
NSOutlineView Class Reference

[Cocoa](#) > [User Experience](#)



2009-02-04



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, Cocoa, ColorSync, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

Shuffle is a trademark of Apple Inc.

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

NSOutlineView Class Reference 7

Overview	7
Tasks	8
Setting the Data Source	8
Working with Expandability	8
Expanding and Collapsing the Outline	8
Redisplaying Information	9
Converting Between Items and Rows	9
Working with the Outline Column	9
Working with Indentation	9
Working with Persistence	10
Supporting Drag and Drop	10
Getting the Parent for an Item	10
Getting the Frame for a Cell	10
Supporting Type Select	11
Working with Tooltips	11
Handling Selection	11
Displaying Cells	11
Moving and Resizing Columns	12
Editing Items	12
Working with Table Columns	12
Returning Row Height Information	12
Customizing Tracking Support	12
Grouping Rows	13
Instance Methods	13
autoresizesOutlineColumn	13
autosaveExpandedItems	13
collapseItem:	14
collapseItem:collapseChildren:	14
dataSource	15
expandItem:	15
expandItem:expandChildren:	16
frameOfOutlineCellAtRow:	16
indentationMarkerFollowsCell	17
indentationPerLevel	17
isExpandable:	17
isItemExpanded:	18
itemAtRow:	18
levelForItem:	19
levelForRow:	19
outlineTableColumn	20

parentForItem:	20
reloadItem:	20
reloadItem:reloadChildren:	21
rowForItem:	21
setAutosizesOutlineColumn:	22
setAutosaveExpandedItems:	22
setDataSource:	23
setDropItem:dropChildIndex:	23
setIndentationMarkerFollowsCell:	23
setIndentationPerLevel:	24
setOutlineTableColumn:	24
shouldCollapseAutoExpandedItemsForDeposited:	25
Delegate Methods	25
outlineView:dataCellForTableColumn:item:	25
outlineView:didClickTableColumn:	26
outlineView:didDragTableColumn:	26
outlineView:heightOfRowByItem:	26
outlineView:isGroupItem:	27
outlineView:mouseDownInHeaderOfTableColumn:	27
outlineView:nextTypeSelectMatchFromItem:toItem:forString:	28
outlineView:selectionIndexesForProposedSelection:	28
outlineView:shouldCollapseItem:	29
outlineView:shouldEditTableColumn:item:	29
outlineView:shouldExpandItem:	30
outlineView:shouldSelectItem:	30
outlineView:shouldSelectTableColumn:	31
outlineView:shouldShowCellExpansionForTableColumn:item:	31
outlineView:shouldTrackCell:forTableColumn:item:	31
outlineView:shouldTypeSelectForEvent:withCurrentSearchString:	32
outlineView:toolTipForCell:rect:tableColumn:item:mouseLocation:	33
outlineView:typeSelectStringForTableColumn:item:	33
outlineView:willDisplayCell:forTableColumn:item:	34
outlineView:willDisplayOutlineCell:forTableColumn:item:	34
outlineViewColumnDidMove:	35
outlineViewColumnDidResize:	35
outlineViewItemDidCollapse:	35
outlineViewItemDidExpand:	36
outlineViewItemWillCollapse:	36
outlineViewItemWillExpand:	36
outlineViewSelectionDidChange:	37
outlineViewSelectionIsChanging:	37
selectionShouldChangeInOutlineView:	37
Constants	38
Drop on Item Index	38
Notifications	38
NSOutlineViewColumnDidMoveNotification	38

NSOutlineViewColumnDidResizeNotification	39
NSOutlineViewItemDidCollapseNotification	39
NSOutlineViewItemDidExpandNotification	39
NSOutlineViewItemWillCollapseNotification	40
NSOutlineViewItemWillExpandNotification	40
NSOutlineViewSelectionDidChangeNotification	40
NSOutlineViewSelectionIsChangingNotification	41

Document Revision History 43

Index 45

NSOutlineView Class Reference

Inherits from	NSTableView : NSControl : NSView : NSResponder : NSObject
Conforms to	NSUserInterfaceValidations (NSTableView) NSAnimatablePropertyContainer (NSView) NSCoding (NSResponder) NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.0 and later.
Declared in	NSOutlineView.h
Companion guides	Outline View Programming Topics for Cocoa Drag and Drop Programming Topics for Cocoa
Related sample code	ColorSyncDevices-Cocoa EnhancedDataBurn QTKitMovieShuffler QTSSInspector SGDevices

Overview

`NSOutlineView` is a subclass of `NSTableView` that uses a row-and-column format to display hierarchical data that can be expanded and collapsed, such as directories and files in a file system. A user can expand and collapse rows, edit values, and resize and rearrange columns.

Important: Like a table view, an outline view does not store its own data, instead it retrieves data values as needed from a data source to which it has a weak reference (see [Communicating With Objects](#)). See the `NSOutlineViewDataSource` informal protocol, which declares the methods that an `NSOutlineView` object uses to access the contents of its data source object.

An outline view has the following features:

- A user can expand and collapse rows.
- Each item in the outline view must be unique. In order for the collapsed state to remain consistent between reloads the item's pointer must remain the same.
- The view gets data from a data source (see the `NSOutlineViewDataSource` informal protocol).
- The view retrieves only the data that needs to be displayed.

The following are commonly-used methods:

<code>numberOfRows</code>	Returns the number of rows in the receiver.
<code>collapseItem:</code> (page 14)	Collapses a given item.
<code>expandItem:</code> (page 15)	Expands a given item.
<code>reloadItem:reloadChildren:</code> (page 21)	Reloads a given item and, optionally, its children.

Tasks

Setting the Data Source

- `setDataSource:` (page 23)
Sets the receiver's data source to a given object.
- `dataSource` (page 15)
Returns the object that provides the data displayed by the receiver.

Working with Expandability

- `isExpandable:` (page 17)
Returns a Boolean value that indicates whether a given item is expandable.
- `isItemExpanded:` (page 18)
Returns a Boolean value that indicates whether a given item is expanded.

Expanding and Collapsing the Outline

- `outlineView:shouldExpandItem:` (page 30) *delegate method*
Returns a Boolean value that indicates whether the outline view should expand a given item.
- `outlineViewItemWillExpand:` (page 36) *delegate method*
Invoked when *notification* is posted—that is, whenever the user is about to expand an item in the outline view.
- `expandItem:` (page 15)
Expands a given item.
- `expandItem:expandChildren:` (page 16)
Expands a specified item and, optionally, its children.
- `outlineViewItemDidExpand:` (page 36) *delegate method*
Invoked when *notification* is posted—that is, whenever the user expands an item in the outline view.
- `outlineView:shouldCollapseItem:` (page 29) *delegate method*
Returns a Boolean value that indicates whether the outline view should collapse a given item.

- [outlineViewItemWillCollapse:](#) (page 36) *delegate method*
Invoked when *notification* is posted—that is, whenever the user is about to collapse an item in the outline view.
- [collapseItem:](#) (page 14)
Collapses a given item.
- [collapseItem:collapseChildren:](#) (page 14)
Collapses a given item and, optionally, its children.
- [outlineViewItemDidCollapse:](#) (page 35) *delegate method*
Invoked when *notification* is posted—that is, whenever the user collapses an item in the outline view.

Redisplaying Information

- [reloadItem:](#) (page 20)
Reloads and redisplays the data for *item*.
- [reloadItem:reloadChildren:](#) (page 21)
Reloads a given item and, optionally, its children.

Converting Between Items and Rows

- [itemAtRow:](#) (page 18)
Returns the item associated with a given row.
- [rowForItem:](#) (page 21)
Returns the row associated with a given item.

Working with the Outline Column

- [setOutlineTableColumn:](#) (page 24)
Sets the table column in which hierarchical data is displayed.
- [outlineTableColumn](#) (page 20)
Returns the table column in which hierarchical data is displayed.
- [autoresizesOutlineColumn](#) (page 13)
Returns a Boolean value that indicates whether the receiver automatically resizes its outline column when the user expands or collapses items.
- [setAutoresizesOutlineColumn:](#) (page 22)
Sets whether the receiver automatically resizes its outline column when the user expands or collapses an item.

Working with Indentation

- [levelForItem:](#) (page 19)
Returns the indentation level for a given item.

- [levelForRow:](#) (page 19)
Returns the indentation level for a given row.
- [setIndentationPerLevel:](#) (page 24)
Sets the per-level indentation.
- [indentationPerLevel](#) (page 17)
Returns the current indentation per level.
- [setIndentationMarkerFollowsCell:](#) (page 23)
Sets whether the indentation marker symbol displayed in the outline column should be indented along with the cell contents, or always displayed left-justified in the column.
- [indentationMarkerFollowsCell](#) (page 17)
Returns a Boolean value that indicates whether the indentation marker symbol displayed in the outline column should be indented along with the cell contents, or always displayed left-justified in the column.

Working with Persistence

- [autosaveExpandedItems](#) (page 13)
Returns a Boolean value that indicates whether the expanded items in the receiver are automatically saved.
- [setAutosaveExpandedItems:](#) (page 22)
Sets whether the expanded items in the receiver are automatically saved.

Supporting Drag and Drop

- [setDropItem:dropChildIndex:](#) (page 23)
Used to “retarget” a proposed drop.
- [shouldCollapseAutoExpandedItemsForDeposited:](#) (page 25)
Returns a Boolean value that indicates whether autoexpanded items should return to their original collapsed state.

Getting the Parent for an Item

- [parentForItem:](#) (page 20)
Returns the parent for a given item.

Getting the Frame for a Cell

- [frameOfOutlineCellAtRow:](#) (page 16)
Returns the frame of the outline cell for a given row.

Supporting Type Select

- `outlineView:typeSelectStringForTableColumn:item:` (page 33) *delegate method*
Returns the string that is used for type selection for a given column and item.
- `outlineView:nextTypeSelectMatchFromItem:toItem:forString:` (page 28) *delegate method*
Returns the first item that matches the searchString from within the range of startItem to endItem
- `outlineView:shouldTypeSelectForEvent:withCurrentSearchString:` (page 32) *delegate method*
Returns a Boolean value that indicates whether type select should proceed for a given event and search string.

Working with Tooltips

- `outlineView:toolTipForCell:rect:tableColumn:item:mouseLocation:` (page 33) *delegate method*
When the cursor pauses over a given cell, the value returned from this method is displayed in a tooltip.
- `outlineView:shouldShowCellExpansionForTableColumn:item:` (page 31) *delegate method*
Returns a Boolean value that indicates whether an expansion tooltip should appear in a given column for a given item.

Handling Selection

- `outlineView:shouldSelectTableColumn:` (page 31) *delegate method*
Returns a Boolean value that indicates whether the outline view should select a given table column.
- `outlineView:shouldSelectItem:` (page 30) *delegate method*
Returns a Boolean value that indicates whether the outline view should select a given item.
- `outlineView:selectionIndexesForProposedSelection:` (page 28) *delegate method*
Invoked to allow the delegate to modify the proposed selection.
- `selectionShouldChangeInOutlineView:` (page 37) *delegate method*
Returns a Boolean value that indicates whether the outline view should change its selection.
- `outlineViewSelectionIsChanging:` (page 37) *delegate method*
Invoked when *notification* is posted—that is, whenever the outline view's selection changes.
- `outlineViewSelectionDidChange:` (page 37) *delegate method*
Invoked when *notification* is posted—that is, immediately after the outline view's selection has changed.

Displaying Cells

- `outlineView:willDisplayCell:forTableColumn:item:` (page 34) *delegate method*
Informs the delegate that *outlineView* is about to display the cell specified by *tableColumn* and *item*.
- `outlineView:willDisplayOutlineCell:forTableColumn:item:` (page 34) *delegate method*
Informs the delegate that an outline view is about to display a cell used to draw the expansion symbol.

- `outlineView:dataCellForTableColumn:item:` (page 25) *delegate method*
Returns the cell to use in a given column for a given item.

Moving and Resizing Columns

- `outlineViewColumnDidMove:` (page 35) *delegate method*
Invoked when *notification* is posted—that is, whenever the user moves a column in the outline view.
- `outlineViewColumnDidResize:` (page 35) *delegate method*
Invoked when *notification* is posted—that is, whenever the user resizes a column in the outline view.

Editing Items

- `outlineView:shouldEditTableColumn:item:` (page 29) *delegate method*
Returns a Boolean value that indicates whether the outline view should allow editing of a given item in a given table column.

Working with Table Columns

- `outlineView:mouseDownInHeaderOfTableColumn:` (page 27) *delegate method*
Sent to the delegate whenever the mouse button is clicked in *outlineView* while the cursor is in a column header *tableColumn*.
- `outlineView:didClickTableColumn:` (page 26) *delegate method*
Sent at the time the mouse button subsequently goes up in *outlineView* and *tableColumn* has been “clicked” without having been dragged anywhere.
- `outlineView:didDragTableColumn:` (page 26) *delegate method*
Sent at the time the mouse button goes up in *outlineView* and *tableColumn* has been dragged during the time the mouse button was down.

Returning Row Height Information

- `outlineView:heightOfRowByItem:` (page 26) *delegate method*
Returns the height in points of the row containing *item*.

Customizing Tracking Support

- `outlineView:shouldTrackCell:forTableColumn:item:` (page 31) *delegate method*
Returns a Boolean value that indicates whether a given cell should be tracked.

Grouping Rows

- [outlineView:isGroupItem:](#) (page 27) *delegate method*
Returns a Boolean that indicates whether a given row should be drawn in the “group row” style.

Instance Methods

autoresizesOutlineColumn

Returns a Boolean value that indicates whether the receiver automatically resizes its outline column when the user expands or collapses items.

- (BOOL)autoresizesOutlineColumn

Return Value

YES if the outline column is automatically resized, otherwise NO.

Discussion

The outline column contains the cells with the expansion symbols and is generally the first column. The default is YES.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setAutoresizesOutlineColumn:](#) (page 22)

Declared In

NSOutlineView.h

autosaveExpandedItems

Returns a Boolean value that indicates whether the expanded items in the receiver are automatically saved.

- (BOOL)autosaveExpandedItems

Return Value

YES if when an item is expanded, the outline view displays the previous expanded state of its contained items, otherwise NO.

Discussion

The outline view information is saved separately for each user and for each application that user uses. Note that if `autosaveName` returns `nil`, this setting is ignored, and outline information isn't saved.

Special Considerations

Starting in Mac OS X version 10.5, the value for `autosaveExpandedItems` is saved out in the nib file. The default value is NO.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [autosaveName](#) (NSTableView)
- [autosaveTableColumns](#) (NSTableView)
- [setAutosaveExpandedItems:](#) (page 22)

Declared In

NSOutlineView.h

collapseItem:

Collapses a given item.

- (void)collapseItem:(id) *item***Parameters***item*

An item in the receiver.

DiscussionIf *item* is not expanded or not expandable, does nothing

If collapsing takes place, posts item collapse notification.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [expandItem:](#) (page 15)

Declared In

NSOutlineView.h

collapseItem:collapseChildren:

Collapses a given item and, optionally, its children.

- (void)collapseItem:(id) *item* collapseChildren:(BOOL) *collapseChildren***Parameters***item*

An item in the receiver.

Starting in Mac OS X version 10.5, passing 'nil' will collapse each item under the root in the outline view.

*collapseChildren*If YES, recursively collapses *item* and its children. If NO, collapses *item* only (identical to [collapseItem:](#) (page 14)).**Discussion**For example, this method is invoked with the *collapseChildren* parameter set to YES when a user Option-clicks the disclosure triangle for an item in the outline view (to collapse the item and all its contained items).

For each item collapsed, posts an item collapsed notification.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [collapseItem:](#) (page 14)
- [expandItem:expandChildren:](#) (page 16)

Declared In

NSOutlineView.h

dataSource

Returns the object that provides the data displayed by the receiver.

- (id)dataSource

Return Value

The object that provides the data displayed by the receiver.

Discussion

See Writing an Outline View Data Source and the NSOutlineViewDataSource Protocol Reference informal protocol specification for more information.

See Also

- [setDataSource:](#) (page 23)

expandItem:

Expands a given item.

- (void)expandItem:(id) *item*

Parameters

item

An item in the receiver.

Discussion

If *item* is not expandable or is already expanded, does nothing.

If expanding takes place, posts an item expanded notification.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [collapseItem:](#) (page 14)

Declared In

NSOutlineView.h

expandItem:expandChildren:

Expands a specified item and, optionally, its children.

```
- (void)expandItem:(id)item expandChildren:(BOOL)expandChildren
```

Parameters

item

An item in the receiver.

Starting in Mac OS X version 10.5, passing 'nil' will expand each item under the root in the outline view.

expandChildren

If YES, recursively expands *item* and its children. If NO, expands *item* only (identical to [expandItem:](#) (page 15)).

Discussion

For example, this method is invoked with the *expandChildren* parameter set to YES when a user Option-clicks the disclosure triangle for an item in the outline view (to expand the item and all its contained items).

For each item expanded, posts an item expanded notification.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [collapseItem:collapseChildren:](#) (page 14)
- [expandItem:](#) (page 15)

Declared In

NSOutlineView.h

frameOfOutlineCellAtRow:

Returns the frame of the outline cell for a given row.

```
- (NSRect)frameOfOutlineCellAtRow:(NSInteger)row
```

Parameters

row

The index of the row for which to return the frame.

Return Value

The frame of the outline cell for the row at index *row*, considering the current indentation and the value returned by [indentationMarkerFollowsCell](#) (page 17). If the row at index *row* is not an expandable row, returns NSZeroRect.

Discussion

You can override this method in a subclass to return a custom frame for the outline button cell. If your override returns an empty rect, no outline cell is drawn for that row. You might do that, for example, so that the disclosure triangle will not be shown for a row that should never be expanded.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

indentationMarkerFollowsCell

Returns a Boolean value that indicates whether the indentation marker symbol displayed in the outline column should be indented along with the cell contents, or always displayed left-justified in the column.

- (BOOL)indentationMarkerFollowsCell

Return Value

YES if the indentation marker is indented along with the cell contents, otherwise NO.

Discussion

The default is YES.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setIndentationMarkerFollowsCell:](#) (page 23)

Declared In

NSOutlineView.h

indentationPerLevel

Returns the current indentation per level.

- (CGFloat)indentationPerLevel

Return Value

The current indentation per level, in points.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setIndentationPerLevel:](#) (page 24)

Declared In

NSOutlineView.h

isExpandable:

Returns a Boolean value that indicates whether a given item is expandable.

- (BOOL)isExpandable:(id) *item*

Parameters

item

An item in the receiver.

Return Value

YES if *item* is expandable—that is, *item* can contain other items, otherwise NO.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [expandItem:](#) (page 15)
- [isItemExpanded:](#) (page 18)

Declared In

NSOutlineView.h

isItemExpanded:

Returns a Boolean value that indicates whether a given item is expanded.

```
- (BOOL)isItemExpanded:(id)item
```

Parameters

item

An item in the receiver.

Return Value

YES if *item* is expanded, otherwise NO.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [expandItem:](#) (page 15)
- [isExpandable:](#) (page 17)

Declared In

NSOutlineView.h

itemAtRow:

Returns the item associated with a given row.

```
- (id)itemAtRow:(NSInteger)row
```

Parameters

row

The index of a row in the receiver.

Return Value

The item associated with *row*.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [rowForItem:](#) (page 21)

Declared In

NSOutlineView.h

levelForItem:

Returns the indentation level for a given item.

- (NSInteger)levelForItem:(id) *item***Parameters***item*

An item in the receiver.

Return ValueThe indentation level for *item*. If *item* is `nil` (which is the root item), returns `-1`.**Discussion**

The levels are zero-based—that is, the first level of displayed items is level 0.

Availability

Available in Mac OS X v10.0 and later.

See Also- [indentationPerLevel](#) (page 17)- [levelForRow:](#) (page 19)**Declared In**

NSOutlineView.h

levelForRow:

Returns the indentation level for a given row.

- (NSInteger)levelForRow:(NSInteger) *row***Parameters***row*

The index of a row in the receiver.

Return ValueThe indentation level for *row*. For an invalid row, returns `-1`.**Discussion**

The levels are zero-based—that is, the first level of displayed items is level 0.

Availability

Available in Mac OS X v10.0 and later.

See Also- [indentationPerLevel](#) (page 17)- [levelForItem:](#) (page 19)**Declared In**

NSOutlineView.h

outlineTableColumn

Returns the table column in which hierarchical data is displayed.

```
- (NSTableColumn *)outlineTableColumn
```

Return Value

The table column in which hierarchical data is displayed.

Discussion

Each level of hierarchical data is indented by the amount specified by [indentationPerLevel](#) (page 17) (the default is 16.0), and decorated with the indentation marker (disclosure triangle) on rows that are expandable.

Special Considerations

Starting in Mac OS X version 10.5, outline table column data is saved in `encodeWithCoder:` and restored in `initWithCoder:`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setOutlineTableColumn:](#) (page 24)

Declared In

NSOutlineView.h

parentForItem:

Returns the parent for a given item.

```
- (id)parentForItem:(id)item
```

Parameters

item

The item for which to return the parent.

Return Value

The parent for *item*, or `nil` if the parent is the root.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

reloadItem:

Reloads and redisplay the data for *item*.

```
- (void)reloadItem:(id)item
```

Availability

Available in Mac OS X v10.0 and later.

See Also

- [reloadItem:reloadChildren:](#) (page 21)

Declared In

NSOutlineView.h

reloadItem:reloadChildren:

Reloads a given item and, optionally, its children.

```
- (void)reloadItem:(id)item reloadChildren:(BOOL)reloadChildren
```

Parameters

item

An item in the receiver.

Starting in Mac OS X version 10.5, passing 'nil' will reload everything under the root in the outline view.

reloadChildren

If YES, recursively reloads *item* and its children. If NO, reloads *item* only (identical to [reloadItem:](#) (page 20)).

It is not necessary, or efficient, to reload children if the item is not expanded.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [reloadItem:](#) (page 20)

Declared In

NSOutlineView.h

rowForItem:

Returns the row associated with a given item.

```
- (NSInteger)rowForItem:(id)item
```

Parameters

item

An item in the receiver.

Return Value

The row associated with *item*, or -1 if *item* is nil or cannot be found.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [itemAtRow:](#) (page 18)

Declared In

NSOutlineView.h

setAutoresizesOutlineColumn:

Sets whether the receiver automatically resizes its outline column when the user expands or collapses an item.

```
- (void)setAutoresizesOutlineColumn:(BOOL)resize
```

Parameters

resize

YES if the outline column is automatically resized, otherwise NO.

Discussion

The outline column contains the cells with the expansion symbols and is generally the first column. The default is YES.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [autoresizesOutlineColumn](#) (page 13)

Declared In

NSOutlineView.h

setAutosaveExpandedItems:

Sets whether the expanded items in the receiver are automatically saved.

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

Discussion

If *flag* is different from the current value, this method also reads in the saved information and sets the outline view's options to match. YES indicates that when an item is expanded, the outline view displays the previous expanded state of its contained items.

The outline information is saved separately for each user and for each application that user uses.

If `autosaveName` returns `nil` or if you haven't implemented the data source methods `outlineView:itemForPersistentObject:` and `outlineView:persistentObjectForItem:`, this setting is ignored, and expanded item information isn't saved.

Note that you can have separate settings for [autosaveExpandedItems](#) (page 13) and `autosaveTableColumns`, so you could, for example, save expanded item information, but not table column positions.

Special Considerations

Starting in Mac OS X version 10.5, the value for `autosaveExpandedItems` is saved out in the nib file. The default value is NO.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [autosaveExpandedItems](#) (page 13)
- `setAutosaveTableColumns:` (NSTableView)

Declared In

NSOutlineView.h

setDataSource:

Sets the receiver's data source to a given object.

- (void)setDataSource:(id)*anObject***Parameters***anObject*

The data source for the receiver. The object must implement the appropriate methods of the NSOutlineViewDataSource Protocol Reference informal protocol.

Discussion

The receiver maintains a weak reference to the data source (see Communicating With Objects). After setting the data source, this method invokes `tile`.

This method raises an `NSInternalInconsistencyException` if *anObject* doesn't respond to all of `outlineView:child:ofItem:`, `outlineView:isItemExpandable:`, `outlineView:numberOfChildrenOfItem:`, and `outlineView:objectValueForTableColumn:byItem:`.

See Also- [dataSource](#) (page 15)**setDropItem:dropChildIndex:**

Used to “retarget” a proposed drop.

- (void)setDropItem:(id)*item* dropChildIndex:(NSInteger)*index***Discussion**

For example, to specify a drop on an item *l*, you specify *item* as *l* and *index* as `NSOutlineViewDropOnItemIndex`. To specify a drop between child 2 and 3 of item *l*, you specify *item* as *l* and *index* as 3 (children are a zero-based index). To specify a drop on an unexpandable item 1, you specify *item* as 1 and *index* as `NSOutlineViewDropOnItemIndex`.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

setIndentationMarkerFollowsCell:

Sets whether the indentation marker symbol displayed in the outline column should be indented along with the cell contents, or always displayed left-justified in the column.

- (void)setIndentationMarkerFollowsCell:(BOOL)*drawInCell***Discussion**

The default is YES, the indentation marker is indented along with the cell contents.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [indentationMarkerFollowsCell](#) (page 17)

Declared In

NSOutlineView.h

setIndentationPerLevel:

Sets the per-level indentation.

```
- (void)setIndentationPerLevel:(CGFloat)newIndentLevel
```

Parameters

newIndentLevel

The indentation per level, in points.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [indentationPerLevel](#) (page 17)

Declared In

NSOutlineView.h

setOutlineTableColumn:

Sets the table column in which hierarchical data is displayed.

```
- (void)setOutlineTableColumn:(NSTableColumn *)outlineTableColumn
```

Parameters

outlineTableColumn

The table column in which hierarchical data is displayed.

Special Considerations

Starting in Mac OS X version 10.5, outline table column data is saved in `encodeWithCoder:` and restored in `initWithCoder:`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [outlineTableColumn](#) (page 20)

Declared In

NSOutlineView.h

shouldCollapseAutoExpandedItemsForDeposited:

Returns a Boolean value that indicates whether autoexpanded items should return to their original collapsed state.

```
- (BOOL)shouldCollapseAutoExpandedItemsForDeposited:(BOOL)deposited
```

Return Value

YES if autoexpanded items should return to their original collapsed state, otherwise NO.

Discussion

Override this method to provide custom behavior. *deposited* tells whether or not the drop terminated due to a successful drop.

This method is called in a variety of situations. For example, it is sent shortly after `outlineView:acceptDrop:item:childIndex:` is processed and also if the drag exits the outline view (exiting the view is treated the same as a failed drop).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

Delegate Methods

outlineView:dataCellForTableColumn:item:

Returns the cell to use in a given column for a given item.

```
- (NSCell *)outlineView:(NSOutlineView *)outlineView
  dataCellForTableColumn:(NSTableColumn *)tableColumn
  item:(id)item
```

Parameters

outlineView

The outline view that sent the message.

tableColumn

The table column for which the cell is required. This value may be `nil`.

item

The item for which the cell is required.

Return Value

The cell to use in column *tableColumn* for item *item*, or `nil` (see Discussion). The cell must properly implement `copyWithZone:` (since it may be copied by the outline view).

Discussion

You can return a different data cell for any particular combination of table column and item, or a cell that will be used for the entire row (a full width cell). If *tableColumn* is non-`nil`, you should return a cell. Typically you should default to returning the result from `[tableColumn dataCellForRow:row]`.

When each row (identified by the item) is being drawn, this method is first called with a `nil` value for `tableColumn`. At this time, you can return a cell that will be used to draw the entire row, acting like a group. If you do return a cell for the 'nil' table column, your implementations of the other corresponding datasource and delegate methods must be prepared to be invoked with a `nil` value for `tableColumn`. If do not return a cell for the 'nil' table column, the method will be called once for each column in the outline view, as usual.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

outlineView:didClickTableColumn:

Sent at the time the mouse button subsequently goes up in `outlineView` and `tableColumn` has been “clicked” without having been dragged anywhere.

```
- (void)outlineView:(NSOutlineView *)outlineView didClickTableColumn:(NSTableColumn *)tableColumn
```

Availability

Available in Mac OS X v10.3 and later.

Declared In

NSOutlineView.h

outlineView:didDragTableColumn:

Sent at the time the mouse button goes up in `outlineView` and `tableColumn` has been dragged during the time the mouse button was down.

```
- (void)outlineView:(NSOutlineView *)outlineView didDragTableColumn:(NSTableColumn *)tableColumn
```

Availability

Available in Mac OS X v10.3 and later.

Declared In

NSOutlineView.h

outlineView:heightOfRowByItem:

Returns the height in points of the row containing `item`.

```
- (CGFloat)outlineView:(NSOutlineView *)outlineView heightOfRowByItem:(id)item
```

Discussion

Values returned by this method should not include intercell spacing and must be greater than 0. Implement this method to support an outline view with varying row heights.

Special Considerations

For large tables in particular, you should make sure that this method is efficient. `NSTableView` may cache the values this method returns, so if you would like to change a row's height make sure to invalidate the row height by calling `noteHeightOfRowsWithIndexesChanged:`. `NSTableView` automatically invalidates its entire row height cache in `reloadData` and `noteNumberOfRowsChanged`.

Availability

Available in Mac OS X v10.4 and later.

Declared In

`NSOutlineView.h`

outlineView:isGroupItem:

Returns a Boolean that indicates whether a given row should be drawn in the “group row” style.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView
    isGroupItem:(id)item
```

Parameters

outlineView

The outline view that sent the message.

item

An item in the outline view.

Return Value

YES to indicate a particular row should have the “group row” style drawn for that row, otherwise NO.

Discussion

If the cell in that row is an instance of `NSTextFieldCell` and contains only a string value, the “group row” style attributes are automatically applied for that cell.

Availability

Available in Mac OS X v10.5 and later.

Declared In

`NSOutlineView.h`

outlineView:mouseDownInHeaderOfTableColumn:

Sent to the delegate whenever the mouse button is clicked in *outlineView* while the cursor is in a column header *tableColumn*.

```
- (void)outlineView:(NSOutlineView *)outlineView
    mouseDownInHeaderOfTableColumn:(NSTableColumn *)tableColumn
```

Availability

Available in Mac OS X v10.3 and later.

Declared In

`NSOutlineView.h`

outlineView:nextTypeSelectMatchFromItem:toItem:forString:

Returns the first item that matches the *searchString* from within the range of *startItem* to *endItem*

```
- (id)outlineView:(NSOutlineView *)outlineView
  nextTypeSelectMatchFromItem:(id)startItem
  toItem:(id)endItem
  forString:(NSString *)searchString
```

Parameters

outlineView

The outline view that sent the message.

startItem

The first item to search.

endItem

The item before which to stop searching. It is possible for *endItem* to be less than *startItem* if the search will wrap.

searchString

The string for which to search.

Return Value

The first item—from within the range of *startItem* to *endItem*—that matches the *searchString*, or *nil* if there is no match.

Discussion

Implement this method if you want to control how type selection works. You should include *startItem* as a possible match, but do not include *endItem*.

It is not necessary to implement this method in order to support type select.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

outlineView:selectionIndexesForProposedSelection:

Invoked to allow the delegate to modify the proposed selection.

```
- (NSIndexSet *)outlineView:(NSOutlineView *)outlineView
  selectionIndexesForProposedSelection:(NSIndexSet *)proposedSelectionIndexes
```

Parameters

outlineView

The outline view that sent the message.

proposedSelectionIndexes

An index set containing the indexes of the proposed selection.

Return Value

An *NSIndexSet* instance containing the indexes of the new selection. Return *proposedSelectionIndexes* if the proposed selection is acceptable, or the value of the table view's existing selection to avoid changing the selection.

Discussion

This method may be called multiple times with one new index added to the existing selection to find out if a particular index can be selected when the user is extending the selection with the keyboard or mouse.

Implementation of this method is optional. If implemented, this method will be called instead of `outlineView:shouldSelectItem:` (page 30).

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

outlineView:shouldCollapseItem:

Returns a Boolean value that indicates whether the outline view should collapse a given item.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView shouldCollapseItem:(id)item
```

Parameters

outlineView

The outline view that sent the message.

item

The item that should collapse.

Return Value

YES to permit *outlineView* to collapse *item*, NO to deny permission.

Discussion

The delegate can implement this method to disallow collapsing of specific items. For example, if the first row of your outline view should not be collapsed, your delegate method could contain this line of code:

```
return [outlineView rowForItem:item]!=0;
```

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineView:shouldEditTableColumn:item:

Returns a Boolean value that indicates whether the outline view should allow editing of a given item in a given table column.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView shouldEditTableColumn:(NSTableColumn *)tableColumn item:(id)item
```

Return Value

YES to permit *outlineView* to edit the cell specified by *tableColumn* and *item*, NO to deny permission.

If this method returns YES, the cell may still not be editable—for example, if you have set up a custom `NSTextFieldCell` as a data cell, it must return YES for `isEditable` to allow editing.

Discussion

The delegate can implement this method to disallow editing of specific cells.

Availability

Available in Mac OS X v10.0 and later.

See Also

- `outlineView:setObjectValue:forTableColumn:byItem:`

Declared In

NSOutlineView.h

outlineView:shouldExpandItem:

Returns a Boolean value that indicates whether the outline view should expand a given item.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView shouldExpandItem:(id)item
```

Parameters

outlineView

The outline view that sent the message.

item

The item that should expand.

Return Value

YES to permit *outlineView* to expand *item*, NO to deny permission.

Discussion

The delegate can implement this method to disallow expanding of specific items.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineView:shouldSelectItem:

Returns a Boolean value that indicates whether the outline view should select a given item.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView shouldSelectItem:(id)item
```

Return Value

YES to permit *outlineView* to select *item*, NO to deny permission.

Discussion

The delegate can implement this method to disallow selection of particular items.

For better performance and finer grain control over the selection, use [outlineView:selectionIndexesForProposedSelection:](#) (page 28).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineView:shouldSelectTableColumn:

Returns a Boolean value that indicates whether the outline view should select a given table column.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView
  shouldSelectTableColumn:(NSTableColumn *)tableColumn
```

Return Value

YES to permit *outlineView* to select *tableColumn*, NO to deny permission.

Discussion

The delegate can implement this method to disallow selection of specific columns.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineView:shouldShowCellExpansionForTableColumn:item:

Returns a Boolean value that indicates whether an expansion tooltip should appear in a given column for a given item.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView
  shouldShowCellExpansionForTableColumn:(NSTableColumn *)tableColumn
  item:(id)item
```

Parameters

outlineView

The outline view that sent the message.

tableColumn

A table column in the outline view.

item

An item in the outline view.

Return Value

YES to allow an expansion tooltip to appear in the column *tableColumn* for item *item*, otherwise NO.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

outlineView:shouldTrackCell:forTableColumn:item:

Returns a Boolean value that indicates whether a given cell should be tracked.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView
  shouldTrackCell:(NSCell *)cell
  forTableColumn:(NSTableColumn *)tableColumn
  item:(id)item
```

Parameters*outlineView*

The outline view that sent the message.

*cell*The cell used to display item *item* in column *tableColumn**tableColumn*

A table column in the outline view.

item

An item in the outline view.

Return ValueYES if the cell should be tracked for the item *item* in column *tableColumn*, otherwise NO.**Discussion**

Normally, only selectable or selected cells can be tracked. If you implement this method, cells which are not selectable or selected can be tracked (and vice-versa). For example, this allows you to have a button cell in a table which does not change the selection, but can still be clicked on and tracked.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

outlineView:shouldTypeSelectForEvent:withCurrentSearchString:

Returns a Boolean value that indicates whether type select should proceed for a given event and search string.

```
- (BOOL)outlineView:(NSOutlineView *)outlineView
  shouldTypeSelectForEvent:(NSEvent *)event
  withCurrentSearchString:(NSString *)searchString
```

Parameters*outlineView*

The outline view that sent the message.

event

The event that caused the message to be sent.

*searchString*The string for which searching is to proceed. The search string is *nil* if no type select has begun.**Return Value**

YES if type select should proceed, otherwise NO.

Discussion

Generally, this method will be called from `keyDown:` and the event will be a key event.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

outlineView:toolTipForCell:rect:tableColumn:item:mouseLocation:

When the cursor pauses over a given cell, the value returned from this method is displayed in a tooltip.

```
- (NSString *)outlineView:(NSOutlineView *)ov toolTipForCell:(NSCell *)cell
    rect:(NSRectPointer)rect tableColumn:(NSTableColumn *)tc item:(id)item
    mouseLocation:(NSPoint)mouseLocation
```

Parameters

ov

The outline view that sent the message.

cell

The cell for which to generate a tooltip.

rect

The proposed active area of the tooltip. To control the default active area, you can modify the *rect* parameter. By default, *rect* is computed as `[cell drawingRectForBounds:cellFrame]`.

tc

The table column that contains *cell*.

item

The item for which to display a tooltip.

mouseLocation

The current mouse location in view coordinates.

Return Value

If you don't want a tooltip at that location, return `nil` or the empty string.

Availability

Available in Mac OS X v10.4 and later.

Declared In

NSOutlineView.h

outlineView:typeSelectStringForTableColumn:item:

Returns the string that is used for type selection for a given column and item.

```
- (NSString *)outlineView:(NSOutlineView *)outlineView
    typeSelectStringForTableColumn:(NSTableColumn *)tableColumn
    item:(id)item
```

Parameters

outlineView

The outline view that sent the message.

tableColumn

A table column in the outline view.

item

An item in the outline view.

Return Value

The string that is used for type selection. You may want to change what is searched for based on what is displayed, or simply return `nil` for that row and/or column to not be searched

Discussion

Implement this method if you want to control the string that is used for type selection. You may want to change what is searched for based on what is displayed, or simply return `nil` to specify that the given row and/or column should not be searched. By default, all cells with text in them are searched.

The default value when this delegate method is not implemented is:

```
[[outlineView preparedCellAtColumn:tableColumn row:[outlineView rowForItem:item]]
 stringValue]
```

and you can return this value from the delegate method if you wish.

Availability

Available in Mac OS X v10.5 and later.

Declared In

NSOutlineView.h

outlineView:willDisplayCell:forTableColumn:item:

Informs the delegate that *outlineView* is about to display the cell specified by *tableColumn* and *item*.

```
- (void)outlineView:(NSOutlineView *)outlineView willDisplayCell:(id)cell
  forTableColumn:(NSTableColumn *)tableColumn item:(id)item
```

Discussion

The delegate can modify *cell* to alter its display attributes; for example, making uneditable values display in italic or gray text.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineView:willDisplayOutlineCell:forTableColumn:item:

Informs the delegate that an outline view is about to display a cell used to draw the expansion symbol.

```
- (void)outlineView:(NSOutlineView *)outlineView willDisplayOutlineCell:(id)cell
  forTableColumn:(NSTableColumn *)tableColumn item:(id)item
```

Discussion

Informs the delegate that *outlineView* is about to display *cell*—an expandable cell (a cell that has the expansion symbol)—for the column and item specified by *tableColumn* and *item*. The delegate can modify cell to alter its display attributes.

This method is not invoked when *outlineView* is about to display a non-expandable cell.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewColumnDidMove:

Invoked when *notification* is posted—that is, whenever the user moves a column in the outline view.

```
- (void)outlineViewColumnDidMove:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewColumnDidMoveNotification](#) (page 38).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewColumnDidResize:

Invoked when *notification* is posted—that is, whenever the user resizes a column in the outline view.

```
- (void)outlineViewColumnDidResize:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewColumnDidResizeNotification](#) (page 39).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewItemDidCollapse:

Invoked when *notification* is posted—that is, whenever the user collapses an item in the outline view.

```
- (void)outlineViewItemDidCollapse:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewItemDidCollapseNotification](#) (page 39).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewItemDidExpand:

Invoked when *notification* is posted—that is, whenever the user expands an item in the outline view.

```
- (void)outlineViewItemDidExpand:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewItemDidExpandNotification](#) (page 39).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewItemWillCollapse:

Invoked when *notification* is posted—that is, whenever the user is about to collapse an item in the outline view.

```
- (void)outlineViewItemWillCollapse:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewItemWillCollapseNotification](#) (page 40).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewItemWillExpand:

Invoked when *notification* is posted—that is, whenever the user is about to expand an item in the outline view.

```
- (void)outlineViewItemWillExpand:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewItemWillExpandNotification](#) (page 40).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewSelectionDidChange:

Invoked when *notification* is posted—that is, immediately after the outline view’s selection has changed.

```
- (void)outlineViewSelectionDidChange:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewSelectionDidChangeNotification](#) (page 40).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

outlineViewSelectionIsChanging:

Invoked when *notification* is posted—that is, whenever the outline view’s selection changes.

```
- (void)outlineViewSelectionIsChanging:(NSNotification *)notification
```

Discussion

This method is invoked as a result of posting an [NSOutlineViewSelectionIsChangingNotification](#) (page 41).

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

selectionShouldChangeInOutlineView:

Returns a Boolean value that indicates whether the outline view should change its selection.

```
- (BOOL)selectionShouldChangeInOutlineView:(NSOutlineView *)outlineView
```

Return Value

YES to permit *outlineView* to change its selection (typically a row being edited), NO to deny permission.

Discussion

For example, if the user is editing a cell and enters an improper value, the delegate can prevent the user from selecting or editing any other cells until a proper value has been entered into the original cell. The delegate can implement this method for complex validation of edited rows based on the values of any of their cells.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

Constants

Drop on Item Index

This constant defines an index that allows you to drop an item directly on a target.

```
enum {
    NSOutlineViewDropOnItemIndex = -1
};
```

Constants

NSOutlineViewDropOnItemIndex

May be used as a valid child index of a drop target item.

In this case, the drop will happen directly on the target item.

Available in Mac OS X v10.0 and later.

Declared in NSOutlineView.h.

Declared In

NSOutlineView.h

Notifications

NSOutlineViewColumnDidMoveNotification

Posted whenever a column is moved by user action in an NSOutlineView object.

The notification object is the NSOutlineView object in which a column moved. The *userInfo* dictionary contains the following information:

Key	Value
@"NSOldColumn"	An NSNumber object containing the integer value of the column's original index
@"NSNewColumn"	An NSNumber object containing the integer value of the column's present index

Availability

Available in Mac OS X v10.0 and later.

See Also

- `moveColumn:toColumn:(NSTableView)`

Declared In

NSOutlineView.h

NSOutlineViewColumnDidResizeNotification

Posted whenever a column is resized in an `NSOutlineView` object.

The notification object is the `NSOutlineView` object in which a column was resized. The *userInfo* dictionary contains the following information:

Key	Value
@NSTableColumn"	The column that was resized.
@NSOldWidth"	An <code>NSNumber</code> object containing the column's original width

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

NSOutlineViewItemDidCollapseNotification

Posted whenever an item is collapsed in an `NSOutlineView` object.

The notification object is the `NSOutlineView` object in which an item was collapsed. A collapsed item's children lose their status as being selected. The *userInfo* dictionary contains the following information:

Key	Value
@NSObject"	The item that was collapsed (an id)

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

NSOutlineViewItemDidExpandNotification

Posted whenever an item is expanded in an `NSOutlineView` object.

The notification object is the `NSOutlineView` object in which an item was expanded. The *userInfo* dictionary contains the following information:

Key	Value
@"NSObject"	The item that was expanded (an id)

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

NSOutlineViewItemWillCollapseNotification

Posted before an item is collapsed (after the user clicks the arrow but before the item is collapsed).

The notification object is the `NSOutlineView` object that contains the item about to be collapsed. A collapsed item's children will lose their status as being selected. The `userInfo` dictionary contains the following information:

Key	Value
@"NSObject"	The item about to be collapsed (an id)

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

NSOutlineViewItemWillExpandNotification

Posted before an item is expanded (after the user clicks the arrow but before the item is collapsed).

The notification object is the outline view that contains an item about to be expanded. The `userInfo` dictionary contains the following information:

Key	Value
@"NSObject"	The item that is to be expanded (an id)

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

NSOutlineViewSelectionDidChangeNotification

Posted after the outline view's selection changes.

The notification object is the outline view whose selection changed. This notification does not contain a `userInfo` dictionary.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

NSOutlineViewSelectionIsChangingNotification

Posted as the outline view's selection changes (while the mouse button is still down).

The notification object is the outline view whose selection is changing. This notification does not contain a *userInfo* dictionary.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSOutlineView.h

Document Revision History

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

Date	Notes
2009-02-04	Added documentation for method <code>outlineView:selectionIndexesForProposedSelection:</code> .
2007-07-26	Clarified definitions for several methods.
	Noted changes for Mac OS version 10.5 in the following methods: autosaveExpandedItems (page 13), setAutosaveExpandedItems: (page 22), collapseItem:collapseChildren: (page 14), expandItem:expandChildren: (page 16), reloadItem:reloadChildren: (page 21), outlineTableColumn (page 20), setOutlineTableColumn: (page 24), frameOfOutlineCellAtRow: (page 16), outlineView:shouldCollapseItem: (page 29) and outlineView:willDisplayOutlineCell:forTableColumn:item: (page 34).
2006-11-07	Added note about equality of items in an <code>NSOutlineView</code> object.
2006-06-28	Added a definition of data source methods.
2006-05-23	First publication of this content as a separate document.
	Added information regarding how items are expanded and collapsed to the following method descriptions: autosaveExpandedItems (page 13), setAutosaveExpandedItems: (page 22), collapseItem:collapseChildren: (page 14), and expandItem:expandChildren: (page 16).

REVISION HISTORY

Document Revision History

Index

A

autoresizesOutlineColumn **instance method** [13](#)
autosaveExpandedItems **instance method** [13](#)

C

collapseItem: **instance method** [14](#)
collapseItem:collapseChildren: **instance method** [14](#)

D

dataSource **instance method** [15](#)
Drop on Item Index [38](#)

E

expandItem: **instance method** [15](#)
expandItem:expandChildren: **instance method** [16](#)

F

frameOfOutlineCellAtRow: **instance method** [16](#)

I

indentationMarkerFollowsCell **instance method** [17](#)
indentationPerLevel **instance method** [17](#)
isExpandable: **instance method** [17](#)
isItemExpanded: **instance method** [18](#)
itemAtRow: **instance method** [18](#)

L

levelForItem: **instance method** [19](#)
levelForRow: **instance method** [19](#)

N

NSOutlineViewColumnDidMoveNotification **notification** [38](#)
NSOutlineViewColumnDidResizeNotification **notification** [39](#)
NSOutlineViewDropOnItemIndex **constant** [38](#)
NSOutlineViewItemDidCollapseNotification **notification** [39](#)
NSOutlineViewItemDidExpandNotification **notification** [39](#)
NSOutlineViewItemWillCollapseNotification **notification** [40](#)
NSOutlineViewItemWillExpandNotification **notification** [40](#)
NSOutlineViewSelectionDidChangeNotification **notification** [40](#)
NSOutlineViewSelectionIsChangingNotification **notification** [41](#)

O

outlineTableColumn **instance method** [20](#)
outlineView:dataCellForTableColumn:item: <NSObject> **delegate method** [25](#)
outlineView:didClickTableColumn: <NSObject> **delegate method** [26](#)
outlineView:didDragTableColumn: <NSObject> **delegate method** [26](#)
outlineView:heightOfRowByItem: <NSObject> **delegate method** [26](#)
outlineView:isGroupItem: <NSObject> **delegate method** [27](#)

outlineView:mouseDownInHeaderOfTableColumn:
 <NSObject> delegate method [27](#)
 outlineView:nextTypeSelectMatchFromItem:toItem:
 forString: <NSObject> delegate method [28](#)
 outlineView:selectionIndexesForProposedSelection:
 <NSObject> delegate method [28](#)
 outlineView:shouldCollapseItem: <NSObject>
 delegate method [29](#)
 outlineView:shouldEditTableColumn:item:
 <NSObject> delegate method [29](#)
 outlineView:shouldExpandItem: <NSObject>
 delegate method [30](#)
 outlineView:shouldSelectItem: <NSObject>
 delegate method [30](#)
 outlineView:shouldSelectTableColumn:
 <NSObject> delegate method [31](#)
 outlineView:shouldShowCellExpansionForTableColumn:
 item: <NSObject> delegate method [31](#)
 outlineView:shouldTrackCell:forTableColumn:item:
 <NSObject> delegate method [31](#)
 outlineView:shouldTypeSelectForEvent:
 withCurrentSearchString: <NSObject> delegate
 method [32](#)
 outlineView:toolTipForCell:rect:tableColumn:item:
 mouseLocation: <NSObject> delegate method [33](#)
 outlineView:typeSelectStringForTableColumn:item:
 <NSObject> delegate method [33](#)
 outlineView:willDisplayCell:forTableColumn:item:
 <NSObject> delegate method [34](#)
 outlineView:willDisplayOutlineCell:forTableColumn:
 item: <NSObject> delegate method [34](#)
 outlineViewColumnDidMove: <NSObject> delegate
 method [35](#)
 outlineViewColumnDidResize: <NSObject> delegate
 method [35](#)
 outlineViewItemDidCollapse: <NSObject> delegate
 method [35](#)
 outlineViewItemDidExpand: <NSObject> delegate
 method [36](#)
 outlineViewItemWillCollapse: <NSObject> delegate
 method [36](#)
 outlineViewItemWillExpand: <NSObject> delegate
 method [36](#)
 outlineViewSelectionDidChange: <NSObject>
 delegate method [37](#)
 outlineViewSelectionIsChanging: <NSObject>
 delegate method [37](#)

P

parentForItem: instance method [20](#)

R

reloadItem: instance method [20](#)
 reloadItem:reloadChildren: instance method [21](#)
 rowForItem: instance method [21](#)

S

selectionShouldChangeInOutlineView: <NSObject>
 delegate method [37](#)
 setAutoresizesOutlineColumn: instance method [22](#)
 setAutosaveExpandedItems: instance method [22](#)
 setDataSource: instance method [23](#)
 setDropItem:dropChildIndex: instance method [23](#)
 setIndentationMarkerFollowsCell: instance
 method [23](#)
 setIndentationPerLevel: instance method [24](#)
 setOutlineTableColumn: instance method [24](#)
 shouldCollapseAutoExpandedItemsForDeposited:
 instance method [25](#)