](#) (page 17)
- [nextDaylightSavingTimeTransitionAfterDate:](#) (page 19)

### Declared In

NSTimeZone.h

## description

Returns the description of the receiver.

```
- (NSString *)description
```

### Return Value

The description of the receiver, including the name, abbreviation, offset from GMT, and whether or not daylight savings time is currently in effect.

### Availability

Available in Mac OS X v10.0 and later.

### Declared In

NSTimeZone.h

## initWithName:

Returns a time zone initialized with a given ID.

```
- (id)initWithName:(NSString *)aName
```

### Parameters

*aName*

The ID for the time zone.

### Return Value

A time zone object initialized with the ID *aName*.

**Discussion**

If *aName* is a known ID, this method calls [initWithName:data:](#) (page 16) with the appropriate data object.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

**initWithName:data:**

Initializes a time zone with a given ID and time zone data.

```
- (id)initWithName:(NSString *)aName data:(NSData *)data
```

**Parameters**

*aName*

The ID for the time zone.

*data*

The data from the time-zone files located at `/usr/share/zoneinfo`.

**Discussion**

You should not call this method directly—use [initWithName:](#) (page 15) to get a time zone object.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

**isDaylightSavingTime**

Returns a Boolean value that indicates whether the receiver is currently using daylight saving time.

```
- (BOOL)isDaylightSavingTime
```

**Return Value**

YES if the receiver is currently using daylight savings time, otherwise NO.

**Discussion**

This method invokes [isDaylightSavingTimeForDate:](#) (page 17) with the current date as the argument.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [isDaylightSavingTimeForDate:](#) (page 17)
- [daylightSavingTimeOffset](#) (page 14)
- [daylightSavingTimeOffsetForDate:](#) (page 15)
- [nextDaylightSavingTimeTransition](#) (page 18)
- [nextDaylightSavingTimeTransitionAfterDate:](#) (page 19)



**Declared In**

NSTimeZone.h

**isDaylightSavingTimeForDate:**

Returns a Boolean value that indicates whether the receiver uses daylight savings time at a given date.

- (BOOL)isDaylightSavingTimeForDate:(NSDate \*)*aDate***Parameters***aDate*

The date against which to test the receiver.

**Return Value**YES if the receiver uses daylight savings time at *aDate*, otherwise NO.**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [isDaylightSavingTime](#) (page 16)
- [daylightSavingTimeOffset](#) (page 14)
- [daylightSavingTimeOffsetForDate:](#) (page 15)
- [nextDaylightSavingTimeTransitionAfterDate:](#) (page 19)

**Declared In**

NSTimeZone.h

**isEqualTimeZone:**

Returns a Boolean value that indicates whether the receiver has the same name and data as another given time zone.

- (BOOL)isEqualTimeZone:(NSTimeZone \*)*aTimeZone***Parameters***aTimeZone*

The time zone to compare with the receiver.

**Return Value**YES if *aTimeZone* and the receiver have the same name and data, otherwise NO.**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

**localizedName:locale:**

Returns the name of the receiver localized for a given locale.

- (NSString \*)localizedName:(NSTimeZoneNameStyle)style locale:(NSLocale \*)locale

#### Parameters

*style*

The format style for the returned string.

*locale*

The locale for which to format the name.

#### Return Value

The name of the receiver localized for *locale* using *style*.

#### Availability

Available in Mac OS X v10.5 and later.

#### Declared In

NSTimeZone.h

## name

Returns the geopolitical region ID that identifies the receiver.

- (NSString \*)name

#### Return Value

The geopolitical region ID that identifies the receiver.

#### Availability

Available in Mac OS X v10.0 and later.

#### Declared In

NSTimeZone.h

## nextDaylightSavingTimeTransition

Returns the date of the next daylight saving time transition for the receiver.

- (NSDate \*)nextDaylightSavingTimeTransition

#### Return Value

The date of the next (after the current instant) daylight saving time transition for the receiver.

#### Availability

Available in Mac OS X v10.5 and later.

#### See Also

- [isDaylightSavingTime](#) (page 16)
- [isDaylightSavingTimeForDate:](#) (page 17)
- [nextDaylightSavingTimeTransitionAfterDate:](#) (page 19)

#### Declared In

NSTimeZone.h

**nextDaylightSavingTimeTransitionAfterDate:**

Returns the next daylight saving time transition after a given date.

```
- (NSDate *)nextDaylightSavingTimeTransitionAfterDate:(NSDate *)aDate
```

**Parameters**

*aDate*

A date.

**Return Value**

The next daylight saving time transition after *aDate*.

**Availability**

Available in Mac OS X v10.5 and later.

**See Also**

- [isDaylightSavingTime](#) (page 16)
- [isDaylightSavingTimeForDate:](#) (page 17)
- [nextDaylightSavingTimeTransition](#) (page 18)

**Declared In**

NSTimeZone.h

**secondsFromGMT**

Returns the current difference in seconds between the receiver and Greenwich Mean Time.

```
- (NSInteger)secondsFromGMT
```

**Return Value**

The current difference in seconds between the receiver and Greenwich Mean Time.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

**secondsFromGMTForDate:**

Returns the difference in seconds between the receiver and Greenwich Mean Time at a given date.

```
- (NSInteger)secondsFromGMTForDate:(NSDate *)aDate
```

**Parameters**

*aDate*

The date against which to test the receiver.

**Return Value**

The difference in seconds between the receiver and Greenwich Mean Time at *aDate*.

**Discussion**

The difference may be different from the current difference if the time zone changes its offset from GMT at different points in the year—for example, the U.S. time zones change with daylight savings time.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSTimeZone.h

## Constants

**NSTimeZoneNameStyle**

Defines a type for time zone name styles.

```
typedef NSInteger NSTimeZoneNameStyle;
```

**Discussion**

See [“Time Zone Name Styles”](#) (page 20) for possible values.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

NSTimeZone.h

**Time Zone Name Styles**

Specify styles for presenting time zone names.

```
enum {
    NSTimeZoneNameStyleStandard,
    NSTimeZoneNameStyleShortStandard,
    NSTimeZoneNameStyleDaylightSaving,
    NSTimeZoneNameStyleShortDaylightSaving
};
```

**Constants**

NSTimeZoneNameStyleStandard

Specifies a standard name style.

Available in Mac OS X v10.5 and later.

Declared in NSTimeZone.h.

NSTimeZoneNameStyleShortStandard

Specifies a short name style.

Available in Mac OS X v10.5 and later.

Declared in NSTimeZone.h.

`NSTimeZoneNameStyleDaylightSaving`  
Specifies a daylight saving name style.  
Available in Mac OS X v10.5 and later.  
Declared in `NSTimeZone.h`.

`NSTimeZoneNameStyleShortDaylightSaving`  
Specifies a short daylight saving name style.  
Available in Mac OS X v10.5 and later.  
Declared in `NSTimeZone.h`.

**Declared In**  
`NSTimeZone.h`

## Notifications

### **NSSystemTimeZoneDidChangeNotification**

Sent when the time zone changed.

**Availability**  
Available in Mac OS X v10.5 and later.

**Declared In**  
`NSTimeZone.h`



# Document Revision History

---

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

Date	Notes
2008-02-08	Enhanced the discussions of the methods <code>systemTimeZone</code> and <code>resetSystemTimeZone</code> .
2007-05-21	Updated to include API introduced in Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History



# Index

---

## A

---

abbreviation **instance method** [13](#)  
abbreviationDictionary **class method** [8](#)  
abbreviationForDate: **instance method** [13](#)

## D

---

data **instance method** [14](#)  
daylightSavingTimeOffset **instance method** [14](#)  
daylightSavingTimeOffsetForDate: **instance method** [15](#)  
defaultTimeZone **class method** [8](#)  
description **instance method** [15](#)

## I

---

initWithName: **instance method** [15](#)  
initWithName:data: **instance method** [16](#)  
isDaylightSavingTime **instance method** [16](#)  
isDaylightSavingTimeForDate: **instance method** [17](#)  
isEqualToTimeZone: **instance method** [17](#)

## K

---

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

## L

---

localizedName:locale: **instance method** [17](#)  
localTimeZone **class method** [9](#)

## N

---

name **instance method** [18](#)  
nextDaylightSavingTimeTransition **instance method** [18](#)  
nextDaylightSavingTimeTransitionAfterDate: **instance method** [19](#)  
NSSystemTimeZoneDidChangeNotification **notification** [21](#)  
NSTimeZoneNameStyle **data type** [20](#)  
NSTimeZoneNameStyleDaylightSaving **constant** [21](#)  
NSTimeZoneNameStyleShortDaylightSaving **constant** [21](#)  
NSTimeZoneNameStyleShortStandard **constant** [20](#)  
NSTimeZoneNameStyleStandard **constant** [20](#)

## R

---

resetSystemTimeZone **class method** [10](#)

## S

---

secondsFromGMT **instance method** [19](#)  
secondsFromGMTForDate: **instance method** [19](#)  
setDefaultTimeZone: **class method** [10](#)  
systemTimeZone **class method** [10](#)

## T

---

Time Zone Name Styles [20](#)  
timeZoneForSecondsFromGMT: **class method** [11](#)  
timeZoneWithAbbreviation: **class method** [11](#)  
timeZoneWithName: **class method** [12](#)  
timeZoneWithName:data: **class method** [12](#)