
Input Method Kit Reference Update

[Cocoa > Internationalization](#)



2007-07-18



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, Mac OS, and Objective-C 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

Introduction to Input Method Kit Reference Update 5

Organization of This Document 5

See Also 5

10.5 Symbol Changes 7

Classes 7

IMKCandidates (New) 7

IMKInputController (New) 8

IMKServer (New) 9

NSObject 9

Protocols 10

IMKMouseHandling (New) 10

IMKStateSetting (New) 10

C Symbols 11

IMKCandidates.h 11

IMKInputController.h 11

IMKServer.h 12

Document Revision History 13

Introduction to Input Method Kit Reference Update

This document summarizes the symbols that have been added to the Input Method Kit 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 *Input Method Kit Framework Reference*.

10.5 Symbol Changes

This article lists the symbols added to `InputMethodKit.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.

IMKCandidates (New)

Complete reference information is available in the [IMKCandidates reference](#).

Instance Methods

<code>attributes</code>	Returns a dictionary of the style attributes used for the candidates window..
<code>dismissesAutomatically</code>	Returns the state of the flag that determines whether the candidates window dismisses automatically.
<code>hide</code>	Hides a candidates window, if it is visible.
<code>initWithServer:panelType:</code>	Returns the initialized IMKCandidates object.
<code>isVisible</code>	Returns whether or not the candidates window is visible.
<code>panelType</code>	Returns the style of the candidates window.
<code>selectionKeys</code>	Returns an array of NSNumber objects where each NSNumber object represents a virtual key code.
<code>selectionKeysKeyLayout</code>	Returns the key layout that maps virtual key codes to selection keys.
<code>setAttributes:</code>	Sets the style attributes for the candidates window.
<code>setDismissesAutomatically:</code>	Sets the state of the flag that determines whether the candidates window dismisses automatically.
<code>setPanelType:</code>	Sets the style of the candidates window.
<code>setSelectionKeys:</code>	Sets the selection keys for the candidates.

<code>setSelectionKeysKeylayout:</code>	Sets the key layout that is used to map virtual key codes to characters.
<code>show:</code>	Shows the candidates window.
<code>showAnnotation:</code>	Displays an annotation string in an annotation window.
<code>updateCandidates</code>	Updates the candidates that are displayed in the candidates window.

IMKInputController (New)

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

Instance Methods

<code>annotationSelected:forCandidate:</code>	Sends the selected candidate string and annotation string to the input controller.
<code>cancelComposition</code>	Stops the current composition and replaces marked text with the original text.
<code>candidateSelected:</code>	Informs an input controller that a new candidate is selected.
<code>candidateSelectionChanged:</code>	Informs an input controller that the current candidate selection in the candidate window has changed.
<code>client</code>	Returns the client object associated with the input controller.
<code>compositionAttributesAtRange:</code>	Returns a dictionary of text attributes.
<code>delegate</code>	
<code>doCommandBySelector:commandDictionary:</code>	Passes commands that are not generated as part of the text input process.
<code>hidePalettes</code>	Informs an input method that it should close any visible user interface.
<code>initWithServer:delegate:client:</code>	Initializes the input control by setting the delegate.
<code>markForStyle:atRange:</code>	Returns a dictionary of text attributes that can mark a range of an attributed string to send to a client.
<code>menu</code>	Returns a menu of commands that are specific to an input method.
<code>replacementRange</code>	Returns the range in the client document that the text should replace.

<code>selectionRange</code>	Returns where the range of the selection that should be placed inside marked text.
<code>server</code>	Returns the server object that manages the input controller.
<code>setDelegate:</code>	Sets the delegate for input controller object.
<code>updateComposition</code>	Informs the input controller that the composition has changed.

IMKServer (New)

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

Instance Methods

<code>bundle</code>	Returns an <code>NSBundle</code> object for the input method.
<code>initWithName:bundleIdentifier:</code>	Creates and returns a server object from property list information contained in the provided bundle.
<code>initWithName:controllerClass:delegateClass:</code>	Creates and returns a server object initialized with the provided parameters.

NSObject

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

Instance Methods

<code>candidates:</code>	Returns an array of candidates.
<code>commitComposition:</code>	Informs the controller that the composition should be committed.
<code>composedString:</code>	Return the current composed string.
<code>didCommandBySelector:client:</code>	Processes a command generated by user action such as typing certain keys or pressing the mouse button.
<code>handleEvent:client:</code>	Handles key down and mouse events.
<code>inputText:client:</code>	Handles key down events that do not map to an action method.

<code>inputText:key:modifiers:client:</code>	Receives Unicode, the key code that generated it, and any modifier flags.
<code>originalString:</code>	Return the a string that consists of the precomposed unicode characters.

Protocols

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

IMKMouseHandling (New)

Complete reference information is available in the [IMKMouseHandling](#) reference.

Instance Methods

<code>mouseDownOnCharacterIndex:coordinate:withModifier:continueTracking:client:</code>	Handles mouse-down event send to an input method.
<code>mouseMovedOnCharacterIndex:coordinate:withModifier:client:</code>	Handles a mouse-moved event sent to an input method.
<code>mouseUpOnCharacterIndex:coordinate:withModifier:client:</code>	Handles a mouse-up event sent to an input method.

IMKStateSetting (New)

Complete reference information is available in the [IMKStateSetting](#) reference.

Instance Methods

<code>activateServer:</code>	Activates the input method server.
<code>deactivateServer:</code>	Deactivates the input method server.
<code>modes:</code>	Returns the modes dictionary associated with the input method.
<code>recognizedEvents:</code>	Returns an unsigned integer that contains a union of event masks
<code>setValue:forTag:client:</code>	Set the value for the provided key.
<code>showPreferences:</code>	Displays a preferences window.
<code>valueForTag:client:</code>	Returns a value object whose key is the provided tag.

C Symbols

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

IMKCandidates.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.

IMKCandidatePanelType	Types of candidates windows provide by the Input Method Kit.
IMKCandidatesLocationHint	Hints that suggest where to place the candidates window.
IMKCandidatesOpacityAttributeName	The opacity attribute for a candidates window.
kIMKLocateCandidatesAboveHint	Place the candidates window above the start of the current text selection.
kIMKLocateCandidatesBelowHint	Place the candidates window below the start of the current text selection.
kIMKLocateCandidatesLeftHint	Place the candidates window to the left of the current text selection.
kIMKLocateCandidatesRightHint	Place the candidates window to the right of the current text selection.
kIMKScrollingGridCandidatePanel	A window that displays a grid and can scroll if necessary.
kIMKSingleColumnScrollingCandidatePanel	A window that displays one column and can scroll if necessary.
kIMKSingleRowSteppingCandidatePanel	A window that displays a single row.

IMKInputController.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.

kIMKCommandClientName	Used to look up the client object; the client conforms to the IMKInputText and NSObject protocols.
kIMKCommandMenuItemName	Used to look up the NSMenuItem object that is passed to menu item actions.

IMKServer.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.

IMKControllerClass	The key used to find an input method input controller class name from the input method bundle.
IMKDelegateClass	The key used to find an input method delegate class name from the input method bundle.
IMKModeDictionary	The key used to obtain an input method mode dictionary from the input method bundle.

Document Revision History

This table describes the changes to *Input Method Kit Reference Update*.

Date	Notes
2007-07-18	Updated with the symbols added to the Input Method Kit framework in Mac OS X v10.5.

