NSTextTable Class Reference

Cocoa > Text & Fonts

2007-04-11

Apple Inc.
© 2007 Apple Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

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

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

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

NSTextTable Class Reference 5

Overview 5
Tasks 5
- Getting and Setting Number of Columns 5
- Getting and Setting Layout Algorithm 5
- Collapsing Borders 6
- Hiding Empty Cells 6
- Determining Layout Rectangles 6
- Drawing the Table 6

Instance Methods 6
- boundsRectForBlock:contentRect:inRect:textContainer:characterRange: 6
- collapsesBorders 7
- drawBackgroundForBlock:withFrame:inView:characterRange:layoutManager: 7
- hidesEmptyCells 8
- layoutAlgorithm 8
- numberOfColumns 9
- rectForBlock:layoutAtPoint:inRect:textContainer:characterRange: 9
- setCollapsesBorders: 10
- setHidesEmptyCells: 10
- setLayoutAlgorithm: 11
- setNumberOfColumns: 11

Constants 12
- NSTextTableLayoutAlgorithm 12

Document Revision History 13

Index 15

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

NSTextTable Class Reference

Inherits from	NSTextBlock : NSObject
Conforms to	NSCoding (NSTextBlock) NSCopying (NSTextBlock) NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.4 and later.
Declared in	NSTextTable.h
Companion guides	Text System Overview Text Layout Programming Guide for Cocoa
Related sample code	iSpend

Overview

An `NSTextTable` object represents a text table as a whole. It is responsible for laying out and drawing the text table blocks it contains, and it maintains the basic parameters of the table.

Tasks

Getting and Setting Number of Columns

`- numberOfColumns` (page 9)
Returns the number of columns in the text table.

`- setNumberOfColumns:` (page 11)
Sets the number of columns in the text table.

Getting and Setting Layout Algorithm

`- layoutAlgorithm` (page 8)
Returns the text table layout algorithm.

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

- `setLayoutAlgorithm:` (page 11)
  Sets the text table layout algorithm.

Collapsing Borders

- `collapsesBorders` (page 7)
  Returns whether the text table borders are collapsible.
- `setCollapsesBorders:` (page 10)
  Sets whether the text table borders are collapsible.

Hiding Empty Cells

- `hidesEmptyCells` (page 8)
  Returns whether the text table hides empty cells.
- `setHidesEmptyCells:` (page 10)
  Sets whether the text table hides empty cells.

Determining Layout Rectangles

- `rectForBlock:layoutAtPoint:inRect:textContainer:characterRange:` (page 9)
  Returns the rectangle within which glyphs should be laid out for a text table block.
- `boundsRectForBlock:contentRect:inRect:textContainer:characterRange:` (page 6)
  Returns the rectangle the text table block actually occupies, including padding, borders, and margins.

Drawing the Table

- `drawBackgroundForBlock:withFrame:inView:characterRange:layoutManager:` (page 7)
  Draws any colors and other decorations for a text table block.

Instance Methods

boundsRectForBlock:contentRect:inRect:textContainer:characterRange:

Returns the rectangle the text table block actually occupies, including padding, borders, and margins.

```
- (NSRect)boundsRectForBlock:(NSTextTableBlock *)block
    contentRect:(NSRect)contentRect inRect:(NSRect)rect
    textContainer:(NSTextContainer *)textContainer characterRange:(NSRange)charRange
```

Parameters

block

The text table block that wants to determine where to layout its glyphs.

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

contentRect
The actual rectangle in which the text was laid out, as determined by `rectForLayoutAtPoint:inRect:textContainer:characterRange:`.

rect
The initial rectangle in *textContainer* proposed by the typesetter.

textContainer
The text container being used for the layout.

charRange
The range of the characters whose glyphs are to be drawn.

Return Value
The rectangle the text table block actually occupies, including padding, borders, and margins.

Discussion
This method is called by the text table block *block* after it is laid out to determine the rectangle the text table block actually occupies, including padding, borders, and margins.

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

See Also
- rectForBlock:layoutAtPoint:inRect:textContainer:characterRange: (page 9)

Declared In
NSTextTable.h

collapsesBorders

Returns whether the text table borders are collapsible.

```
- (BOOL)collapsesBorders
```

Return Value
`YES` if the text table borders are collapsible, `NO` otherwise.

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

See Also
- setCollapsesBorders: (page 10)

Declared In
NSTextTable.h

drawBackgroundForBlock:withFrame:inView:characterRange:layoutManager:

Draws any colors and other decorations for a text table block.

```
- (void)drawBackgroundForBlock:(NSTextTableBlock *)block withFrame:(NSRect)frameRect
    inView:(NSView *)controlView characterRange:(NSRange)charRange
   layoutManager:(NSLayoutManager *)layoutManager
```


2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

Parameters

block

The text table block that wants to draw its background.

frameRect

The area in which drawing occurs.

controlView

The view controlling the drawing.

charRange

The range of the characters whose glyphs are to be drawn.

layoutManager

The layout manager controlling the typesetting.

Discussion

This methods is called by the text table block *block* to draw any colors and other decorations before the text is drawn.

Availability

Available in Mac OS X v10.4 and later.

Declared In

NSTextTable.h

hidesEmptyCells

Returns whether the text table hides empty cells.

```
- (BOOL)hidesEmptyCells
```

Return Value

YES if the text table hides empty cells, NO otherwise.

Discussion

If empty cells are hidden, locations with empty cells allow the background of the enclosing block or text container to show through.

Availability

Available in Mac OS X v10.4 and later.

See Also

- setHidesEmptyCells: (page 10)

Declared In

NSTextTable.h

layoutAlgorithm

Returns the text table layout algorithm.

```
- (NSTextTableLayoutAlgorithm)layoutAlgorithm
```

Return Value

The text table layout algorithm.

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

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

See Also
- `setLayoutAlgorithm:` (page 11)

Declared In
`NSTextTable.h`

numberOfColumns

Returns the number of columns in the text table.

```
- (NSUInteger)numberOfColumns
```

Return Value
The number of columns in the text table.

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

See Also
- `setNumberOfColumns:` (page 11)

Declared In
`NSTextTable.h`

rectForBlock:layoutAtPoint:inRect:textContainer:characterRange:

Returns the rectangle within which glyphs should be laid out for a text table block.

```
- (NSRect)rectForBlock:(NSTextTableBlock *)block layoutAtPoint:(NSPoint)startingPoint
    inRect:(NSRect)rect textContainer:(NSTextContainer *)textContainer
   characterRange:(NSRange)charRange
```

Parameters

block
The text table block that wants to determine where to layout its glyphs.

startingPoint
The location, in container coordinates, where layout begins.

rect
The rectangle in which the block is constrained to lie. For top-level blocks, this is the container rectangle of *textContainer*; for nested blocks, this is the layout rectangle of the enclosing block.

textContainer
The text container being used for the layout.

charRange
The range of the characters whose glyphs are to be drawn.

Return Value
The rectangle within which glyphs should be laid out.

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

Discussion
This method is called by the text table block *block* to determine the rectangle within which glyphs should be laid out for the text table block.

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

See Also
- boundsRectForBlock:contentRect:inRect:textContainer:characterRange: (page 6)

Declared In
NSTextTable.h

setCollapsesBorders:

Sets whether the text table borders are collapsible.

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

Parameters
flag
YES if the text table borders should be collapsible, NO otherwise.

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

See Also
- collapsesBorders (page 7)

Related Sample Code
iSpend

Declared In
NSTextTable.h

setHidesEmptyCells:

Sets whether the text table hides empty cells.

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

Parameters
flag
YES if the text table should hide empty cells, NO otherwise.

Discussion
If empty cells are hidden, locations with empty cells allow the background of the enclosing block or text container to show through.

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

See Also
- hidesEmptyCells (page 8)

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

Related Sample Code
iSpend

Declared In
`NSTextTable.h`

setLayoutAlgorithm:

Sets the text table layout algorithm.

```
- (void)setLayoutAlgorithm:(NSTextTableLayoutAlgorithm)algorithm
```

Parameters

algorithm
The new layout algorithm.

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

See Also
- `layoutAlgorithm` (page 8)

Related Sample Code
iSpend

Declared In
`NSTextTable.h`

setNumberOfColumns:

Sets the number of columns in the text table.

```
- (void)setNumberOfColumns:(NSUInteger)numCols
```

Parameters

numCols
The new number of columns.

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

See Also
- `numberOfColumns` (page 9)

Related Sample Code
iSpend

Declared In
`NSTextTable.h`

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

Constants

NSTextTableLayoutAlgorithm

These constants, specifying the type of text table layout algorithm, are used with `setLayoutAlgorithm:` (page 11).

```
enum {
    NSTextTableAutomaticLayoutAlgorithm = 0,
    NSTextTableFixedLayoutAlgorithm     = 1
};
typedef NSUInteger NSTextTableLayoutAlgorithm;
```

Constants

`NSTextTableAutomaticLayoutAlgorithm`

Specifies automatic layout algorithm

Available in Mac OS X v10.4 and later.

Declared in `NSTextTable.h`.

`NSTextTableFixedLayoutAlgorithm`

Specifies fixed layout algorithm

Available in Mac OS X v10.4 and later.

Declared in `NSTextTable.h`.

Declared In

`NSTextTable.h`

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

Document Revision History

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

Date	Notes
2007-04-11	Reformatted descriptions of constants.
2006-05-23	First publication of this content as a separate document.

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

Document Revision History

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.

Index

B

boundsRectForBlock:contentRect:inRect:textContainer:characterRange: instance method 6

C

collapsesBorders instance method 7

D

drawBackgroundForBlock:withFrame:inView:characterRange:layoutManager: instance method 7

H

hidesEmptyCells instance method 8

L

layoutAlgorithm instance method 8

N

NSTextTableAutomaticLayoutAlgorithm constant 12
NSTextTableFixedLayoutAlgorithm constant 12
NSTextTableLayoutAlgorithm 12
numberOfColumns instance method 9

R

rectForBlock:layoutAtPoint:inRect:textContainer:characterRange: instance method 9

S

setCollapsesBorders: instance method 10
setHidesEmptyCells: instance method 10
setLayoutAlgorithm: instance method 11
setNumberOfColumns: instance method 11

2007-04-11 | © 2007 Apple Inc. All Rights Reserved.