NSSearchFieldCell Class Reference

Cocoa > User Experience

2008-03-11

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

NSSearchFieldCell Class Reference

Inherits from	NSTextFieldCell : NSActionCell : NSCell : NSObject
Conforms to	NSCoding (NSCell) NSCopying (NSCell) NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.3 and later.
Companion guide	Search Fields
Declared in	NSSearchFieldCell.h

Overview

The `NSSearchFieldCell` class defines the programmatic interface for text fields that are optimized for text-based searches. An `NSSearchFieldCell` object is "wrapped" by an NSSearchField control object, which directly inherits from the `NSTextField` class. The search field implemented by these classes presents a standard user interface for searches, including a search button, a cancel button, and a pop-up icon menu for listing recent search strings and custom search categories.

When the user types and then pauses, the cell's action message is sent to its target. You can query the cell's string value for the current text to search for. Do not rely on the sender of the action to be an `NSMenu` object because the menu may change. If you need to change the menu, modify the search menu template and call the `setSearchMenuTemplate:` method to update.

Tasks

Managing Buttons

- `setSearchButtonCell:` (page 14)
  Sets the button cell used to display the search-button image
- `searchButtonCell` (page 10)
  Returns the button cell used to display the search-button image.
- `resetSearchButtonCell` (page 9)
  Resets the search button cell to its default attributes.

- `setCancelButtonCell:` (page 12)
  Sets the button cell object used to display the cancel-button image
- `cancelButtonCell` (page 7)
  Returns the button cell object used to display the cancel-button image.
- `resetCancelButtonCell` (page 9)
  Resets the cancel button cell to its default attributes.

Custom Layout

- `searchTextRectForBounds:` (page 11)
  Modifies the bounding rectangle for the search-text field cell.
- `searchButtonRectForBounds:` (page 10)
  Modifies the bounding rectangle for the search button cell.
- `cancelButtonRectForBounds:` (page 7)
  Modifies the bounding rectangle for the cancel button cell.

Managing Menu Templates

- `setSearchMenuTemplate:` (page 14)
  Sets the menu template object used to dynamically construct the receiver's pop-up icon menu.
- `searchMenuTemplate` (page 11)
  Returns the menu template object used to dynamically construct the search pop-up icon menu.

Managing Search Modes

- `setSendsWholeSearchString:` (page 15)
  Sets whether the receiver sends the search action message when the user clicks the search button (or presses return) or after each keystroke.
- `sendsWholeSearchString` (page 12)
  Returns a Boolean value indicating whether the receiver sends the search action message when the user clicks the search button (or presses return) or after each keystroke.
- `sendsSearchStringImmediately` (page 12)
  Returns a Boolean value indicating whether the receiver sends its action immediately upon being notified of changes to the search field text or after a brief pause.
- `setSendsSearchStringImmediately:` (page 15)
  Sets whether the cell sends its action message to the target immediately upon notification of any changes to the search field text or after a brief pause.

Managing Recent Search Strings

- `setMaximumRecents:` (page 13)
  Sets the maximum number of search strings that can appear in the search menu

- maximumRecents (page 8)
  Returns the maximum number of recent search strings to display in the custom search menu.
- setRecentSearches: (page 14)
  Sets the list of recent search strings to list in the pop-up icon menu of the receiver.
- recentSearches (page 9)
  Returns the list of recent search strings for the control.
- setRecentsAutosaveName: (page 13)
  Sets the autosave name under which the receiver automatically archives the list of recent search strings.
- recentsAutosaveName (page 8)
  Returns the key under which the prior list of recent search strings has been archived.

Instance Methods

cancelButtonCell

Returns the button cell object used to display the cancel-button image.

```
- (NSButtonCell *)cancelButtonCell
```

Return Value
The cancel button cell.

Availability
Available in Mac OS X v10.3 and later.

See Also
- setCancelButtonCell: (page 12)
- resetCancelButtonCell (page 9)

Declared In
`NSSearchFieldCell.h`

cancelButtonRectForBounds:

Modifies the bounding rectangle for the cancel button cell.

```
- (NSRect)cancelButtonRectForBounds:(NSRect)rect
```

Parameters

rect
  The current bounding rectangle for the cancel button.

Return Value
The updated bounding rectangle to use for the cancel button. The default value is the value passed into the *rect* parameter.

Discussion

Subclasses can override this method to return a new bounding rectangle for the cancel button cell. You might use this method to provide a custom layout for the search field control.

Availability

Available in Mac OS X v10.3 and later.

See Also

`- searchButtonRectForBounds:` (page 10)
`- searchTextRectForBounds:` (page 11)

Declared In

`NSSearchFieldCell.h`

maximumRecents

Returns the maximum number of recent search strings to display in the custom search menu.

```
- (NSInteger)maximumRecents
```

Return Value

The maximum number of search strings that can appear in the menu. This value is between `0` and `254`.

Availability

Available in Mac OS X v10.3 and later.

See Also

`- setMaximumRecents:` (page 13)

Declared In

`NSSearchFieldCell.h`

recentsAutosaveName

Returns the key under which the prior list of recent search strings has been archived.

```
- (NSString *)recentsAutosaveName
```

Return Value

The autosave name, which is used as a key in the standard user defaults to save the recent searches. The default value is `nil`, which causes searches not to be autosaved.

Availability

Available in Mac OS X v10.3 and later.

See Also

`- setRecentsAutosaveName:` (page 13)

Declared In

`NSSearchFieldCell.h`

recentSearches

Returns the list of recent search strings for the control.

```
- (NSArray *)recentSearches
```

Return Value

An array of `NSString` objects, each of which contains a search string either displayed in the search menu or from a recent autosave archive. If there have been no recent searches and no prior searches saved under an autosave name, this array may be empty.

Availability

Available in Mac OS X v10.3 and later.

See Also

- `setRecentsAutosaveName:` (page 13)
- `setRecentSearches:` (page 14)

Declared In

`NSSearchFieldCell.h`

resetCancelButtonCell

Resets the cancel button cell to its default attributes.

```
- (void)resetCancelButtonCell
```

Discussion

This method resets the target, action, regular image, and pressed image. By default, when users click the cancel button, the `delete:` action message is sent up the responder chain to the first `NSText` object that can handle it. This method gives you a way to customize the cancel button for specific situations and then reset the button defaults without having to undo changes individually.

Availability

Available in Mac OS X v10.3 and later.

See Also

- `setCancelButtonCell:` (page 12)
- `cancelButtonCell` (page 7)

Declared In

`NSSearchFieldCell.h`

resetSearchButtonCell

Resets the search button cell to its default attributes.

```
- (void)resetSearchButtonCell
```

Discussion

This method resets the target, action, regular image, and pressed image. By default, when users click the search button or press the Return key, the action defined for the receiver is sent to its designated target. This method gives you a way to customize the search button for specific situations and then reset the button defaults without having to undo changes individually.

Availability

Available in Mac OS X v10.3 and later.

See Also

`– setSearchButtonCell:` (page 14)

`– searchButtonCell` (page 10)

Declared In

`NSSearchFieldCell.h`

searchButtonCell

Returns the button cell used to display the search-button image.

```
- (NSButtonCell *)searchButtonCell
```

Return Value

The search button cell.

Availability

Available in Mac OS X v10.3 and later.

See Also

`– setSearchButtonCell:` (page 14)

`– resetSearchButtonCell` (page 9)

Declared In

`NSSearchFieldCell.h`

searchButtonRectForBounds:

Modifies the bounding rectangle for the search button cell.

```
- (NSRect)searchButtonRectForBounds:(NSRect)rect
```

Parameters

`rect`

The current bounding rectangle for the search button.

Return Value

The updated bounding rectangle to use for the search button. The default value is the value passed into the *`rect`* parameter.

Discussion

Subclasses can override this method to return a new bounding rectangle for the search button cell. You might use this method to provide a custom layout for the search field control.

Availability
Available in Mac OS X v10.3 and later.

See Also
- `cancelButtonRectForBounds:` (page 7)
- `searchTextRectForBounds:` (page 11)

Declared In
`NSSearchFieldCell.h`

searchMenuTemplate

Returns the menu template object used to dynamically construct the search pop-up icon menu.

```
- (NSMenu *)searchMenuTemplate
```

Return Value
The current menu template.

Availability
Available in Mac OS X v10.3 and later.

See Also
- `setSearchMenuTemplate:` (page 14)

Declared In
`NSSearchFieldCell.h`

searchTextRectForBounds:

Modifies the bounding rectangle for the search-text field cell.

```
- (NSRect)searchTextRectForBounds:(NSRect)rect
```

Parameters

rect

The current bounding rectangle for the search text field.

Return Value
The updated bounding rectangle to use for the search text field. The default value is the value passed into the *rect* parameter.

Discussion
Subclasses can override this method to return a new bounding rectangle for the text-field cell object. You might use this method to provide a custom layout for the search field control.

Availability
Available in Mac OS X v10.3 and later.

See Also
- `cancelButtonRectForBounds:` (page 7)
- `searchButtonRectForBounds:` (page 10)

Declared In

`NSSearchFieldCell.h`

sendsSearchStringImmediately

Returns a Boolean value indicating whether the receiver sends its action immediately upon being notified of changes to the search field text or after a brief pause.

```
- (BOOL)sendsSearchStringImmediately
```

Return Value

`YES` if the cell sends its action immediately upon notification of any changes to the search field; otherwise, `NO`.

Availability

Available in Mac OS X v10.4 and later.

See Also

`- setSendsSearchStringImmediately:` (page 15)

Declared In

`NSSearchFieldCell.h`

sendsWholeSearchString

Returns a Boolean value indicating whether the receiver sends the search action message when the user clicks the search button (or presses return) or after each keystroke.

```
- (BOOL)sendsWholeSearchString
```

Return Value

`YES` if the action message is sent all at once when the user clicks the search button or presses return; otherwise, `NO` if the search string is sent after each keystroke. The default value is `NO`.

Availability

Available in Mac OS X v10.3 and later.

See Also

`- setSendsWholeSearchString:` (page 15)

Declared In

`NSSearchFieldCell.h`

setCancelButtonCell:

Sets the button cell object used to display the cancel-button image

```
- (void)setCancelButtonCell:(NSButtonCell *)cell
```

Parameters

`cell`

The cancel button cell.

Availability

Available in Mac OS X v10.3 and later.

See Also

- `cancelButtonCell` (page 7)
- `resetCancelButtonCell` (page 9)

Declared In

`NSSearchFieldCell.h`

setMaximumRecents:

Sets the maximum number of search strings that can appear in the search menu

```
- (void)setMaximumRecents:(NSInteger)maxRecents
```

Parameters

maxRecents

The maximum number of search strings that can appear in the menu. This value can be between `0` and `254`. Specifying a value less than `0` sets the value to the default, which is `10`. Specifying a value greater than `254` sets the maximum to `254`.

Discussion

When the limit is exceeded, the oldest search string on the menu is dropped.

Availability

Available in Mac OS X v10.3 and later.

See Also

- `maximumRecents` (page 8)

Declared In

`NSSearchFieldCell.h`

setRecentsAutosaveName:

Sets the autosave name under which the receiver automatically archives the list of recent search strings.

```
- (void)setRecentsAutosaveName:(NSString *)name
```

Parameters

name

The autosave name, which is used as a key in the standard user defaults to save the recent searches. If you specify `nil` or an empty string for this parameter, no autosave name is set and searches are not autosaved.

Availability

Available in Mac OS X v10.3 and later.

See Also

- `recentsAutosaveName` (page 8)
- `setRecentSearches:` (page 14)

Declared In
`NSSearchFieldCell.h`

setRecentSearches:

Sets the list of recent search strings to list in the pop-up icon menu of the receiver.

```
- (void)setRecentSearches:(NSArray *)searches
```

Parameters

searches
An array of `NSString` objects containing the search strings.

Discussion
You might use this method to set the recent list of searches from an archived copy.

Availability
Available in Mac OS X v10.3 and later.

See Also
`- recentSearches` (page 9)

Declared In
`NSSearchFieldCell.h`

setSearchButtonCell:

Sets the button cell used to display the search-button image

```
- (void)setSearchButtonCell:(NSButtonCell *)cell
```

Parameters

cell
The search button cell.

Availability
Available in Mac OS X v10.3 and later.

See Also
`- searchButtonCell` (page 10)
`- resetSearchButtonCell` (page 9)

Declared In
`NSSearchFieldCell.h`

setSearchMenuTemplate:

Sets the menu template object used to dynamically construct the receiver's pop-up icon menu.

```
- (void)setSearchMenuTemplate:(NSMenu *)menu
```

Parameters

menu

The menu template to use.

Discussion

The receiver looks for the tag constants described in "`Menu tags`" (page 16) to determine how to populate the menu with items related to recent searches. See "Configuring a Search Menu" for a sample of how you might set up the search menu template.

Availability

Available in Mac OS X v10.3 and later.

See Also

`– searchMenuTemplate` (page 11)

Declared In

`NSSearchFieldCell.h`

setSendsSearchStringImmediately:

Sets whether the cell sends its action message to the target immediately upon notification of any changes to the search field text or after a brief pause.

```
- (void)setSendsSearchStringImmediately:(BOOL)flag
```

Parameters

flag

`YES` to send the cell's action immediately upon notification of any changes to the search field; otherwise, `NO` if you want the cell to pause briefly before sending its action message. Pausing gives the user the opportunity to type more text into the search field before initiating the search.

Availability

Available in Mac OS X v10.4 and later.

See Also

`– sendsSearchStringImmediately` (page 12)

Declared In

`NSSearchFieldCell.h`

setSendsWholeSearchString:

Sets whether the receiver sends the search action message when the user clicks the search button (or presses return) or after each keystroke.

```
- (void)setSendsWholeSearchString:(BOOL)flag
```

Parameters

flag

`YES` to send the action message all at once when the user clicks the search button or presses return; otherwise, `NO` to send the search string after each keystroke.

Availability

Available in Mac OS X v10.3 and later.

See Also
`- sendsWholeSearchString` (page 12)

Declared In
`NSSearchFieldCell.h`

Constants

Menu tags

The `NSSearchFieldCell` class uses these tag constants for identifying special menu items in the search-menu template set by `setSearchMenuTemplate:` (page 14). When an `NSSearchFieldCell` object dynamically constructs the actual search menu from this template, it shows or hides the tagged items as directed.

```
#define NSSearchFieldRecentsTitleMenuItemTag     1000
#define NSSearchFieldRecentsMenuItemTag          1001
#define NSSearchFieldClearRecentsMenuItemTag     1002
#define NSSearchFieldNoRecentsMenuItemTag        1003
```

Constants

`NSSearchFieldRecentsTitleMenuItemTag`

Identifies the menu item that is the title of the menu group for recent search strings.

This item is hidden if there are no recent strings.

You may use this tagged item for separator characters that also do not appear if there are no recent strings to display.

Available in Mac OS X v10.3 and later.

Declared in `NSSearchFieldCell.h`.

`NSSearchFieldRecentsMenuItemTag`

Identifies where recent search strings should appear in the “recents” menu group.

Available in Mac OS X v10.3 and later.

Declared in `NSSearchFieldCell.h`.

`NSSearchFieldClearRecentsMenuItemTag`

Identifies the menu item for clearing the current set of recent string searches in the menu.

This item is hidden if there are no recent strings.

Available in Mac OS X v10.3 and later.

Declared in `NSSearchFieldCell.h`.

`NSSearchFieldNoRecentsMenuItemTag`

Identifies the menu item that describes a lack of recent search strings (for example, “No recent searches”).

This item is hidden if there have been recent searches.

Available in Mac OS X v10.3 and later.

Declared in `NSSearchFieldCell.h`.

Availability
Available in Mac OS X v10.3 and later.

Declared In

`NSSearchFieldCell.h`

Document Revision History

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

Date	Notes
2008-03-11	Disambiguated the descriptions of the methods setSendsSearchStringImmediately: and setSendsWholeSearchString:.
2006-05-23	First publication of this content as a separate document.

Index