
IMKStateSetting Protocol Reference

[Cocoa > Internationalization](#)



2007-06-05



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

IMKStateSetting Protocol Reference 5

Overview	5
Tasks	5
Activating and Deactivating the Server	5
Showing a Preferences Window	5
Getting the Supported Events	5
Getting the Mode Dictionary	6
Getting and Setting Values	6
Instance Methods	6
activateServer:	6
deactivateServer:	6
modes:	7
recognizedEvents:	7
setValue:forTag:client:	8
showPreferences:	8
valueForTag:client:	8

Document Revision History 11

Index 13

IMKStateSetting Protocol Reference

Adopted by	IMKInputController
Framework	System/Library/Frameworks/InputMethodKit.framework
Availability	Available in Mac OS X v10.5 and later.
Declared in	IMKInputController.h

Overview

The `IMKStateSetting` protocol defines methods for setting or accessing values that indicate the state of an input method.

Tasks

Activating and Deactivating the Server

- [activateServer:](#) (page 6)
Activates the input method server.
- [deactivateServer:](#) (page 6)
Deactivates the input method server.

Showing a Preferences Window

- [showPreferences:](#) (page 8)
Displays a preferences window.

Getting the Supported Events

- [recognizedEvents:](#) (page 7)
Returns an unsigned integer that contains a union of event masks

Getting the Mode Dictionary

- [modes:](#) (page 7)
Returns the modes dictionary associated with the input method.

Getting and Setting Values

- [valueForTag:client:](#) (page 8)
Returns a value object whose key is the provided tag.
- [setValue:forTag:client:](#) (page 8)
Set the value for the provided key.

Instance Methods

activateServer:

Activates the input method server.

```
(void)activateServer:(id)sender
```

Parameters

sender

The object sending the activation message.

Availability

Available in Mac OS X v10.5 and later.

See Also

- [deactivateServer:](#) (page 6)

Declared In

IMKInputController.h

deactivateServer:

Deactivates the input method server.

```
(void)deactivateServer:(id)sender
```

Parameters

sender

The object sending the deactivation message.

Availability

Available in Mac OS X v10.5 and later.

See Also

- [activateServer:](#) (page 6)

Declared In

IMKInputController.h

modes:

Returns the modes dictionary associated with the input method.

- (NSDictionary*)modes:(id)sender

Parameters*sender*

The client object requesting the modes dictionary.

Return Value

The modes dictionary associated with the input method. The dictionary should be an autoreleased object.

Discussion

Typically a client object calls this method to build the text input menu. By calling the input method rather than reading the modes from the `Info.plist` file, the input method can dynamically modify the modes supported.

Availability

Available in Mac OS X v10.5 and later.

Declared In

IMKInputController.h

recognizedEvents:

Returns an unsigned integer that contains a union of event masks

- (NSUInteger)recognizedEvents:(id)sender

Parameters*sender*

The client object requesting the supported events.

Return ValueAn unsigned integer that contains a union of event masks (See the `NSEvent.h` header file).**Discussion**

A client calls this method to check whether an input method supports an event. The default implementation returns `NSKeyDownMask`. If your input method handles only key down events, the Input Method Kit provides the default mouse handling. The default mouse-down handling behavior is as follows: If there is an active composition area and the user clicks in the text but outside of the composition area, the Input Method Kit sends your input method a `commitComposition:` message. This happens only for input methods that return only the default value—`NSKeyDownMask`.

Availability

Available in Mac OS X v10.5 and later.

Declared In

IMKInputController.h

setValue:forTag:client:

Set the value for the provided key.

```
- (void)setValue:(id)value forTag:(long)tag client:(id)sender
```

Parameters

value

The value, specified as the appropriate object (such as `NSNumber`), to set.

tag

The key whose value you want to set.

sender

The client setting the value.

Availability

Available in Mac OS X v10.5 and later.

See Also

- [valueForTag:client:](#) (page 8)

Declared In

IMKInputController.h

showPreferences:

Displays a preferences window.

```
- (void)showPreferences:(id)sender
```

Parameters

sender

The object sending the message to show the preference window.

Discussion

This method looks for a nib file that contains a window controller class and a preferences utility. If found, it displays the window. To use this method you must create a menu item in your input method menu whose action is `showPreferences:`. When a user selects that item, the Input Method Kit invokes your `showPreferences:` method. The default implementation looks for a nib file named `preferences.nib`. If found, it allocates a window controller class loads the nib file. You can provide a custom window controller class by naming the class in your input method `info.plist` file, providing a key-value pair. The key must be `InputMethodServerPreferencesWindowControllerClass` and the associated value must be the name of your custom class.

Availability

Available in Mac OS X v10.5 and later.

Declared In

IMKInputController.h

valueForTag:client:

Returns a value object whose key is the provided tag.

- (id)valueForTag:(long)tag client:(id)sender

Parameters

tag

The key whose value you want to retrieve.

sender

The client requesting the value.

Return Value

The value object. The returned object should be autoreleased.

Availability

Available in Mac OS X v10.5 and later.

See Also

- [setValue:forTag:client:](#) (page 8)

Declared In

IMKInputController.h

Document Revision History

This table describes the changes to *IMKStateSetting Protocol Reference*.

Date	Notes
2007-06-05	New document that describes the protocol used to set or access state values for an input method client session.

REVISION HISTORY

Document Revision History

Index

A

activateServer: protocol instance method [6](#)

D

deactivateServer: protocol instance method [6](#)

M

modes: protocol instance method [7](#)

R

recognizedEvents: protocol instance method [7](#)

S

setValue:forTag:client: protocol instance method [8](#)

showPreferences: protocol instance method [8](#)

V

valueForTag:client: protocol instance method [8](#)