NSTableDataSource Protocol Reference

Cocoa > User Experience

2007-04-10

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

NSTableDataSource Protocol Reference

(informal protocol)

Framework	/System/Library/Frameworks/AppKit.framework
Companion guide	Table View Programming Guide
Declared in	NSTableView.h

Overview

The `NSTableDataSource` informal protocol declares the methods that an instance of `NSTableView` uses to access the contents of its data source object.

> **Note:** Some of the methods in this protocol, such as `tableView:objectValueForTableColumn:row:` (page 8) and `numberOfRowsInTableView:` (page 6) along with other methods that return data, are called very frequently, so they must be efficient.

Tasks

Getting Values

`– numberOfRowsInTableView:` (page 6)
Returns the number of records managed for *`aTableView`* by the data source object.

`– tableView:objectValueForTableColumn:row:` (page 8)
Invoked by the table view to return the data object associated with the specified row and column.

Setting Values

`– tableView:setObjectValue:forTableColumn:row:` (page 8)
Set the data object for an item in a given row in a given column.

Dragging

`– tableView:acceptDrop:row:dropOperation:` (page 7)
Invoked by *`aTableView`* when the mouse button is released over a table view that previously decided to allow a drop.

- `tableView:namesOfPromisedFilesDroppedAtDestination:forDraggedRowsWithIndexes:` (page 7)
  Returns an array of filenames that represent the *`indexSet`* rows for a drag to *`dropDestination`*.
- `tableView:validateDrop:proposedRow:proposedDropOperation:` (page 9)
  Used by *`aTableView`* to determine a valid drop target.
- `tableView:writeRowsWithIndexes:toPasteboard:` (page 10)
  Returns a Boolean value that indicates whether a drag operation is allowed.

Sorting

- `tableView:sortDescriptorsDidChange:` (page 9)
  Invoked by *`aTableView`* to indicate that sorting may need to be done.

Deprecated Methods

- `tableView:writeRows:toPasteboard:` (page 13) Deprecated in Mac OS X v10.4 and later
  This method has been deprecated. You should implement `tableView:writeRowsWithIndexes:toPasteboard:` (page 10) instead.

Instance Methods

numberOfRowsInTableView:

Returns the number of records managed for *`aTableView`* by the data source object.

```
- (NSInteger)numberOfRowsInTableView:(NSTableView *)aTableView
```

Parameters

`aTableView`
  The table view that sent the message.

Return Value
The number of rows in *`aTableView`*.

Discussion
An instance of `NSTableView` uses this method to determine how many rows it should create and display.

Note: `numberOfRowsInTableView:` is called very frequently, so it must be efficient.

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

Declared In
`NSTableView.h`

tableView:acceptDrop:row:dropOperation:

Invoked by *aTableView* when the mouse button is released over a table view that previously decided to allow a drop.

```
- (BOOL)tableView:(NSTableView *)aTableView acceptDrop:(id < NSDraggingInfo >)info
     row:(NSInteger)row dropOperation:(NSTableViewDropOperation)operation
```

Parameters

aTableView

The table view that sent the message.

info

An object that contains more information about this dragging operation.

row

The index of the proposed target row.

operation

The type of dragging operation.

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.

To accept a drop on the second row, *row* would be 2 and *operation* would be NSTableViewDropOn. To accept a drop below the last row, *row* would be [aTableView numberOfRows] and *operation* would be NSTableViewDropAbove.

Implementation of this method is optional.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSTableView.h

tableView:namesOfPromisedFilesDroppedAtDestination:forDraggedRowsWithIndexes:

Returns an array of filenames that represent the *indexSet* rows for a drag to *dropDestination*.

```
- (NSArray *)tableView:(NSTableView *)aTableView
    namesOfPromisedFilesDroppedAtDestination:(NSURL *)dropDestination
    forDraggedRowsWithIndexes:(NSIndexSet *)indexSet
```

Parameters

aTableView

The table view that sent the message.

dropDestination

The drop location where the files are created.

indexSet

The indexes of the items being dragged.

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

Discussion
This method is called when a destination has accepted a promise drag.

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

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

Declared In
`NSTableView.h`

tableView:objectValueForTableColumn:row:

Invoked by the table view to return the data object associated with the specified row and column.

```
- (id)tableView:(NSTableView *)aTableView objectValueForTableColumn:(NSTableColumn
    *)aTableColumn row:(NSInteger)rowIndex
```

Parameters

aTableView
    The table view that sent the message.

aTableColumn
    A column in in *aTableView*.

rowIndex
    The row of the item in *aTableColumn*.

Return Value
An item in the data source in the specified tableColumn of the view.

Discussion

Note: `tableView:objectValueForTableColumn:row:` is called each time the table cell needs to be redisplayed, so it must be efficient.

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

Declared In
`NSTableView.h`

tableView:setObjectValue:forTableColumn:row:

Set the data object for an item in a given row in a given column.

```
- (void)tableView:(NSTableView *)aTableView setObjectValue:(id)anObject
   forTableColumn:(NSTableColumn *)aTableColumn row:(NSInteger)rowIndex
```

Parameters

aTableView
: The table view that sent the message.

anObject
: The new value for the item.

aTableColumn
: A column in *aTableView*.

rowIndex
: The row of the item in *aTableColumn*.

Discussion

Implementation of this method is optional.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSTableView.h`

tableView:sortDescriptorsDidChange:

Invoked by *aTableView* to indicate that sorting may need to be done.

```
- (void)tableView:(NSTableView *)aTableView sortDescriptorsDidChange:(NSArray
    *)oldDescriptors
```

Parameters

aTableView
: The table 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 manage them itself, you can get the current sort descriptors by sending *aTableView* a `sortDescriptors` message.

Implementation of this method is optional.

Availability

Available in Mac OS X v10.3 and later.

Declared In

`NSTableView.h`

tableView:validateDrop:proposedRow:proposedDropOperation:

Used by *aTableView* to determine a valid drop target.

```
- (NSDragOperation)tableView:(NSTableView *)aTableView validateDrop:(id <
    NSDraggingInfo >)info proposedRow:(NSInteger)row
    proposedDropOperation:(NSTableViewDropOperation)operation
```

Parameters

aTableView

The table view that sent the message.

info

An object that contains more information about this dragging operation.

row

The index of the proposed target row.

operation

The type of dragging operation proposed.

Return Value

The dragging operation the data source will perform.

Discussion

The data source may "retarget" a drop if desired by calling `setDropRow:dropOperation:` and returning something other than `NSDragOperationNone`. One may choose to retarget for various reasons (e.g. for better visual feedback when inserting into a sorted position).

To propose a drop on the second row, *row* would be 2 and *operation* would be `NSTableViewDropOn`. To propose a drop below the last row, *row* would be `[aTableView numberOfRows]` and *operation* would be `NSTableViewDropAbove`.

Implementation of this method is optional.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSTableView.h`

tableView:writeRowsWithIndexes:toPasteboard:

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

```
- (BOOL)tableView:(NSTableView *)aTableView writeRowsWithIndexes:(NSIndexSet
    *)rowIndexes toPasteboard:(NSPasteboard *)pboard
```

Parameters

aTableView

The table view that sent the message.

rowIndexes

An index set of row numbers that will be participating in the drag.

pboard

The pasteboard to which to write the drag data.

Return Value

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

Discussion

Invoked by *aTableView* 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 *`pboard`* (data, owner, and so on). The drag image and other drag-related information will be set up and provided by the table view once this call returns with `YES`.

Implementation of this method is optional.

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

Declared In
`NSTableView.h`

Deprecated NSTableDataSource Methods

A method identified as deprecated has been superseded and may become unsupported in the future.

Deprecated in Mac OS X v10.4 and later

tableView:writeRows:toPasteboard:

This method has been deprecated. You should implement `tableView:writeRowsWithIndexes:toPasteboard:` (page 10) instead. (Deprecated in Mac OS X v10.4 and later.)

```
- (BOOL)tableView:(NSTableView *)aTableView writeRows:(NSArray *)rows
    toPasteboard:(NSPasteboard *)pboard
```

Discussion

Invoked by `aTableView` 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 `pboard` (data, owner, and so on). The drag image and other drag-related information will be set up and provided by the table view once this call returns with `YES`. `rows` is the list of row numbers that will be participating in the drag.

Implementation of this method is optional.

Availability

Deprecated in Mac OS X v10.4 and later.

Declared In

`NSTableView.h`

Document Revision History

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

Date	Notes
2007-04-10	Corrected minor typos.
2006-05-23	Corrected minor typos.
	First publication of this content as a separate document.

Index