# NSOutlineViewDataSource Protocol Reference

Developer

# Contents

# NSOutlineViewDataSource Protocol Reference

| | |
|---|---|
| **Conforms to** | NSObject |
| **Framework** | /System/Library/Frameworks/AppKit.framework |
| **Availability** | Available in OS X v10.6 and later. |
| **Companion guide** | Outline View Programming Topics |
| **Declared in** | NSOutlineView.h |
| **Related sample code** | SidebarDemo |

## Overview

`NSOutlineView` objects support a data source delegate in addition to the regular delegate object. The `NSOutlineViewDataSource` protocol defines methods that the outline view invokes as necessary to retrieve data and information about the data from the data source delegate, and—optionally—to update data values.

All the methods in the `NSOutlineViewDataSource` protocol are marked as `@optional`. While this is true, there are cases were you must implement some methods to achieve required functionality, specifically when working with conventional data sources rather than data that is provided by Cocoa bindings.

### Required and Optional Methods Using Programmatic Conventions and Cocoa Bindings

If you are using conventional data sources for content you must implement the basic methods that provide the outline view with data: `outlineView:child:ofItem:` (page 6), `outlineView:isItemExpandable:` (page 9), `outlineView:numberOfChildrenOfItem:` (page 11), and `outlineView:objectValueForTableColumn:byItem:` (page 12). Applications that acquire their data using Cocoa bindings do not need to implement these methods.

Similarly, when using conventional data sources , if you want to allow the user to edit values, you must implement `outlineView:setObjectValue:forTableColumn:byItem:` (page 14). When these methods are invoked by the outline view, `nil` as the `item` refers to the "root" item. `NSOutlineView` requires that each

item in the outline view be unique. In order for the collapsed state of an outline view to remain consistent between reloads you must always return the same object for an item. When using Cocoa bindings to provide outline view content, there is no requirement to implement this method.

> **Note:** Some of the methods in this `protocol`, such as `outlineView:child:ofItem:` (page 6) and `outlineView:numberOfChildrenOfItem:` (page 11) along with other methods that return data, are called very frequently, so they must be efficient.

# Tasks

## Working with Items in a View

– `outlineView:child:ofItem:` (page 6)
Returns the child item at the specified index of a given item.

– `outlineView:isItemExpandable:` (page 9)
Returns a Boolean value that indicates whether the a given item is expandable.

– `outlineView:numberOfChildrenOfItem:` (page 11)
Returns the number of child items encompassed by a given item.

– `outlineView:objectValueForTableColumn:byItem:` (page 12)
Invoked by `outlineView` to return the data object associated with the specified `item`.

– `outlineView:setObjectValue:forTableColumn:byItem:` (page 14)
Set the data object for a given item in a given column.

## Supporting Drag and Drop

– `outlineView:acceptDrop:item:childIndex:` (page 5)
Returns a Boolean value that indicates whether a drop operation was successful.

– `outlineView:validateDrop:proposedItem:proposedChildIndex:` (page 16)
Used by an outline view to determine a valid drop target.

– `outlineView:namesOfPromisedFilesDroppedAtDestination:forDraggedItems:` (page 10)
Returns an array of filenames for the created files that the receiver promises to create.

– `outlineView:draggingSession:endedAtPoint:operation:` (page 7) *required method*
Implement this method to know when the given dragging session has ended. (required)

– `outlineView:draggingSession:willBeginAtPoint:forItems:` (page 8) *required method*

Implement this method know when the given dragging session is about to begin and potentially modify the dragging session. (required)

– `outlineView:pasteboardWriterForItem:` (page 13) *required method*

Implement this method to enable the table to be an `NSDraggingSource` that supports dragging multiple items. (required)

– `outlineView:updateDraggingItemsForDrag:` (page 16) *required method*

Implement this method to enable the table to update dragging items as they are dragged over the view. (required)

### Supporting Object Persistence

– `outlineView:itemForPersistentObject:` (page 10)

Invoked by `outlineView` to return the item for the archived `object`.

– `outlineView:persistentObjectForItem:` (page 13)

Invoked by `outlineView` to return an archived object for `item`.

### Working with a Pasteboard

– `outlineView:writeItems:toPasteboard:` (page 17)

Returns a Boolean value that indicates whether a drag operation is allowed.

### Sorting

– `outlineView:sortDescriptorsDidChange:` (page 15)

Invoked by an outline view to notify the data source that the descriptors changed and the data may need to be resorted.

## Instance Methods

### outlineView:acceptDrop:item:childIndex:

*Returns a Boolean value that indicates whether a drop operation was successful.*

```
- (BOOL)outlineView:(NSOutlineView *)outlineView acceptDrop:(id<NSDraggingInfo>)info
 item:(id)item childIndex:(NSInteger)index
```

**Parameters**

`outlineView`

The outline view that sent the message. `outlineView` must have previously allowed a drop.

`info`

An object that contains more information about this dragging operation.

`item`

The parent of the item over which the cursor was placed when the mouse button was released.

`index`

The index of the child of `item` over which the cursor was placed when the mouse button was released.

**Return Value**

`YES` if the drop operation was successful, otherwise `NO`.

**Discussion**

The data source should incorporate the data from the dragging pasteboard in the implementation of this method. You can get the data for the drop operation from info using the `draggingPasteboard` method.

The return value indicates success or failure of the drag operation to the system.

**Availability**

Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

**See Also**

`shouldCollapseAutoExpandedItemsForDeposited:` (NSOutlineView)

**Declared in**

`NSOutlineView.h`

## outlineView:child:ofItem:

*Returns the child item at the specified index of a given item.*

```
- (id)outlineView:(NSOutlineView *)outlineView child:(NSInteger)index ofItem:(id)item
```

**Parameters**

`outlineView`

The outline view that sent the message.

`index`
The index of the child item from `item` to return.

`item`
An item in the data source.

**Return Value**
The child item at `index` of a `item`. If `item` is `nil`, returns the appropriate child item of the root object.

**Discussion**
Children of a given parent `item` are accessed sequentially. In order for the collapsed state of the outline view to remain consistent when it is reloaded you must always return the same object for a specified `child` and `item`.

> **Important:** While this method is marked as `@optional` in the protocol, **you must implement this method if you are not providing the data for the outline view using Cocoa bindings.**
>
> Do not call `reloadData` from this method.

**Special Considerations**
The `outlineView:child:ofItem:` (page 6) method is called very frequently, so it must be efficient.

**Availability**
Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

**See Also**
– `outlineView:numberOfChildrenOfItem:` (page 11)

**Declared in**
`NSOutlineView.h`

## outlineView:draggingSession:endedAtPoint:operation:

*Implement this method to know when the given dragging session has ended. (required)*

```
– (void)outlineView:(NSOutlineView *)outlineView draggingSession:(NSDraggingSession
 *)session endedAtPoint:(NSPoint)screenPoint operation:(NSDragOperation)operation
```

**Parameters**
`outlineView`
The outline view in which the drag began.

`session`
The dragging session that ended.

`screenPoint`
The point onscreen at which the drag ended.

`operation`
A mask specifying the types of drag operations permitted by the dragging source.

**Discussion**
You can implement this optional delegate method to know when the dragging source operation ended at a specific location, such as the trash (by checking for an operation of `NSDragOperationDelete`).

**Availability**
Available in OS X v10.7 and later.

**Declared in**
`NSOutlineView.h`

## outlineView:draggingSession:willBeginAtPoint:forItems:

*Implement this method know when the given dragging session is about to begin and potentially modify the dragging session. (required)*

```
– (void)outlineView:(NSOutlineView *)outlineView draggingSession:(NSDraggingSession
 *)session willBeginAtPoint:(NSPoint)screenPoint forItems:(NSArray *)draggedItems
```

**Parameters**
`outlineView`
The outline view in which the drag is about to begin.

`session`
The dragging session that is about to begin.

`screenPoint`
The point onscreen at which the drag is to begin.

`draggedItems`
A array of items to be dragged, excluding items for which `outlineView:pasteboardWriterForItem:` (page 13) returns `nil`.

### Discussion

The `draggedItems` array directly matches the pasteboard writer array used to begin the dragging session with the `NSView` method `beginDraggingSessionWithItems:event:source:`. Hence, the order is deterministic, and can be used in `outlineView:acceptDrop:item:childIndex:` (page 5) when enumerating the `NSDraggingInfo` protocol's pasteboard classes.

### Availability

Available in OS X v10.7 and later.

### Declared in

`NSOutlineView.h`

## outlineView:isItemExpandable:

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

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

### Parameters

`outlineView`

The outline view that sent the message.

`item`

An item in the data source.

### Return Value

`YES` if `item` can be expanded to display its children, otherwise `NO`.

### Discussion

This method may be called quite often, so it must be efficient.

> **Important:** While this method is marked as `@optional` in the protocol, **you must implement this method if you are not providing the data for the outline view using Cocoa bindings.**
>
> Do not call `reloadData` from this method.

### Availability

Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

### Declared in

`NSOutlineView.h`

## outlineView:itemForPersistentObject:

*Invoked by `outlineView` to return the item for the archived `object`.*

```
– (id)outlineView:(NSOutlineView *)outlineView itemForPersistentObject:(id)object
```

**Parameters**

`outlineView`

The outline view that sent the message.

`object`

An archived representation of an item in `outlineView`'s data source.

**Return Value**

The unarchived item corresponding to `object`. If the item is an archived object, this method may return the object.

**Discussion**

When the outline view is restoring the saved expanded items, this method is called for each expanded item, to translate the archived object to an outline view item.

**Special Considerations**

You must implement this method if you are automatically saving expanded items (that is, if `autosaveExpandedItems` returns `YES`).

**Availability**

Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

**Declared in**

`NSOutlineView.h`

## outlineView:namesOfPromisedFilesDroppedAtDestination:forDraggedItems:

*Returns an array of filenames for the created files that the receiver promises to create.*

```
– (NSArray *)outlineView:(NSOutlineView *)outlineView
namesOfPromisedFilesDroppedAtDestination:(NSURL *)dropDestination
forDraggedItems:(NSArray *)items
```

**Parameters**

`outlineView`

The outline view that sent the message.

`dropDestination`
The drop location where the files are created.

`items`
The items being dragged.

**Return Value**
An array of filenames (not full paths) for the created files that the receiver promises to create.

**Discussion**
For more information on file promise dragging, see documentation on the `NSDraggingSource` protocol and `namesOfPromisedFilesDroppedAtDestination:`.

**Availability**
Available in OS X v10.4 and later.

Available as part of an informal protocol prior to OS X v10.6.

**Declared in**
`NSOutlineView.h`

## outlineView:numberOfChildrenOfItem:

*Returns the number of child items encompassed by a given item.*

```
- (NSInteger)outlineView:(NSOutlineView *)outlineView numberOfChildrenOfItem:(id)item
```

**Parameters**
`outlineView`
The outline view that sent the message.

`item`
An item in the data source.

**Return Value**
The number of child items encompassed by `item`. If `item` is `nil`, this method should return the number of children for the top-level item.

**Discussion**
The `outlineView:numberOfChildrenOfItem:` (page 11) method is called very frequently, so it must be efficient.

**Important:** While this method is marked as `@optional` in the protocol, **you must implement this method if you are not providing the data for the outline view using Cocoa bindings.**

Do not call `reloadData` from this method.

**Availability**
Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

**Declared in**
`NSOutlineView.h`

## outlineView:objectValueForTableColumn:byItem:

*Invoked by `outlineView` to return the data object associated with the specified `item`.*

```
- (id)outlineView:(NSOutlineView *)outlineView
objectValueForTableColumn:(NSTableColumn *)tableColumn byItem:(id)item
```

**Parameters**

`outlineView`
The outline view that sent the message.

`tableColumn`
A column in `outlineView`.

`item`
An item in the data source in the specified `tableColumn` of the view.

**Discussion**
The item is located in the specified `tableColumn` of the view.

**Important:** While this method is marked as `@optional` in the protocol, **you must implement this method if you are not providing the data for the outline view using Cocoa bindings.**

Do not call `reloadData` from this method.

**Availability**
Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

**Declared in**
NSOutlineView.h

## outlineView:pasteboardWriterForItem:

*Implement this method to enable the table to be an `NSDraggingSource` that supports dragging multiple items. (required)*

```
- (id<NSPasteboardWriting>)outlineView:(NSOutlineView *)outlineView
pasteboardWriterForItem:(id)item
```

**Parameters**
`outlineView`
The outline view in which the drag begins.

`item`
The item for which to return a pasteboard writer.

**Return Value**
A custom object that implements `NSPasteboardWriting` protocol (or simply use `NSPasteboardItem`).

**Discussion**
If this method is implemented, then `outlineView:writeItems:toPasteboard:` (page 17) is not called.

**Availability**
Available in OS X v10.7 and later.

**Declared in**
NSOutlineView.h

## outlineView:persistentObjectForItem:

*Invoked by `outlineView` to return an archived object for `item`.*

```
- (id)outlineView:(NSOutlineView *)outlineView persistentObjectForItem:(id)item
```

**Parameters**
`outlineView`
The outline view that sent the message.

`item`
The item for which to return an archived object.

### Return Value

An archived representation of `item`. If the item is an archived object, this method may return the item.

### Discussion

When the outline view is saving the expanded items, this method is called for each expanded item, to translate the outline view item to an archived object.

### Special Considerations

You must implement this method if you are automatically saving expanded items (that is, if `autosaveExpandedItems` returns `YES`).

### Availability

Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

### Declared in

`NSOutlineView.h`

## outlineView:setObjectValue:forTableColumn:byItem:

*Set the data object for a given item in a given column.*

```
- (void)outlineView:(NSOutlineView *)outlineView setObjectValue:(id)object
forTableColumn:(NSTableColumn *)tableColumn byItem:(id)item
```

### Parameters

`outlineView`

The outline view that sent the message.

`object`

The new value for the item.

`tableColumn`

A column in `outlineView`.

`item`

An item in the data source in the specified `tableColumn` of the view.

### Discussion

The item is located in the specified `tableColumn` of the view.

**Important:** While this method is marked as `@optional` in the protocol, **you must implement this method if you are not providing the data for the outline view using Cocoa bindings.**

Do not call `reloadData` from this method.

**Availability**
Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

**Declared in**
`NSOutlineView.h`

## outlineView:sortDescriptorsDidChange:

*Invoked by an outline view to notify the data source that the descriptors changed and the data may need to be resorted.*

```
- (void)outlineView:(NSOutlineView *)outlineView sortDescriptorsDidChange:(NSArray *)oldDescriptors
```

**Parameters**

`outlineView`

The outline view that sent the message.

`oldDescriptors`

An array that contains the previous descriptors.

**Discussion**
The data source typically sorts and reloads the data, and adjusts the selections accordingly. If you need to know the current sort descriptors and the data source does not itself manage them, you can get `outlineView`'s current sort descriptors by sending it a `sortDescriptors` message.

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

Available as part of an informal protocol prior to OS X v10.6.

**Declared in**
`NSOutlineView.h`

## outlineView:updateDraggingItemsForDrag:

*Implement this method to enable the table to update dragging items as they are dragged over the view. (required)*

```
– (void)outlineView:(NSOutlineView *)outlineView
updateDraggingItemsForDrag:(id<NSDraggingInfo>)draggingInfo
```

**Parameters**

`outlineView`

The outline view in which the drag occurs.

`draggingInfo`

The dragging info object.

**Discussion**

Implementing this method is required for multi-image dragging. A typical implementation calls the passed-in dragging info object's `enumerateDraggingItemsWithOptions:forView:classes:searchOptions:usingBlock:` method and sets the dragging item's `imageComponentsProvider` property to a proper image based on the content. For NSView-based table views, you can use the `NSTableCellView` method `draggingImageComponents`.

**Availability**

Available in OS X v10.7 and later.

**Declared in**

`NSOutlineView.h`

## outlineView:validateDrop:proposedItem:proposedChildIndex:

*Used by an outline view to determine a valid drop target.*

```
– (NSDragOperation)outlineView:(NSOutlineView *)outlineView
validateDrop:(id<NSDraggingInfo>)info proposedItem:(id)item
proposedChildIndex:(NSInteger)index
```

**Parameters**

`outlineView`

The outline view that sent the message.

`info`

An object that contains more information about this dragging operation.

`item`

The proposed parent.

`index`
The proposed child location.

**Return Value**
A value that indicates which dragging operation the data source will perform.

**Discussion**
Based on the mouse position, the outline view will suggest a proposed drop location. The data source may "retarget" a drop if desired by calling `setDropItem:dropChildIndex:` and returning something other than `NSDragOperationNone`. You may choose to retarget for various reasons (for example, for better visual feedback when inserting into a sorted position).

Implementation of this method is optional.

**Availability**
Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

**Declared in**
`NSOutlineView.h`

## outlineView:writeItems:toPasteboard:

*Returns a Boolean value that indicates whether a drag operation is allowed.*

```
- (BOOL)outlineView:(NSOutlineView *)outlineView writeItems:(NSArray *)items
toPasteboard:(NSPasteboard *)pboard
```

**Parameters**
`outlineView`
The outline view that invoked the method.

`items`
An array of the items participating in the drag.

`pboard`
The pasteboard to which to write the drag data.

**Return Value**
`YES` if the drag operation is allowed, otherwise `NO`.

### Discussion

Invoked by `outlineView` after it has been determined that a drag should begin, but before the drag has been started.

To refuse the drag, return `NO`. To start a drag, return `YES` and place the drag data onto the `pboard` (data, owner, and so on). The drag image and other drag-related information will be set up and provided by the outline view once this call returns with `YES`.

### Availability

Available in OS X v10.0 and later.

Available as part of an informal protocol prior to OS X v10.6.

### Declared in

`NSOutlineView.h`

# Document Revision History

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

| Date | Notes |
| --- | --- |
| 2013-12-16 | Updated for OS X v10.7. Added guidance not to call reloadData from "required" methods that provide the outline view with data. |
| 2010-06-06 | Changed @required methods to be @optional. Updated overview. |
| 2010-03-24 | Corrected description of outlineView:objectValueForTableColumn:byItem:. |
| 2009-04-27 | Updated for OS X v10.6. The NSOutlineViewDataSource informal protocol is now a formal protocol. |
| 2007-02-19 | Corrected minor typographical errors. |
| 2006-05-23 | Clarified that outlineView:objectValueForTableColumn:item: must return the same object for a requested item to maintain the collapsed state between reloads.<br>First publication of this content as a separate document. |

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010