

---

# IMService Class Reference

[Audio & Video](#)



2009-07-17



Apple Inc.  
© 2009 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, Bonjour, Cocoa, iChat, 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

## **IMService Class Reference 5**

---

Overview	5
Tasks	5
Accessing Instant Messaging Services	5
Accessing Service Attributes	5
Accessing Buddies	6
Class Methods	6
allServices	6
imageNameForStatus:	7
myIdleTime	7
myStatus	7
notificationCenter	8
serviceWithName:	8
Instance Methods	9
infoForAllScreenNames	9
infoForPreferredScreenNames	9
infoForScreenName:	10
localizedName	10
localizedShortName	10
name	11
peopleWithScreenName:	11
screenNamesForPerson:	12
status	12
Constants	12
Screen Name Properties	12
Person Capability Values	14
IMServiceStatus	15
IMPersonStatus	16
Notifications	17
IMMyStatusChangedNotification	17
IMPersonInfoChangedNotification	17
IMPersonStatusChangedNotification	17
IMServiceStatusChangedNotification	18
IMStatusImagesChangedAppearanceNotification	18

## **Appendix A      Deprecated IMService Methods 19**

---

Deprecated in Mac OS X v10.5	19
imageURLForStatus:	19

**Document Revision History 21**

---

**Index 23**

---

# IMService Class Reference

---

<b>Inherits from</b>	NSObject
<b>Conforms to</b>	NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/InstantMessage.framework
<b>Availability</b>	Available in Mac OS X v10.4 and later.
<b>Companion guide</b>	Instant Message Programming Guide
<b>Declared in</b>	IMService.h
<b>Related sample code</b>	ABPresence iChatTheater

## Overview

The `IMService` class provides methods for getting information about an instant message service. Each `IMService` object represents one service available through iChat. Class methods such as `allServices` and `serviceName:` return these objects. Each object represents a single instant messaging service, allowing you to access the iChat status of the user, the user's list of buddies, and other information that can be integrated into your application. A variety of status notifications related to the user's status and preferences are posted by the `IMService` custom notification center.

## Tasks

### Accessing Instant Messaging Services

- + `allServices` (page 6)  
Returns an array of the currently available services.
- + `serviceName:` (page 8)  
Returns the specified service.

### Accessing Service Attributes

- + `imageNameForStatus:` (page 7)  
Returns the name of the image for the specified status of a person.

- + [myIdleTime](#) (page 7)  
Returns the number of seconds that the current user is idle.
- + [myStatus](#) (page 7)  
Returns the status of the current user.
- + [notificationCenter](#) (page 8)  
Returns the custom notification center for the service.
- [localizedName](#) (page 10)  
Returns the user-visible localized name of the service.
- [localizedShortName](#) (page 10)  
Returns a short version, if available, of the user-visible localized name of the service.
- [name](#) (page 11)  
Returns the fixed canonical name of the service.
- [status](#) (page 12)  
Returns the login status of the service.
- + [imageUrlForStatus:](#) (page 19) **Deprecated in Mac OS X v10.5**  
Returns the URL of the image for the specified status of a person.

## Accessing Buddies

- [peopleWithScreenName:](#) (page 11)  
Returns Address Book entries that match the specified screen name of a buddy.
- [screenNamesForPerson:](#) (page 12)  
Returns an array of strings that are valid screen names for the specified person.
- [infoForAllScreenNames](#) (page 9)  
Returns information about all buddies for the service.
- [infoForPreferredScreenNames](#) (page 9)  
Returns information about just the preferred accounts for all buddies.
- [infoForScreenName:](#) (page 10)  
Returns information about a buddy with the specified screen name.

## Class Methods

### allServices

Returns an array of the currently available services.

```
+ (NSArray *)allServices
```

#### Return Value

Returns an NSArray of IMService objects corresponding to the current available services (AIM, Bonjour, and so on.).

#### Availability

Available in Mac OS X v10.4 and later.

**See Also**

+ [serviceName:](#) (page 8)

**Declared In**

IMService.h

**imageNameForStatus:**

Returns the name of the image for the specified status of a person.

```
+ (NSString *)imageNameForStatus:(IMPersonStatus)status
```

**Parameters**

*status*

The status of a person. See “[IMPersonStatus](#)” (page 16) for possible values.

**Return Value**

The name of an image that reflects the current online status of a person; it is usually a colored bubble or triangle.

**Availability**

Available in Mac OS X v10.5 and later.

**See Also**

+ [myStatus](#) (page 7)

**Related Sample Code**

ABPresence

**Declared In**

IMService.h

**myIdleTime**

Returns the number of seconds that the current user is idle.

```
+ (NSDate *)myIdleTime
```

**Return Value**

The number of seconds that the current user is idle.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

IMService.h

**myStatus**

Returns the status of the current user.

```
+ (IMPersonStatus)myStatus
```

**Return Value**

A code representing the status of the current user. See “[IMPersonStatus](#)” (page 16) for possible values.

**Discussion**

This status is global across all services.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

+ [imageNameForStatus:](#) (page 7)

**Declared In**

IMService.h

**notificationCenter**

Returns the custom notification center for the service.

```
+ (NSNotificationCenter *)notificationCenter
```

**Return Value**

A custom notification center that manages IMService notifications.

**Availability**

Available in Mac OS X v10.4 and later.

**Related Sample Code**

ABPresence

iChatTheater

**Declared In**

IMService.h

**serviceName:**

Returns the specified service.

```
+ (IMService *)serviceName:(NSString *)name
```

**Parameters**

*name*

A service name as returned by a previous call to the [name](#) (page 11) method. Hard-coding the service names is not recommended.

**Return Value**

The service specified by *name*.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

+ [allServices](#) (page 6)

- [name](#) (page 11)

**Declared In**  
IMService.h

## Instance Methods

### infoForAllScreenNames

Returns information about all buddies for the service.

- (NSArray \*)infoForAllScreenNames

#### Return Value

The dictionaries returned by [infoForScreenName:](#) (page 10) for all buddies.

#### Discussion

If the current user has multiple buddies for the same person (determined by the user's Address Book), this method returns the information for all of the accounts belonging to that person.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [infoForPreferredScreenNames](#) (page 9)
- [infoForScreenName:](#) (page 10)

**Declared In**  
IMService.h

### infoForPreferredScreenNames

Returns information about just the preferred accounts for all buddies.

- (NSArray \*)infoForPreferredScreenNames

#### Return Value

An array of the dictionaries returned by [infoForScreenName:](#) (page 10) for all preferred accounts.

#### Discussion

If the current user has multiple buddies for the same person (determined by the user's Address Book), this method returns only the information for the preferred accounts belonging to that person. The preferred account is determined by iChat, using a combination of capabilities (video chat capability, audio chat capability, and so on), status (available, idle, away), and other user attributes.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [infoForAllScreenNames](#) (page 9)
- [infoForScreenName:](#) (page 10)

**Declared In**  
IMService.h

## infoForScreenName:

Returns information about a buddy with the specified screen name.

```
- (NSDictionary *)infoForScreenName:(NSString *)screenName
```

### Parameters

*screenName*

A screen name for a buddy.

### Return Value

Information about a buddy with the specified screen name. See “[Screen Name Properties](#)” (page 12) for the key-value pairs that appear in this dictionary.

### Availability

Available in Mac OS X v10.4 and later.

### See Also

- [infoForAllScreenNames](#) (page 9)
- [infoForPreferredScreenNames](#) (page 9)

**Declared In**  
IMService.h

## localizedName

Returns the user-visible localized name of the service.

```
- (NSString *)localizedName
```

### Return Value

The user-visible localized name of the service, such as "AOL Instant Messenger" or "Bonjour".

### Availability

Available in Mac OS X v10.4 and later.

### See Also

- [localizedShortName](#) (page 10)
- [name](#) (page 11)

**Declared In**  
IMService.h

## localizedShortName

Returns a short version, if available, of the user-visible localized name of the service.

```
- (NSString *)localizedShortName
```

**Return Value**

The user-visible short localized name of the service, such as "AOL".

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

- [localizedName](#) (page 10)
- [name](#) (page 11)

**Declared In**

IMService.h

**name**

Returns the fixed canonical name of the service.

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

**Return Value**

The fixed canonical name of the service. This string is not intended to be visible to the user and therefore is not localized.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

- [localizedName](#) (page 10)
- [localizedShortName](#) (page 10)

**Declared In**

IMService.h

**peopleWithScreenName:**

Returns Address Book entries that match the specified screen name of a buddy.

```
- (NSArray *)peopleWithScreenName:(NSString *)screenName
```

**Parameters**

*screenName*

The screen name of a buddy.

**Return Value**

An array of Address Book entries that match the specified screen name of a buddy. Returns an empty array if there is no match.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

- [screenNamesForPerson:](#) (page 12)

**Related Sample Code**

ABPresence

**Declared In**

IMService.h

**screenNamesForPerson:**

Returns an array of strings that are valid screen names for the specified person.

- (NSArray \*)screenNamesForPerson:(ABPerson \*)*person***Parameters***person*

An entry in the Address Book.

**Return Value**

An array of valid screen names for the specified person. Returns an empty array if there is no match.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**- [peopleWithScreenName:](#) (page 11)**Declared In**

IMService.h

**status**

Returns the login status of the service.

- (IMServiceStatus)status

**Return Value**The login status of the service. One of the constants described in “[IMServiceStatus](#)” (page 15).**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

IMService.h

## Constants

### Screen Name Properties

Keys for information about a person logged in to an instant message service—specifically, a buddy that appears in the user’s buddy list:

```
extern NSString *IMPPersonAVBusyKey;
extern NSString *IMPPersonCapabilitiesKey;
extern NSString *IMPPersonEmailKey;
extern NSString *IMPPersonFirstNameKey;
extern NSString *IMPPersonIdleSinceKey;
extern NSString *IMPPersonLastNameKey;
extern NSString *IMPPersonPictureDataKey;
extern NSString *IMPPersonScreenNameKey;
extern NSString *IMPPersonServiceNameKey;
extern NSString *IMPPersonStatusKey;
extern NSString *IMPPersonStatusMessageKey;
```

### Constants

#### IMPPersonAVBusyKey

Used to obtain a person's busy status. The value is an `NSNumber` set to 0 if the person's audio/video capabilities are available, or 1 if they are busy.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

#### IMPPersonCapabilitiesKey

Used to obtain a person's iChat capabilities. The value is an `NSArray` of capability properties. See [“Person Capability Values”](#) (page 14) for more information.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

#### IMPPersonEmailKey

Used to obtain a person's email address. The value is an `NSString` containing the person's email address. This is a key used directly by Bonjour; however, if a person has an Address Book entry associated with a relevant AIM account, this key reflects the first email address of that person.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

#### IMPPersonFirstNameKey

Used to obtain a person's first name. The value is an `NSString` containing the person's first name. This is a key used directly by Bonjour; however, if a person has an Address Book entry associated with a relevant AIM account, this key reflects the first name of that person.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

#### IMPPersonIdleSinceKey

Used to obtain a person's idle status. The value is an `NSDate` containing the time, in seconds, since the last user activity. Available if the person's status is idle.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

#### IMPPersonLastNameKey

Used to obtain a person's last name. The value is an `NSString` containing the person's last name. This is a key used directly by Bonjour; however, if a person has an Address Book entry associated with a relevant AIM account, this key reflects the last name of that person.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

**IMPersonPictureDataKey**

Used to obtain a person's image. The value is an `NSData` containing the image for the person's icon.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

**IMPersonScreenNameKey**

Used to obtain a person's screen name. The value is an `NSString` containing the service-specific identifier for a person. For example, "User123" or "steve@mac.com" for AIM, and "John Doe" for Bonjour.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

**IMPersonServiceNameKey**

Used to obtain a person's service name. The value is an `NSString` containing the name of the service this person belongs to.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

**IMPersonStatusKey**

Used to obtain a person's online status. The value is an `NSNumber` representing the current online status of the person, if known. See "[IMPersonStatus](#)" (page 16) for more information.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

**IMPersonStatusMessageKey**

Used to obtain a person's status message. The value is an `NSString` containing the person's current status message.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

**Discussion**

These keys appear in the dictionary returned by the [infoForScreenName:](#) (page 10) method.

**Declared In**

`InstantMessage/IMService.h`

## Person Capability Values

A person's iChat capabilities accessed using the [IMPersonCapabilitiesKey](#) (page 13) key.

```
extern NSString *IMCapabilityAudioConference;
extern NSString *IMCapabilityDirectIM;
extern NSString *IMCapabilityFileSharing;
extern NSString *IMCapabilityFileTransfer;
extern NSString *IMCapabilityText;
extern NSString *IMCapabilityVideoConference;
```

**Constants****IMCapabilityAudioConference**

A person has audio chat capability.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

IMCapabilityDirectIM

A person has direct connect capability.

Available in Mac OS X v10.4 and later.

Declared in IMService.h.

IMCapabilityFileSharing

A person has file sharing capability.

Available in Mac OS X v10.4 and later.

Declared in IMService.h.

IMCapabilityFileTransfer

A person has file transfer capability.

Available in Mac OS X v10.4 and later.

Declared in IMService.h.

IMCapabilityText

A person has text capability.

Available in Mac OS X v10.4 and later.

Declared in IMService.h.

IMCapabilityVideoConference

A person has video chat capability.

Available in Mac OS X v10.4 and later.

Declared in IMService.h.

**Declared In**

InstantMessage/IMService.h

## IMServiceStatus

The states of a service.

```
enum {
    IMServiceStatusLoggedOut,
    IMServiceStatusDisconnected,
    IMServiceStatusLoggingOut,
    IMServiceStatusLoggingIn,
    IMServiceStatusLoggedIn
};
typedef NSUInteger IMServiceStatus;
```

**Constants**

IMServiceStatusLoggedOut

A service is currently logged out.

Available in Mac OS X v10.4 and later.

Declared in IMService.h.

IMServiceStatusDisconnected

A service was disconnected, not by the user but by the system or because of an error.

Available in Mac OS X v10.4 and later.

Declared in IMService.h.

`IMServiceStatusLoggingOut`  
**A service is in the process of logging out.**  
**Available in Mac OS X v10.4 and later.**  
**Declared in `IMService.h`.**

`IMServiceStatusLoggingIn`  
**A service is in the process of logging in.**  
**Available in Mac OS X v10.4 and later.**  
**Declared in `IMService.h`.**

`IMServiceStatusLoggedIn`  
**A service is currently logged in.**  
**Available in Mac OS X v10.4 and later.**  
**Declared in `IMService.h`.**

**Declared In**

`InstantMessage/IMService.h`

## IMPersonStatus

The state of a person across all services.

```
enum {
    IMPersonStatusUnknown,
    IMPersonStatusOffline,
    IMPersonStatusIdle,
    IMPersonStatusAway,
    IMPersonStatusAvailable,
    IMPersonStatusNoStatus
};
typedef NSUInteger IMPersonStatus;
```

**Constants**

`IMPersonStatusUnknown`  
**The person's status is unknown.**  
**Available in Mac OS X v10.4 and later.**  
**Declared in `IMService.h`.**

`IMPersonStatusOffline`  
**The person is currently offline.**  
**Available in Mac OS X v10.4 and later.**  
**Declared in `IMService.h`.**

`IMPersonStatusIdle`  
**The person is currently idle.**  
**Available in Mac OS X v10.4 and later.**  
**Declared in `IMService.h`.**

`IMPersonStatusAway`  
**The person is currently away.**  
**Available in Mac OS X v10.4 and later.**  
**Declared in `IMService.h`.**

**IMPersonStatusAvailable**

The person is currently available.

Available in Mac OS X v10.4 and later.

Declared in `IMService.h`.

**IMPersonStatusNoStatus**

No status is available.

Available in Mac OS X v10.5 and later.

Declared in `IMService.h`.

**Discussion**

This is accessed using the `IMPersonStatusKey` (page 14) key for a buddy or returned by the `myStatus` (page 7) method for the current user.

**Declared In**

`InstantMessage/IMService.h`

## Notifications

### **IMMyStatusChangedNotification**

Posted by the `IMService` custom notification center when the local user changes their online status. The notification object is an `IMService` object. The user information dictionary does not contain keys. The receiver should send `myStatus` (page 7) to the notification object to get the new online status.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

`IMService.h`

### **IMPersonInfoChangedNotification**

Posted by the `IMService` custom notification center when a screen name changes some aspect of its published information. The notification object is an `IMService` object. The user information dictionary always contains the `IMPersonServiceNameKey` (page 14) key and may contain any of the other keys as described in “[Screen Name Properties](#)” (page 12). If a particular attribute is removed, the value for the relevant key is `NSNull`.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

`IMService.h`

### **IMPersonStatusChangedNotification**

Posted by the `IMService` custom notification center when a different buddy (screen name) logs in, logs off, goes away, and so on. The notification object is an `IMService` object. The user information dictionary always contain the `IMPersonScreenNameKey` (page 14) and `IMPersonStatusKey` (page 14) keys, and no others.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

IMService.h

**IMServiceStatusChangedNotification**

Posted by the `IMService` custom notification center when the status of a service changes—the current user logs in, logs off, goes away, and so on. The notification object is an `IMService` object. The user information dictionary does not contain keys. The receiver should send `status` (page 12) to the notification object to get the new service status.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

IMService.h

**IMStatusImagesChangedAppearanceNotification**

Posted by the `IMService` custom notification center when the current user changes his or her preferred images for displaying status. The notification object is `nil`. This notification does not contain a user information dictionary. Use the `imageNameForStatus:` (page 7) method to get the new images.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

IMService.h

# Deprecated IMService Methods

---

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

## Deprecated in Mac OS X v10.5

### **imageURLForStatus:**

Returns the URL of the image for the specified status of a person. (Deprecated in Mac OS X v10.5.)

```
+ (NSURL *)imageURLForStatus:(IMPersonStatus)status
```

#### **Parameters**

*status*

The status of a person. See “[IMPersonStatus](#)” (page 16) for possible values.

#### **Return Value**

An image that reflects the current online status of a person; the image is usually a colored bubble or triangle.

#### **Availability**

Available in Mac OS X v10.4 and later.

Deprecated in Mac OS X v10.5.

#### **See Also**

+ [imageNameForStatus:](#) (page 7)

#### **Declared In**

IMService.h



# Document Revision History

---

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

Date	Notes
2009-07-17	Minor edits throughout.
2008-10-15	Minor edit.
2007-07-10	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.
	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History

# Index

---

## A

---

allServices **class method** 6

## I

---

imageNameForStatus: **class method** 7  
imageURLForStatus: **class method** 19  
IMCapabilityAudioConference **constant** 14  
IMCapabilityDirectIM **constant** 15  
IMCapabilityFileSharing **constant** 15  
IMCapabilityFileTransfer **constant** 15  
IMCapabilityText **constant** 15  
IMCapabilityVideoConference **constant** 15  
IMMyStatusChangedNotification **notification** 17  
IMPersonAVBusyKey **constant** 13  
IMPersonCapabilitiesKey **constant** 13  
IMPersonEmailKey **constant** 13  
IMPersonFirstNameKey **constant** 13  
IMPersonIdleSinceKey **constant** 13  
IMPersonInfoChangedNotification **notification** 17  
IMPersonLastNameKey **constant** 13  
IMPersonPictureDataKey **constant** 14  
IMPersonScreenNameKey **constant** 14  
IMPersonServiceNameKey **constant** 14  
**IMPersonStatus** 16  
IMPersonStatusAvailable **constant** 17  
IMPersonStatusAway **constant** 16  
IMPersonStatusChangedNotification **notification** 17  
IMPersonStatusIdle **constant** 16  
IMPersonStatusKey **constant** 14  
IMPersonStatusMessageKey **constant** 14  
IMPersonStatusNoStatus **constant** 17  
IMPersonStatusOffline **constant** 16  
IMPersonStatusUnknown **constant** 16  
**IMServiceStatus** 15  
IMServiceStatusChangedNotification **notification** 18  
IMServiceStatusDisconnected **constant** 15

IMServiceStatusLoggedIn **constant** 16  
IMServiceStatusLoggedOut **constant** 15  
IMServiceStatusLoggingIn **constant** 16  
IMServiceStatusLoggingOut **constant** 16  
IMStatusImagesChangedAppearanceNotification **notification** 18  
infoForAllScreenNames **instance method** 9  
infoForPreferredScreenNames **instance method** 9  
infoForScreenName: **instance method** 10

## L

---

localizedName **instance method** 10  
localizedShortName **instance method** 10

## M

---

myIdleTime **class method** 7  
myStatus **class method** 7

## N

---

name **instance method** 11  
notificationCenter **class method** 8

## P

---

peopleWithScreenName: **instance method** 11  
**Person Capability Values** 14

## S

---

**Screen Name Properties** 12  
screenNamesForPerson: **instance method** 12

serviceName: class method [8](#)  
status instance method [12](#)