Instant Message Reference Update

Cocoa > Apple Applications

2007-07-18

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

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

Introduction to Instant Message Reference Update

This document summarizes the symbols that have been added to the Instant Message framework. The full reference documentation notes in what version a symbol was introduced, but sometimes it's useful to see only the new symbols for a given release.

If you are not familiar with this framework you should refer to the complete framework reference documentation.

Organization of This Document

Symbols are grouped by class or protocol for Objective-C and by header file for C. For each symbol there is a link to complete documentation, if available, and a brief description, if available.

See Also

For reference documentation on this framework, see *Instant Message Framework Reference*.

10.5 Symbol Changes

This article lists the symbols added to `InstantMessage.framework` in Mac OS X v10.5.

Classes

All of the classes with new symbols are listed alphabetically, with their new class, instance, and delegate methods described.

IMAVManager (New)

Complete reference information is available in the `IMAVManager` reference.

Class Methods

`sharedAVManager`	Returns the shared instance of the IMAVManager object, creating it if the object doesn't exist yet.

Instance Methods

`audioDeviceChannels`	Returns an array of audio device channel numbers used by the receiver.
`audioDeviceUID`	Returns the audio device UID.
`numberOfAudioChannels`	Returns the number of audio channels.
`setNumberOfAudioChannels:`	Sets the number of audio channels.
`setVideoDataSource:`	Sets the receiver's video data source object that provides video data to iChat AV.
`setVideoOptimizationOptions:`	Sets the video optimization options.
`start`	Starts sending audio and video to iChat AV.
`state`	Returns the current state of the receiver.
`stop`	Stops sending audio and video to iChat AV.
`videoDataSource`	Returns the receiver's video data source object.

`videoOptimizationOptions`	Returns the video optimization options.

IMService

Complete reference information is available in the `IMService` reference.

Class Methods

`imageNameForStatus:`	Returns the name of the image for the specified status of a person.

Protocols

All of the protocols with new symbols are listed alphabetically, with their new methods described.

IMVideoDataSource

Complete reference information is available in the `IMVideoDataSource` reference.

Instance Methods

`getOpenGLBufferContext:pixelFormat:`	
`getPixelBufferPixelFormat:`	
`renderIntoOpenGLBuffer:onScreen:forTime:`	
`renderIntoPixelBuffer:forTime:`	

C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

IMAVManager.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`IMAVInactive`	An IMAVManager object is not available to send audio/video to iChat AV because the user has not started a session.
`IMAVManagerState`	The state of an IMAVManager object.
`IMAVManagerStateChangedNotification`	Posted by the IMService class custom notification center when the iChat AV input state changes.
`IMAVNotAvailable`	
`IMAVPending`	iChat AV is not ready to receive content from an IMAVManager object.
`IMAVRequested`	The user selected this client to begin iChat Theater. The client should send start to the IMAVManager object to begin an iChat Theater session.
`IMAVRunning`	An IMAVManager object is actively sending audio/video content to iChat AV.
`IMAVShuttingDown`	An IMAVManager object is shutting down and will soon change to the IMAVInactive state.
`IMAVStartingUp`	An IMAVManager object is starting up and will soon change to the IMAVPending or IMAVRunning state.
`IMAVStopped`	
`IMVideoOptimizationDefault`	Shared video is played alongside the user's local video, and the video is full-motion. This is the default.
`IMVideoOptimizationOptions`	The characteristics of the video source to allow for optimization of CPU and bandwidth usage.
`IMVideoOptimizationReplacement`	Do not send the user's local video, instead devote full CPU and bandwidth resources to the shared video.
`IMVideoOptimizationStills`	Shared video remains unchanged for many sequential frames (such as a photo slideshow). This is a hint that the required bandwidth is lower than that of full-motion video. Incorrectly setting this option may result in poor video quality.

IMService.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`IMComparePersonStatus`	Compares two IMPersonStatus values to determine which one has a higher availablity.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`IMMyStatusChangedNotification`	Posted by the IMService custom notification center when the local user changes their online status.

10.4 Symbol Changes

This article lists the symbols added to `InstantMessage.framework` in Mac OS X v10.4.

Classes

All of the classes with new symbols are listed alphabetically, with their new class, instance, and delegate methods described.

IMService (New)

Complete reference information is available in the `IMService` reference.

Class Methods

`allServices`	Returns an array of the currently available services.
`imageURLForStatus:`	Returns the URL of the image for the specified status of a person.
`myIdleTime`	Returns the number of seconds that the current user is idle.
`myStatus`	Returns the status of the current user.
`notificationCenter`	Returns the custom notification center for the service.
`serviceWithName:`	Returns the specified service.

Instance Methods

`infoForAllScreenNames`	Returns information about all buddies for the service.
`infoForPreferredScreenNames`	Returns information about just the preferred accounts for all buddies.
`infoForScreenName:`	Returns information about a buddy with the specified screen name.
`localizedName`	Returns the user-visible localized name of the service.
`localizedShortName`	Returns a short version, if available, of the user-visible localized name of the service.
`name`	Returns the fixed canonical name of the service.

peopleWithScreenName:	Returns Address Book entries that match the specified screen name of a buddy.
screenNamesForPerson:	Returns an array of strings that are valid screen names for the specified person.
status	Returns the login status of the service.

C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

IMService.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

IMCapabilityAudioConference	A person has audio chat capability.
IMCapabilityDirectIM	A person has direct connect capability.
IMCapabilityFileSharing	A person has file sharing capability.
IMCapabilityFileTransfer	A person has file transfer capability.
IMCapabilityText	A person has text capability.
IMCapabilityVideoConference	A person has video chat capability.
IMPersonAVBusyKey	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.
IMPersonCapabilitiesKey	Used to obtain a person’s iChat capabilities. The value is an NSArray of capability properties. See “Person Capability Values” for more information.
IMPersonEmailKey	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.

IMPersonFirstNameKey	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.
IMPersonIdleSinceKey	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.
IMPersonInfoChangedNotification	Posted by the IMService custom notification center when a screen name changes some aspect of its published information.
IMPersonLastNameKey	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.
IMPersonPictureDataKey	Used to obtain a person’s image. The value is an NSData containing the image for the person’s icon.
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.
IMPersonServiceNameKey	Used to obtain a person’s service name. The value is an NSString containing the name of the service this person belongs to.
IMPersonStatus	The state of a person across all services.
IMPersonStatusAvailable	The person is currently available.
IMPersonStatusAway	The person is currently away.
IMPersonStatusChangedNotification	Posted by the IMService custom notification center when a different buddy (screen name) logs in, logs off, goes away, and so on.
IMPersonStatusIdle	The person is currently idle.
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” for more information.

IMPersonStatusMessageKey	Used to obtain a person's status message. The value is an NSString containing the person's current status message.
IMPersonStatusOffline	The person is currently offline.
IMPersonStatusUnknown	The person's status is unknown.
IMServiceStatus	The states of a service.
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.
IMServiceStatusDisconnected	A service was disconnected, not by the user but by the system or because of an error.
IMServiceStatusLoggedIn	A service is currently logged in.
IMServiceStatusLoggedOut	A service is currently logged out.
IMServiceStatusLoggingIn	A service is in the process of logging in.
IMServiceStatusLoggingOut	A service is in the process of logging out.
IMStatusImagesChangedAppearanceNotification	Posted by the IMService custom notification center when the current user changes his or her preferred images for displaying status.

Document Revision History

This table describes the changes to *Instant Message Reference Update*.

Date	Notes
2007-07-18	Updated with the symbols added to the Instant Message framework in Mac OS X v10.5.
2005-04-29	Added information about additional documentation on the Instant Message framework.
	New document that summarizes the symbols added to the Instant Message framework in Mac OS X v10.4.