

---

# Carbon Reference Update

[Carbon > Human Interface Toolbox](#)



2007-07-18



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

.Mac is a registered service mark of Apple Inc.

Apple, the Apple logo, AppleScript, Bonjour, Carbon, Cocoa, ColorSync, eMac, FireWire, Mac, Mac OS, MacApp, Objective-C, Panther, Quartz, QuickDraw, QuickTime, and Spaces are trademarks of Apple Inc., registered in the United States and other countries.

Aperture, Finder, and Spotlight are trademarks 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

---

## **Introduction to Carbon Reference Update 5**

---

Organization of This Document 5  
See Also 5

---

## **10.5 Symbol Changes 7**

---

Classes 7  
Protocols 7  
    IMKTextInput (New) 7  
    IMKUnicodeTextInput (New) 8  
C Symbols 8  
    CommonPanels 8  
    OpenScripting 9  
    HI Toolbox 10  
    Print 38  
    ImageCapture 39  
    NavigationServices 45

---

## **10.4 Symbol Changes 47**

---

Classes 47  
    NSObject 47  
C Symbols 48  
    OpenScripting 48  
    HI Toolbox 49  
    Print 76  
    Ink 77  
    ImageCapture 78  
    NavigationServices 80

---

## **10.3 Symbol Changes 81**

---

Classes 81  
C Symbols 81  
    CarbonSound 81  
    CommonPanels 82  
    OpenScripting 83  
    HI Toolbox 84  
    Print 116  
    Ink 117  
    ImageCapture 121

## **10.2 Symbol Changes 123**

---

Classes 123  
C Symbols 123  
    CarbonSound 123  
    CommonPanels 126  
    SecurityHI 128  
    OpenScripting 128  
    HIToolbox 129  
    Print 158  
    ImageCapture 159

## **10.1 Symbol Changes 161**

---

Classes 161  
C Symbols 161  
    CarbonSound 161  
    CommonPanels 161  
    OpenScripting 162  
    HIToolbox 162  
    Print 179  
    ImageCapture 179  
    NavigationServices 180

## **Document Revision History 183**

---

# Introduction to Carbon Reference Update

---

This document summarizes the symbols that have been added to the Carbon framework. The full reference documentation notes in what version a symbol was introduced, but sometimes it's useful to see only the new symbols for a given release.

If you are not familiar with this framework you should refer to the complete framework reference documentation.

## Organization of This Document

Symbols are grouped by class or protocol for Objective-C and by header file for C. For each symbol there is a link to complete documentation, if available, and a brief description, if available.

## See Also

For reference documentation on this framework, see *Carbon Framework Reference*.



# 10.5 Symbol Changes

---

This article lists the symbols added to `Carbon.framework` in Mac OS X v10.5.

## Classes

All of the classes with new symbols are listed alphabetically, with their new class, instance, and delegate methods described.

## Protocols

All of the protocols with new symbols are listed alphabetically, with their new methods described.

## IMKTextInput (New)

---

Complete reference information is available in the `IMKTextInput` reference.

### Instance Methods

---

<code>attributedStringFromRange:</code>	Returns the attributed string for the provided range of text.
<code>attributesForCharacterIndex:lineHeightRectangle:</code>	Returns a dictionary of text attributes for the text at the provided character index.
<code>bundleIdentifier</code>	Returns the bundle identifier for the process that the input session is attached to.
<code>characterIndexForPoint:tracking:inMarkedRange:</code>	Returns the location in the text document that maps to a global point.
<code>insertText:replacementRange:</code>	Sends fully converted text to an input session.
<code>length</code>	Returns the length of the text document.
<code>markedRange</code>	Returns the range in the document that is occupied by marked text, that is, the current inline session.

<code>overrideKeyboardWithKeyboardNamed:</code>	Overrides the current keyboard.
<code>selectedRange</code>	Returns the current selection range.
<code>selectInputMode:</code>	Selects a new input mode.
<code>setMarkedText:selectionRange:replacementRange:</code>	Inserts the provided text and marks it to indicate that it is part of an active input session.
<code>supportsUnicode</code>	Tests to see if the current input session supports Unicode text.
<code>validAttributesForMarkedText</code>	Returns an array of names for the attributes supported by the receiver.
<code>windowLevel</code>	Returns the window level for a client window.

## IMKUnicodeTextInput (New)

---

Complete reference information is available in the `IMKUnicodeTextInput` reference.

### Instance Methods

---

<code>insertText:</code>	
--------------------------	--

## C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

## CommonPanels

---

### ColorPicker.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>ColorChangedUPP</code>	Defines a universal procedure pointer to a color-changed callback.
<code>UserEventUPP</code>	Defines a universal procedure pointer to an event-filter callback.



## FontPanel.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kEventParamCTFontDescriptor	
kEventParamViewAttributesDictionary	
kFontPanelBackgroundColorAttributeName	
kFontPanelMouseTrackingState	
kFontSelectionCoreTextType	
typeCTFontDescriptorRef	

## OpenScripting

---

### AppleScript.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

ASCopySourceAttributes	Gets the current text style attributes AppleScript uses to display script text.
ASSetSourceAttributes	Sets the text style attributes used by the AppleScript component to display scripts.

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kASSelectCopySourceAttributes	
kASSelectSetSourceAttributes	

## OSA.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

OSACopyDisplayString	Converts a script value to an attributed Unicode text string, which your application can display to the user.
OSACopySourceString	Decompiles the script data for the specified script and returns a copy of the equivalent source data as an attributed Unicode text string.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kOSASelectCopyDisplayString	
kOSASelectCopySourceString	

## HI Toolbox

---

### AEInteraction.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

AEProcessEvent	
----------------	--

### Appearance.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kAppearanceRegionToolbarButton	
kThemeAdornmentArrowRightArrow	
kThemeMetricButtonRoundedHeight	The height of a round-ended button (for example, the Kind button in a Finder Search query.)

<code>kThemeMetricButtonRoundedRecessedHeight</code>	The height of the inset round-ended button (for example, the Servers button in a Finder Search query.)
<code>kThemeMetricSeparatorSize</code>	The height of a horizontal separator, or the width of a vertical separator, drawn with the <code>HIThemeDrawSeparator</code> theme primitive.
<code>kThemeMetricTexturedPushButtonHeight</code>	The height of the push button control designed for use in a textured window.
<code>kThemeMetricTexturedSmallPushButtonHeight</code>	The height of the small push button control designed for use in a textured window.
<code>kThemeSpecifiedFont</code>	
<code>kThemeTrackHideTrack</code>	

## CarbonEvents.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kEventAppUpdateDockTile</code>	
<code>kEventControlFocusPartChanged</code>	
<code>kEventControlOptimalBoundsChanged</code>	
<code>kEventHotKeyExclusive</code>	
<code>kEventHotKeyNoOptions</code>	
<code>kEventMouseScroll</code>	
<code>kEventParamAppleEvent</code>	
<code>kEventParamAppleEventReply</code>	
<code>kEventParamCGImageRef</code>	
<code>kEventParamControlDrawEngraved</code>	
<code>kEventParamDisplayChangeFlags</code>	
<code>kEventParamDisplayDevice</code>	
<code>kEventParamMenuContextHeight</code>	
<code>kEventParamMouseWheelSmoothHorizontalDelta</code>	

## 10.5 Symbol Changes

kEventParamMouseWheelSmoothVerticalDelta	
kEventParamPostOptions	
kEventParamTextInputReplyATSTFont	
kEventParamTextInputReplyAttributedString	
kEventParamTextInputReplyCTFontRef	
kEventParamTextInputSendAttributedString	
kEventParamTextInputSendMouseEvent	
kEventParamTSMdocAccessBaselineDelta	
kEventParamTSMdocAccessReplyCTFontRef	
kEventParamTSMdocAccessReplyCTGlyphInfoRef	
kEventParamWindowContentBounds	
kEventSystemDisplayReconfigured	
kEventSystemDisplaysAsleep	
kEventSystemDisplaysAwake	
kEventTextInputIsMouseEventInInlineInputArea	
kEventTSMdocumentAccessGetFirstRectForRange	
kEventWindowAttributesChanged	
kEventWindowFocusLost	
kEventWindowFocusRestored	
kEventWindowGetIdealStandardState	
kEventWindowTitleChanged	
kEventWindowUpdateDockTile	
kHICommandCloseFile	
kMenuContextDontUpdateEnabled	
kMenuContextDontUpdateIcon	
kMenuContextDontUpdateKey	
kMenuContextDontUpdateText	
kMenuContextInspection	

## 10.5 Symbol Changes

kMouseParamsClickAndHold	
kMouseParamsDragInitiation	
kMouseParamsProxyIcon	
kMouseTrackingClientEvent	
kMouseTrackingScrollWheel	
kTrackMouseLocationOptionIncludeScrollWheel	
typeByteCount	
typeByteOffset	
typeCGDisplayChangeFlags	
typeCGDisplayID	
typeCGFloat	
typeCGFloat72DPIGlobal	
typeCGFloatScreenPixel	
typeCGImageRef	
typeEventTargetOptions	
typeHPoint72DPIGlobal	
typeHPointScreenPixel	
typeHRect72DPIGlobal	
typeHRectScreenPixel	
typeHSize72DPIGlobal	
typeHSizeScreenPixel	
typeRefCon	
typeSignedByteCount	
typeSignedByteOffset	

## CarbonEventsCore.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CopyEventCGEvent	
CreateEventWithCGEvent	
HIGetMousePosition	
RemoveEventParameter	
RemoveStandardEventHandler	

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

typeCTFontRef	
typeCTGlyphInfoRef	

### Controls.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlContentAlertIconRes	
kControlContentTag	
kControlThemeTextFontTag	
kControlThemeTextHorizontalFlushTag	
kControlThemeTextInfoTag	
kControlThemeTextTruncationTag	
kControlThemeTextVerticalFlushTag	

### Dialogs.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kAlertFlagsUseCompositing	
kDialogFlagsUseCompositing	

kStdCFStringAlertVersionTwo
-----------------------------

## Events.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kVK_ANSI_0
kVK_ANSI_1
kVK_ANSI_2
kVK_ANSI_3
kVK_ANSI_4
kVK_ANSI_5
kVK_ANSI_6
kVK_ANSI_7
kVK_ANSI_8
kVK_ANSI_9
kVK_ANSI_A
kVK_ANSI_B
kVK_ANSI_Backslash
kVK_ANSI_C
kVK_ANSI_Comma
kVK_ANSI_D
kVK_ANSI_E
kVK_ANSI_Equal
kVK_ANSI_F
kVK_ANSI_G
kVK_ANSI_Grave
kVK_ANSI_H
kVK_ANSI_I

## 10.5 Symbol Changes

kVK_ANSI_J	
kVK_ANSI_K	
kVK_ANSI_Keypad0	
kVK_ANSI_Keypad1	
kVK_ANSI_Keypad2	
kVK_ANSI_Keypad3	
kVK_ANSI_Keypad4	
kVK_ANSI_Keypad5	
kVK_ANSI_Keypad6	
kVK_ANSI_Keypad7	
kVK_ANSI_Keypad8	
kVK_ANSI_Keypad9	
kVK_ANSI_KeypadClear	
kVK_ANSI_KeypadDecimal	
kVK_ANSI_KeypadDivide	
kVK_ANSI_KeypadEnter	
kVK_ANSI_KeypadEquals	
kVK_ANSI_KeypadMinus	
kVK_ANSI_KeypadMultiply	
kVK_ANSI_KeypadPlus	
kVK_ANSI_L	
kVK_ANSI_LeftBracket	
kVK_ANSI_M	
kVK_ANSI_Minus	
kVK_ANSI_N	
kVK_ANSI_O	
kVK_ANSI_P	
kVK_ANSI_Period	



## 10.5 Symbol Changes

kVK_ANSI_Q	
kVK_ANSI_Quote	
kVK_ANSI_R	
kVK_ANSI_RightBracket	
kVK_ANSI_S	
kVK_ANSI_Semicolon	
kVK_ANSI_Slash	
kVK_ANSI_T	
kVK_ANSI_U	
kVK_ANSI_V	
kVK_ANSI_W	
kVK_ANSI_X	
kVK_ANSI_Y	
kVK_ANSI_Z	
kVK_CapsLock	
kVK_Command	
kVK_Control	
kVK_Delete	
kVK_DownArrow	
kVK_End	
kVK_Escape	
kVK_F1	
kVK_F10	
kVK_F11	
kVK_F12	
kVK_F13	
kVK_F14	
kVK_F15	

## 10.5 Symbol Changes

kVK_F16	
kVK_F17	
kVK_F18	
kVK_F19	
kVK_F2	
kVK_F20	
kVK_F3	
kVK_F4	
kVK_F5	
kVK_F6	
kVK_F7	
kVK_F8	
kVK_F9	
kVK_ForwardDelete	
kVK_Function	
kVK_Help	
kVK_Home	
kVK_ISO_Section	
kVK_JIS_Eisu	
kVK_JIS_Kana	
kVK_JIS_KeypadComma	
kVK_JIS_Underscore	
kVK_JIS_Yen	
kVK_LeftArrow	
kVK_Mute	
kVK_Option	
kVK_PageDown	
kVK_PageUp	

kVK_Return	
kVK_RightArrow	
kVK_RightControl	
kVK_RightOption	
kVK_RightShift	
kVK_Shift	
kVK_Space	
kVK_Tab	
kVK_UpArrow	
kVK_VolumeDown	
kVK_VolumeUp	

## HIArchive.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kHIArchiveDecodingForEditor	
-----------------------------	--

## HIButtonViews.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlPushButtonAnimatingTag	
kControlPushButtonContentTag	
kControlPushButtonIconAlignmentTag	

## HIClockView.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlClockAbsoluteTimeTag	
------------------------------	--

## HICocoaView.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HICocoaViewCreate	Creates a Carbon view that serves as a wrapper for a Cocoa view.
HICocoaViewGetView	Returns the Cocoa view associated with an existing Carbon wrapper view.
HICocoaViewSetView	Associates a Cocoa view with a HICocoaView wrapper view.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlKindHICocoaView	Control kind for a view that wraps a Cocoa view.
kEventParamNSView	

## HIDataBrowser.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kDataBrowserAttributeAutoHideScrollBars	
kDataBrowserAttributeReserveGrowBoxSpace	
kEventClassDataBrowser	
kEventDataBrowserDrawCustomItem	
kEventParamDataBrowserItemID	
kEventParamDataBrowserItemState	
kEventParamDataBrowserPropertyID	

## HI DisclosureViews.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HI DisclosureTriangleGetDisplaysTitle	
HI DisclosureTriangleSetDisplaysTitle	

## HI ImageViews.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HI ImageViewGetAutoTransform	
HI ImageViewSetAutoTransform	

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HI ImageViewAutoTransformOptions	
kHI ImageViewAutoTransformNone	
kHI ImageViewAutoTransformOnDeactivate	
kHI ImageViewAutoTransformOnDisable	

## HI Object.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HI ObjectAddDelegate	
HI ObjectCopyDelegates	
HI ObjectFromEventTarget	
HI ObjectGetEventHandlerObject	

HIObjectRemoveDelegate
------------------------

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIDelegatePosition
hiObjectCannotSubclassSingletonErr
hiObjectDelegateAlreadyExistsErr
hiObjectDelegateNotFoundErr
kEventClassDelegate
kEventDelegateGetGroupClasses
kEventDelegateGetTargetClasses
kEventDelegateInstalled
kEventDelegateIsGroup
kEventDelegateRemoved
kEventHIObjectCreatedFromArchive
kEventHIObjectGetInitParameters
kEventParamAfterDelegates
kEventParamBeforeDelegates
kEventParamDecodingForEditor
kEventParamDelegateGroupClasses
kEventParamDelegateGroupParameters
kEventParamDelegateTarget
kEventParamDelegateTargetClasses
kEventParamInitParameters
kHIClassOptionSingleton
kHIDelegateAfter
kHIDelegateAfterKey
kHIDelegateAll

kHIDelegateBefore	
kHIDelegateBeforeKey	
kHIObjectCustomDataDelegateGroupParametersKey	
kHIObjectInitParamDescription	
kHIObjectInitParamEventName	
kHIObjectInitParamEventType	
kHIObjectInitParamUserName	

## HIScrollView.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kHIScrollViewOptionsDisableSmoothScrolling	
kHIScrollViewOptionsFillGrowArea	

## HISlider.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HISliderGetThumbOrientation	
HISliderGetTickMarkCount	
HISliderSetThumbOrientation	
HISliderSetTickMarkCount	

## HITextLengthFilter.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kEventParamTextLength	
-----------------------	--

## HITextViews.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlEditTextSpellCheckAsYouTypeTag	
kControlEditTextSpellCheckingTag	
kHITextViewClassID	The class ID for the HITextView class.

## HITheme.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIThemeBeginFocus	Para
HIThemeEndFocus	Para
HIThemeGetTextColorForThemeBrush	Para
HIThemeGetUIFontType	Para

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIThemeFocusRing	
kHIThemeFocusRingAbove	
kHIThemeFocusRingBelow	
kHIThemeFocusRingOnly	
kHIThemeTextBoxOptionEngraved	
kHIThemeTextHorizontalFlushDefault	
kHIThemeTextInfoVersionOne	
kHIThemeTextTruncationDefault	
kHIThemeTextVerticalFlushDefault	



## HIToolbar.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIToolbarItemCopyIconRef	
--------------------------	--

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kHICommandCycleToolbarModeLarger	
----------------------------------	--

kHICommandCycleToolbarModeSmaller	
-----------------------------------	--

kHICommandToggleAllToolbars	
-----------------------------	--

kHICommandToggleToolbar	
-------------------------	--

## HIToolboxDebugging.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

DebugFlashWindowUpdateRgn	
---------------------------	--

DebugFlashWindowVisRgn	
------------------------	--

DebugPrintControl	
-------------------	--

DebugPrintControlHierarchy	
----------------------------	--

DebugPrintDialogInfo	
----------------------	--

DebugPrintEvent	
-----------------	--

DebugPrintEventQueue	
----------------------	--

DebugPrintMainEventQueue	
--------------------------	--

DebugPrintMenu	
----------------	--

DebugPrintMenuItem	
--------------------	--

DebugPrintMenuList	
--------------------	--

DebugPrintTracedEvents	
DebugPrintWindow	
DebugPrintWindowList	
DebugSetEventTraceEnabled	
DebugTraceEvent	
DebugTraceEventByName	

## HIView.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIViewAdvanceFocusWithOptions	
HIViewCopyImageContentWithSize	
HIViewDrawNSImage	
HIViewGetFocus	
HIViewGetThemeTextInfo	
HIViewReleaseImageContent	
HIViewRetainImageContent	
HIViewSetFocus	
HIViewSetImageContent	
HIViewSetSubviewsNeedDisplayInShape	
HIViewSetTextFont	
HIViewSetTextHorizontalFlush	
HIViewSetTextTruncation	
HIViewSetTextVerticalFlush	
HIViewSetUpTextColor	
HIViewShowsFocus	
HIViewTrackMouseLocation	
HIViewTrackMouseShape	

**Data Types & Constants**

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HTypeAndCreator	
iconTypeAndCreator	
imageFile	
imageResource	
kHIViewContentAlertIconType	
kHIViewContentIconTypeAndCreator	
kHIViewContentImageFile	
kHIViewContentImageResource	
kHIViewContentNSImage	
kHIViewFocusOnAnyControl	
kHIViewFocusTraditionally	
kHIViewFocusWithoutWrapping	
nsImage	

**HIWindowViews.h**

---

**Data Types & Constants**

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kHIViewWindowCloseBoxID	
kHIViewWindowCollapseBoxID	
kHIViewWindowTitleID	
kHIViewWindowToolbarButtonID	
kHIViewWindowToolbarID	
kHIViewWindowZoomBoxID	
kHIWindowDragPart	Identifies the draggable part of a window frame view.

kHIWindowTitleBarPart	Identifies the title bar part of a window frame view.
kHIWindowTitleProxyIconPart	Identifies the proxy icon part of a window frame title view.

## IMKInputSession.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

IMKLocationToOffsetMappingMode	List modes for mapping a screen location to a text offset.
IMKTextOrientationName	The key used to find the client text orientation value, which is part of the dictionary returned by the method <code>attributesForCharacterIndex: lineHeightRectangle:</code> .
kIMKMouseTrackingMode	Mouse tracking mode.
kIMKNearestBoundaryMode	Nearest boundary mode.

## MacApplication.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIApplicationCreateDockTileContext	Returns a Quartz graphics context for drawing in the application Dock tile.
HIApplicationGetFocus	Returns either the modeless or effective focused window.
HIDictionaryWindowShow	Displays dictionary search result in a dictionary window.

## MacTextEditor.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

TXNShowOffset	
---------------	--

## MacWindows.h

## Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIWindowChangeAttributes	Changes the attributes of a window.
HIWindowConstrain	Moves and resizes a window to be within a specified bounding rectangle.
HIWindowCopyAvailablePositioningShape	Copies the available window positioning shape on a display.
HIWindowCopyDrawers	Obtains an array of the drawers that are attached to a window.
HIWindowCopyShape	Retrieves a shape that describes a region of a window.
HIWindowCreate	Creates a standard or custom window.
HIWindowCreateCollapsedDockTileContext	Creates a Quartz graphics context for drawing a collapsed window's Dock tile.
HIWindowFindAtLocation	Finds a window in the current process at a specified location.
HIWindowFromCGWindowID	Returns the window in the current process with a specified Quartz window ID.
HIWindowGetAvailablePositioningBounds	Gets the available window positioning bounds on a display.
HIWindowGetBackingLocation	
HIWindowGetBounds	Gets the bounds of a specified region of a window.
HIWindowGetCGWindowID	Returns the Quartz window ID assigned to a window.
HIWindowGetContentBorderThickness	
HIWindowGetGreatestAreaDisplay	Finds the display with the greatest area of intersection with a window region.
HIWindowGetIdealUserState	Gets the bounds of a window's content region in its user state.
HIWindowGetSharingType	
HIWindowGetThemeBackground	Gets the theme background brush for a window.
HIWindowIsAttributeAvailable	Returns a Boolean value indicating whether a window attribute is valid for a specified window class.

<code>HIWindowIsInStandardState</code>	Returns a Boolean value indicating whether a window is zoomed out to its standard state.
<code>HIWindowReleaseCollapsedDockTileContext</code>	Releases a Quartz graphics context for drawing a collapsed window's Dock tile.
<code>HIWindowSetBackingLocation</code>	
<code>HIWindowSetBounds</code>	Sets the bounds of a window based on either the structure or content region.
<code>HIWindowSetContentBorderThickness</code>	
<code>HIWindowSetIdealUserState</code>	Sets the bounds of a window's content region in its user state.
<code>HIWindowSetSharingType</code>	
<code>HIWindowSetToolbarView</code>	Sets a custom toolbar view for a window.
<code>HIWindowShowsFocus</code>	Returns a Boolean value indicating whether a window's content should show focus indicators such as focus rings.
<code>HIWindowTestAttribute</code>	Returns a Boolean value indicating whether a window has a specified attribute.
<code>HIWindowTrackProxyDrag</code>	Tracks the drag of a window proxy icon.

### Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HIContentBorderMetrics</code>	
<code>HIWindowBackingLocation</code>	
<code>HIWindowSharingType</code>	
<code>kHIToolbarViewDrawBackgroundTag</code>	A <code>SetControlData</code> tag that is used by the standard window frame view to inform the toolbar view whether the view should draw its background or leave its background transparent. The data for this tag is a Boolean. If the data value is true, the toolbar view should draw its background as it desires. If the data value is false, the toolbar view should leave its background transparent so that the window's root view can show through the toolbar view. Currently, the toolbar view will be asked to leave its background transparent for windows with the textured or unified appearance.
<code>kHIWindowBackingLocationDefault</code>	

## 10.5 Symbol Changes

<code>kHIWindowBackingLocationMainMemory</code>	
<code>kHIWindowBackingLocationVideoMemory</code>	
<code>kHIWindowBitAsyncDrag</code>	The window server drags the window automatically.
<code>kHIWindowBitAutoViewDragTracking</code>	The window automatically installs Drag Manager callbacks to detect drag actions, and automatically sends HView drag Carbon events.
<code>kHIWindowBitCanBeVisibleWithoutLogin</code>	The window can be made visible prior to user login.
<code>kHIWindowBitCloseBox</code>	The window has a close box.
<code>kHIWindowBitCollapseBox</code>	The window has a collapse box.
<code>kHIWindowBitCompositing</code>	The window uses HView-based compositing, which means that the entire window is comprised of HViews, and can be treated thusly.
<code>kHIWindowBitDoesNotCycle</code>	The window does not participate in window cycling invoked by Command-~ or keyboard shortcuts defined in the Keyboard & Mouse preference pane.
<code>kHIWindowBitDoesNotHide</code>	The window does not hide when the application is hidden.
<code>kHIWindowBitFrameworkScaled</code>	The window's content is scaled to match the display scale factor.
<code>kHIWindowBitHideOnFullScreen</code>	The window is automatically hidden during full-screen mode (when the menubar is invisible) and shown afterwards.
<code>kHIWindowBitHideOnSuspend</code>	The window is hidden automatically on suspend and shown on resume.
<code>kHIWindowBitIgnoreClicks</code>	The window never receives mouse events, even in areas that are opaque.
<code>kHIWindowBitInWindowMenu</code>	The window title appears in the system-generated Window menu.
<code>kHIWindowBitLiveResize</code>	The window supports live resizing.
<code>kHIWindowBitNoActivates</code>	The window does not receive activate events.
<code>kHIWindowBitNoConstrain</code>	The window is not repositioned by the default <code>kEventWindowConstrain</code> handler in response to changes in monitor size, Dock position, and so on.
<code>kHIWindowBitNoShadow</code>	The window has no shadow.
<code>kHIWindowBitNoTexturedContentSeparator</code>	Indicates that no border is drawn between the toolbar and window content.

<code>kHIWindowBitNoTitleBar</code>	The window's title bar can be hidden.
<code>kHIWindowBitNoUpdates</code>	The window does not receive update events.
<code>kHIWindowBitOpaqueForEvents</code>	The window receives mouse events even for areas of the window that are transparent (that is, have an alpha channel component of zero).
<code>kHIWindowBitResizable</code>	The window has a resize tab or box and is resizable.
<code>kHIWindowBitRoundBottomBarCorners</code>	
<code>kHIWindowBitSideTitlebar</code>	The window has a vertical title bar on the side of the window.
<code>kHIWindowBitStandardHandler</code>	The window supports the standard window event handler.
<code>kHIWindowBitTextured</code>	The window uses the textured or brushed-metal appearance.
<code>kHIWindowBitTexturedSquareCorners</code>	Indicates that a textured window should have square corners.
<code>kHIWindowBitToolbarButton</code>	The window has a toolbar button.
<code>kHIWindowBitUnifiedTitleAndToolbar</code>	The window draws its window title and toolbar using a unified appearance that has no separator between the two areas.
<code>kHIWindowBitZoomBox</code>	The window has a zoom box.
<code>kHIWindowCanJoinAllSpaces</code>	
<code>kHIWindowIgnoreObscuringWindows</code>	
<code>kHIWindowMenuCreator</code>	The property creator for accessing standard window menu item properties.
<code>kHIWindowMenuWindowTag</code>	The property tag for accessing standard window menu item properties that hold windows (values of type <code>WindowRef</code> ). Menu items with the <code>kHICommandSelectWindow</code> command ID will have a property with this tag that contains the window to be activated when that item is selected.
<code>kHIWindowMoveToActiveSpace</code>	
<code>kHIWindowSharingNone</code>	
<code>kHIWindowSharingReadOnly</code>	
<code>kHIWindowSharingReadWrite</code>	



<code>kHIWindowVisibleInAllSpaces</code>	If this bit is set, the window is visible in all Spaces workspaces. If this bit is not set, the window is only visible in the workspace in which it was created.
<code>kWindowCanBeVisibleWithoutLoginAttribute</code>	If the bit specified by this mask is set, the window can be made visible prior to user login.
<code>kWindowConstrainMoveMinimum</code>	Move the window the minimum amount necessary to be onscreen. This option is only supported by the function <code>HIWindowConstrain</code> . This option applies if a partial fit is not allowed ( <code>kWindowConstrainAllowPartial</code> is not specified) or a partial fit is allowed, but the window is not even partially visible. In either case, the window will be moved just enough to be slightly onscreen. You may customize the minimum amount that is required to be visible by passing the desired dimensions in the <code>inMinimumSize</code> parameter to <code>HIWindowConstrain</code> .
<code>kWindowConstrainUseSpecifiedBounds</code>	Use the specified bounds of the window region to be constrained. This option is only supported by the function <code>HIWindowConstrain</code> . The bounds are specified using the <code>ioBounds</code> parameter, allowing you to constrain a window to a hypothetical location. For example, if you plan to move your window such that its content region is at a certain location, and you want to know in advance before moving the window whether the window would be offscreen at that location, you can use this option.
<code>kWindowHasRoundBottomBarCornersAttribute</code>	
<code>kWindowOverlayProc</code>	
<code>kWindowTexturedSquareCornersAttribute</code>	See <code>kHIWindowBitTexturedSquareCorners</code> .
<code>kWindowUnifiedTitleAndToolbarAttribute</code>	If the bit specified by this mask is set, the window draws its window title and toolbar using a unified appearance that has no separator between the two areas.
<code>WindowDefUPP</code>	Defines a UPP to a specified window definition.

## Menus.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HIMenuCopyAttributedItemText</code>	
---	--

HIMenuCopyFont	
HIMenuCopyItemViewClass	
HIMenuCopyParents	
HIMenuGetItemView	
HIMenuItemViewCancelMenuTracking	
HIMenuItemViewGetEnclosingMenuItem	
HIMenuSetAttributedItemText	
HIMenuSetFont	
HIMenuSetItemViewClass	
HIMenuSetSelection	

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

attributedText	
kMenuItemDataAllDataVersionThree	
kMenuItemDataAttributedText	
kMenuItemDataFont	
MenuDefUPP	Defines a universal procedure pointer to a menu definition function.

### TextInputSources.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

TISCopyCurrentASCIICapableKeyboardInputSource	Copies a text input source reference for the most recently used ASCII-capable keyboard input source.
TISCopyCurrentASCIICapable-KeyboardLayoutInputSource	Copies a text input source reference for the most recently used ASCII-capable keyboard layout.

<code>TISCopyCurrentKeyboardInputSource</code>	Copies a text input source reference for the currently selected keyboard input source.
<code>TISCopyCurrentKeyboardLayoutInputSource</code>	Copies a text input source reference for the keyboard layout currently in use.
<code>TISCopyInputMethodKeyboardLayoutOverride</code>	Copies a text input source reference for the currently selected input method's keyboard layout override.
<code>TISCopyInputSourceForLanguage</code>	Copies a text input source reference for the input source that should be used to input the specified language.
<code>TISCreateASCIICapableInputSourceList</code>	Creates a list of ASCII-capable keyboard input sources.
<code>TISCreateInputSourceList</code>	Creates a list of input sources that match specified properties.
<code>TISDeselectInputSource</code>	Deselects the specified input source.
<code>TISDisableInputSource</code>	Disables the specified input source.
<code>TISEnableInputSource</code>	Enables the specified input source.
<code>TISGetInputSourceProperty</code>	Gets the value of a specified property for a specified input source.
<code>TISInputSourceGetTypeID</code>	Gets the <code>CTypeID</code> of a text input source reference.
<code>TISRegisterInputSource</code>	Registers any new input sources in a file or bundle so a text input source reference can be obtained immediately for each new input source.
<code>TISSelectInputSource</code>	Selects the specified input source.
<code>TISSetInputMethodKeyboardLayoutOverride</code>	Sets the keyboard layout override for an input method or mode.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kTISCategoryInkInputSource</code>	One input source category value associated with the property key <code>kTISPropertyInputSourceCategory</code> .
<code>kTISCategoryKeyboardInputSource</code>	One input source category value associated with the property key <code>kTISPropertyInputSourceCategory</code> .

<code>kTISCategoryPaletteInputSource</code>	One input source category value associated with the property key <code>kTISPropertyInputSourceCategory</code> .
<code>kTISNotifyEnabledKeyboardInputSourcesChanged</code>	The name of the CF distributed notification for a change to the set of enabled keyboard input sources.
<code>kTISNotifySelectedKeyboardInputSourceChanged</code>	The name of the CF distributed notification for a change to the selected keyboard input source.
<code>kTISPropertyBundleID</code>	This property key designates a property value, which is a <code>CFStringRef</code> for the reverse DNS BundleID associated with the input source.
<code>kTISPropertyIconImageURL</code>	This property key designates a property value, which is a <code>CFURLRef</code> indicating the file containing the image (typically TIFF) to be used as the input source icon.
<code>kTISPropertyIconRef</code>	This property key designates a property value, which is an <code>IconRef</code> value for the input source icon.
<code>kTISPropertyInputModeID</code>	This property key designates a property value, which is a <code>CFStringRef</code> for a reverse DNS string that identifies a particular usage class for input modes.
<code>kTISPropertyInputSourceCategory</code>	This property key designates a property value, which is a <code>CFStringRef</code> indicating the category of an input source.
<code>kTISPropertyInputSourceID</code>	This property key designates a property value, which is a <code>CFStringRef</code> for the unique reverse DNS name associated with the input source.
<code>kTISPropertyInputSourceIsASCIICapable</code>	This property key designates a property value, which is a <code>CFBooleanRef</code> indicating whether the input source is intended to be capable of ASCII input.
<code>kTISPropertyInputSourceIsEnableCapable</code>	This property key designates a property value which is a <code>CFBooleanRef</code> indicating whether the input source can ever be programmatically enabled using <code>TISEnableInputSource</code> .
<code>kTISPropertyInputSourceIsEnabled</code>	This property key designates a property value, which is a <code>CFBooleanRef</code> indicating whether the input source is currently enabled.

<code>kTISPropertyInputSourceIsSelectCapable</code>	This property key designates a property value, which is a <code>CFBooleanRef</code> indicating whether the input source can ever be programmatically selected using <code>TISSelectInputSource</code> .
<code>kTISPropertyInputSourceIsSelected</code>	This property key designates a property value, which is a <code>CFBooleanRef</code> indicating whether the input source is currently selected.
<code>kTISPropertyInputSourceLanguages</code>	This property key designates a property value, which is a <code>CFArrayRef</code> for an array of <code>CFStringRef</code> s, where each string is the language code for a language that can be input using the input source. You can use this constant with the <code>TISGetInputSourceProperty</code> function, but you may not use it in the filter dictionary passed to the <code>TISCreateInputSourceList</code> function.
<code>kTISPropertyInputSourceType</code>	This property key designates a property value, which is a <code>CFStringRef</code> indicating the specific type of an input source.
<code>kTISPropertyLocalizedName</code>	This property key designates a property value, which is a <code>CFStringRef</code> for the input source's localized name as intended for user interface use.
<code>kTISPropertyUnicodeKeyLayoutData</code>	This property key designates a property value, which is a <code>CFDataRef</code> that refers to the 'uchr' keyboard layout data for a keyboard layout input source.
<code>kTISTypeCharacterPalette</code>	One input source type value associated with the property key <code>kTISPropertyInputSourceType</code> .
<code>kTISTypeInk</code>	One input source type value associated with the property key <code>kTISPropertyInputSourceType</code> .
<code>kTISTypeKeyboardInputMethodModeEnabled</code>	One input source type value associated with the property key <code>kTISPropertyInputSourceType</code> .
<code>kTISTypeKeyboardInputMethodWithoutModes</code>	One input source type value associated with the property key <code>kTISPropertyInputSourceType</code> .
<code>kTISTypeKeyboardInputMode</code>	One input source type value associated with the property key <code>kTISPropertyInputSourceType</code> .
<code>kTISTypeKeyboardLayout</code>	One input source type value associated with the property key <code>kTISPropertyInputSourceType</code> .
<code>kTISTypeKeyboardViewer</code>	One input source type value associated with the property key <code>kTISPropertyInputSourceType</code> .

TISInputSourceRef	Defines a pointer to an opaque CF object that unambiguously (within a single process) represents a text input source.
-------------------	---

## TextServices.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kTSM24Version	
kTSMDocumentEnabledInputSourcesPropertyTag	
kTSMDocumentInputSourceOverridePropertyTag	

## Print

---

## PMAApplication.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

PMShowPageSetupDialogAsSheet	Displays a Page Setup dialog as a sheet attached to a window.
PMShowPrintDialogWithOptions	Displays a Print dialog with a specified set of controls.
PMShowPrintDialogWithOptionsAsSheet	Displays a Print dialog with a specified set of controls as a sheet attached to a window.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kPMHideInlineItems	Show nothing in the inline portion of the Print dialog.
kPMShowDefaultInlineItems	Show the default set of items (copies & page range) in the inline portion of the Print dialog.
kPMShowInlineCopies	Show the copies edit-text field, the collated check box, and the two-sided check box (if the printer supports it) in the top portion of the Print dialog.

<code>kPMSHOWInlineOrientation</code>	Show the page orientation buttons in the top portion of the Print dialog.
<code>kPMSHOWInlinePageRange</code>	Show the page range radio buttons and the from & to page edit-text fields in the top portion of the Print dialog.
<code>kPMSHOWInlinePageRangeWithSelection</code>	Show the page range radio buttons with the addition of a selection button and the From & To Page edit-text fields in the top portion of the Print dialog.
<code>kPMSHOWInlinePaperSize</code>	Show the formatting paper size popup menu in the top portion of the Print dialog.
<code>kPMSHOWInlineScale</code>	Show the scale edit-text field in the top portion of the Print dialog.
<code>kPMSHOWPageAttributesPDE</code>	Add a separate pane to the Print dialog that contains the Page Setup dialog information (paper size, orientation, and scale).
<code>PMPrintDialogOptionFlags</code>	Flags that specify items to show in the expanded Print dialog in Mac OS X v10.5 and later.

## ImageCapture

---

### ICAApplication.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>ICACopyObjectData</code>	
<code>ICARegisterForEventNotification</code>	
<code>ICASendNotification</code>	
<code>ICASendNotificationAndWaitForReply</code>	
<code>ICAShowDeviceBrowser</code>	

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>commandCode</code>	
<code>containerType</code>	

## 10.5 Symbol Changes

dataUsageMode	
eventCode	
ICAError	
ICASessionID	
kICABonjourServiceNameKey	
kICABonjourServiceTypeKey	
kICABonjourTXTRecordKey	
kICACameraPassThruNotUsed	
kICACameraPassThruReceive	
kICACameraPassThruSend	
kICACreationDateStringKey	
kICADataPropertyKey	
kICADataSizeKey	
kICADataTypeKey	
kICADeviceBrowserDeviceRefKey	
kICADeviceCapabilitiesKey	
kICADeviceIconPathKey	
kICADevicePropArtist	
kICADevicePropBatteryLevel	
kICADevicePropBurstInterval	
kICADevicePropBurstNumber	
kICADevicePropCaptureDelay	
kICADevicePropCompressionSetting	
kICADevicePropContrast	
kICADevicePropCopyrightInfo	
kICADevicePropDateTime	
kICADevicePropDigitalZoom	
kICADevicePropEffectMode	



## 10.5 Symbol Changes

kICADevicePropExposureBiasCompensation	
kICADevicePropExposureIndex	
kICADevicePropExposureMeteringMode	
kICADevicePropExposureProgramMode	
kICADevicePropExposureTime	
kICADevicePropFlashMode	
kICADevicePropFNumber	
kICADevicePropFocalLength	
kICADevicePropFocusDistance	
kICADevicePropFocusMeteringMode	
kICADevicePropFocusMode	
kICADevicePropFunctionalMode	
kICADevicePropImageSize	
kICADevicePropIRGBGain	
kICADevicePropSharpness	
kICADevicePropStillCaptureMode	
kICADevicePropTimelapseInterval	
kICADevicePropTimelapseNumber	
kICADevicePropUndefined	
kICADevicePropUploadURL	
kICADevicePropWhiteBalance	
kICADevicesArrayKey	
kICADeviceSharedKey	
kICADeviceTypeCamera	
kICADeviceTypeKey	
kICADeviceTypeScanner	
kICADeviceUnsupportedErr	
kICADeviceUsedKey	

## 10.5 Symbol Changes

kICADeviceWebSharedKey	
kICAErrorKey	
kICAEventClassPTPStandard	
kICAEventClassPTPVendor	
kICAExecutableArchitectureKey	
kICAExtendedNotificationPB	
kICAExtensionInternalErr	
kICAFrameworkInternalErr	
kICAInvalidSessionErr	
kICALockStatusKey	
kICAMediaDurationInSecondsKey	
kICAMediaHeightKey	
kICAMediaWidthKey	
kICAMessageCameraPassThrough	
kICAMessageGetEventData	
kICAModificationDateStringKey	
kICANotificationClassKey	
kICANotificationClassProprietary	
kICANotificationClassPTPStandard	
kICANotificationClassPTPVendor	
kICANotificationDataCookieKey	
kICANotificationDataIsBigEndianKey	
kICANotificationDataKey	
kICANotificationDataSizeKey	
kICANotificationDeviceICAObjectKey	
kICANotificationDeviceListICAObjectKey	
kICANotificationICAObjectKey	
kICANotificationImageBytesPerRowKey	

## 10.5 Symbol Changes

kICANotificationImageDataKey	
kICANotificationImageDataSizeKey	
kICANotificationImageHeightKey	
kICANotificationImageKey	
kICANotificationImageNumberOfRowsKey	
kICANotificationImageStartRowKey	
kICANotificationImageWidthKey	
kICANotificationNumberOfImagesRemainingKey	
kICANotificationPercentDownloadedKey	
kICANotificationRawEventKey	
kICANotificationScannerButtonTypeKey	
kICANotificationScannerDocumentNameKey	
kICANotificationTypeCaptureComplete	
kICANotificationTypeDeviceAdded	
kICANotificationTypeDeviceConnectionProgress	
kICANotificationTypeDeviceInfoChanged	
kICANotificationTypeDevicePropertyChanged	
kICANotificationTypeDeviceRemoved	
kICANotificationTypeDeviceWasReset	
kICANotificationTypeDownloadProgressStatus	
kICANotificationTypeKey	
kICANotificationTypeObjectAdded	
kICANotificationTypeObjectInfoChanged	
kICANotificationTypeObjectRemoved	
kICANotificationTypeProprietary	
kICANotificationTypeRequestObjectTransfer	
kICANotificationTypeScannerButtonPressed	
kICANotificationTypeScannerPageDone	

## 10.5 Symbol Changes

kICANotificationTypeScannerScanDone	
kICANotificationTypeScannerSessionClosed	
kICANotificationTypeScanProgressStatus	
kICANotificationTypeStoreAdded	
kICANotificationTypeStoreFull	
kICANotificationTypeStoreInfoChanged	
kICANotificationTypeStoreRemoved	
kICANotificationTypeTransactionCanceled	
kICANotificationTypeUnreportedStatus	
kICAObjectKey	
kICAObjectNameKey	
kICARawKey	
kICARefconKey	
kICARemoteDeviceKey	
kICAThumbnailFormatPNG	
kICAThumbnailPropertyKey	
kICAThumbnailSizeKey	
kICATypeFloat	
kICATypeUInt8	
kICAUSBProductIDKey	
kICAUSBVendorIDKey	
kMetaDataDictionaryKey	
notificationDictionary	
notificationDictionary	
numOfInputParams	
numOfOutputParams	
objectOfInterest	
replyCode	

resultCode	
transactionID	

## ICACamera.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kICACapabilityCanCameraCaptureNewImage	
kICACapabilityCanCameraDeleteAll	
kICACapabilityCanCameraDeleteOne	
kICACapabilityCanCameraSyncClock	
kICACapabilityCanCameraUploadData	
kICACapabilityMayStoreNewImagesInTempStore	
kICAFilesystemUndefined	
kICAPropertyCameraArtist	
kICAPropertyCameraCopyrightInfo	
kICAPropertyCameraUploadURL	
kICASTorageUndefined	

## NavigationServices

---

### Navigation.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kNavClientPopupExtensionTag	
-----------------------------	--



# 10.4 Symbol Changes

---

This article lists the symbols added to `Carbon.framework` in Mac OS X v10.4.

## Classes

All of the classes with new symbols are listed alphabetically, with their new class, instance, and delegate methods described.

### NSObject

---

Complete reference information is available in the `NSObject` reference.

#### Instance Methods

---

<code>initWithBundle:</code>	Initializes the plug-in.
<code>jobTemplate</code>	Returns the job template for the current printer.
<code>pageFormat</code>	Returns the current page format object.
<code>panelKind</code>	Returns the kind ID of the pane.
<code>panelName</code>	Returns the localized name of the pane.
<code>panelView</code>	Returns a view that contains the pane's user interface.
<code>panelViewDidResize</code>	Notifies the printing system that the size of the pane's view has changed.
<code>PDEPanelsForType:withHostInfo:</code>	Returns an array of panes to be added to a printing dialog.
<code>PMPrinter</code>	Returns the current printer object.
<code>ppdFile</code>	Returns a CUPS object used to access information about the current printer.
<code>PPDOptionKeyValueDidChange:ppdChoice:</code>	Handles a change in the value of a PPD option that the pane supports.
<code>printerInfoTicket</code>	Returns the printer information ticket for the current printer.

<code>printSession</code>	Returns the current print session object.
<code>printSettings</code>	Returns the current print settings object.
<code>restoreValuesAndReturnError:</code>	Restores the pane's settings.
<code>saveValuesAndReturnError:</code>	Verifies that the pane's settings are valid and have been saved.
<code>shouldHide</code>	Prepares the pane to be hidden.
<code>summaryInfo</code>	Returns localized text descriptions of the pane's current settings.
<code>supportedPPDOptionKeys</code>	Declares that the pane supports one or more PPD options.
<code>willChangePPDOptionKeyValue:ppdChoice:</code>	Requests a change in the value of a PPD option.
<code>willShow</code>	Prepares the pane for display.

## C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

### OpenScripting

---

#### ASDebugging.h

---

##### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>OSACopyScriptingDefinition</code>	Creates a copy of a scripting definition (sdef) from the specified file or bundle.
---	--

#### ASRegistry.h

---

##### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kASNumericStrings</code>	Should strings be considered as numbers?
--------------------------------	--



<code>kASNumericStringsConsiderMask</code>	If bit at this position is set, consider strings to represent numerical values for comparison. For example, compare the string "1.01" as if it were the number 1.01.
<code>kASNumericStringsIgnoreMask</code>	If bit at this position is set, do not compare strings as numeric values.

## HI Toolbox

---

### Appearance.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kThemeAdornmentHeaderButtonNoSortArrow</code>	
<code>kThemeBevelButtonInset</code>	
<code>kThemeBevelButtonLarge</code>	
<code>kThemeBevelButtonMedium</code>	
<code>kThemeBevelButtonSmall</code>	
<code>kThemeBrushListViewColumnDivider</code>	
<code>kThemeBrushListViewEvenRowBackground</code>	
<code>kThemeBrushListViewOddRowBackground</code>	
<code>kThemeCheckBoxMini</code>	
<code>kThemeCheckBoxSmall</code>	
<code>kThemeDisclosureTriangle</code>	
<code>kThemeIndeterminateBarLarge</code>	
<code>kThemeIndeterminateBarMedium</code>	
<code>kThemeIndeterminateBarMini</code>	
<code>kThemeMetricHSliderTickOffset</code>	The horizontal start offset for the first tick mark on a horizontal slider.
<code>kThemeMetricScrollBarMinThumbHeight</code>	The minimum height for a thumb on a scroll bar.

## 10.4 Symbol Changes

<code>kThemeMetricScrollBarMinThumbWidth</code>	The minimum width for a thumb on a scroll bar.
<code>kThemeMetricSliderMinThumbHeight</code>	The minimum height for a thumb on a slider.
<code>kThemeMetricSliderMinThumbWidth</code>	The minimum width for a thumb on a slider.
<code>kThemeMetricSmallScrollBarMinThumbHeight</code>	The minimum height for a thumb on a small scroll bar.
<code>kThemeMetricSmallScrollBarMinThumbWidth</code>	The minimum width for a thumb on a small scroll bar.
<code>kThemeMetricVSliderTickOffset</code>	The vertical start offset for the first tick mark on a vertical slider.
<code>kThemeProgressBarLarge</code>	
<code>kThemeProgressBarMedium</code>	
<code>kThemeProgressBarMini</code>	
<code>kThemePushButtonInset</code>	
<code>kThemePushButtonInsetSmall</code>	
<code>kThemeRadioButtonMini</code>	
<code>kThemeRadioButtonSmall</code>	
<code>kThemeRoundButtonHelp</code>	
<code>kThemeRoundButtonLarge</code>	
<code>kThemeScrollBarMedium</code>	
<code>kThemeScrollBarMini</code>	
<code>kThemeScrollBarSmall</code>	
<code>kThemeSliderMedium</code>	
<code>kThemeSliderMini</code>	
<code>kThemeSliderSmall</code>	
<code>kThemeWindowHasToolbarButton</code>	

## CarbonEvents.h

## Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CopyServicesMenuCommandKeys	Obtains information about command key shortcuts in an application's Services menu.
GetSymbolicHotKeyMode	Obtains the current hot key mode.
HICopyAccessibilityActionDescription	Returns the system-defined action description string for a standard action.
HICopyAccessibilityRoleDescription	Returns the system-defined role description string for a standard role or role-subrole pair.
PopSymbolicHotKeyMode	Removes a hot key mode request from the hot key mode stack.
PushSymbolicHotKeyMode	Sets a new mode for enabling and disabling symbolic hot keys.

## Data Types &amp; Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIModalClickResult	Define constants that describe responses to the kEventWindowGetModalityClick event.
HIViewInstallEventHandler	
kEventAppFocusDrawer	Request to shift keyboard focus to the drawer in the focused window.
kEventParamControlPrefersShape	typeBoolean
kEventParamModalClickResult	On exit, a value indicating how the click should be handled. For details, see "Modal Window Click Constants."
kEventParamModalWindow	On exit, the modal window that caused the click to be blocked, if any.
kEventParamShape	typeHIShapeRef
kEventParamTSMDocAccessLineBounds	typeCFMutableArrayRef

<code>kEventParamWindowModality</code>	On exit, the modality of the modal window that is in front of the clicked window, if any. This parameter is required only if the <code>kEventParamModalClickResult</code> parameter contains <code>kHIModalClickIsModal</code> .
<code>kEventTextInputFilterText</code>	Sent before any final-form text is sent to the user focus.
<code>kEventWindowFocusDrawer</code>	Focus should be shifted to the drawer of a window.
<code>kEventWindowGetClickModality</code>	Sent to a window by the event dispatcher target before dispatching a mouse-down or mouse-up event to the clicked window.
<code>kEventWindowHandleActivate</code>	The window has received a <code>kEventWindowActivated</code> event, and its contents should become active.
<code>kEventWindowHandleDeactivate</code>	The window has received a <code>kEventWindowDeactivated</code> event, and its contents should become inactive.
<code>kEventWindowSheetClosed</code>	A sheet has closed.
<code>kEventWindowSheetClosing</code>	A sheet is closing.
<code>kEventWindowSheetOpened</code>	A sheet has opened.
<code>kEventWindowSheetOpening</code>	A sheet is opening.
<code>kHICommandChangeSpelling</code>	Change the spelling. Events with this command ID are only generated in Mac OS X v10.4 and later. If spell checking has been enabled in MLTE or an <code>HITextView</code> , this command is handled automatically.
<code>kHICommandCheckSpelling</code>	Spell check the document now. Events with this command ID are only generated in Mac OS X v10.4 and later. If spell checking has been enabled in MLTE or an <code>HITextView</code> , this command is handled automatically.
<code>kHICommandCheckSpellingAsYouType</code>	Begin interactive spell checking. Events with this command ID are only generated in Mac OS X v10.4 and later. If spell checking has been enabled in MLTE or an <code>HITextView</code> , this command is handled automatically.

<code>kHICommandIgnoreSpelling</code>	Ignore this word while spell checking this text view. Events with this command ID are only generated in Mac OS X v10.4 and later. If spell checking has been enabled in MLTE or an <code>HITextView</code> , this command is handled automatically.
<code>kHICommandLearnWord</code>	Learn this spelling for all documents. Events with this command ID are generated only in Mac OS X v10.4 and later. If spell checking has been enabled in MLTE or an <code>HITextView</code> , this command is handled automatically.
<code>kHICommandShowSpellingPanel</code>	Display the spelling panel if it is not already visible.
<code>kHIHotKeyModeAllDisabled</code>	All hot keys are disabled.
<code>kHIHotKeyModeAllDisabledExceptUniversalAccess</code>	All hot keys are disabled except for the Universal Access hot keys (that is, zooming, white-on-black, and enhanced contrast).
<code>kHIHotKeyModeAllEnabled</code>	All hot keys are enabled.
<code>kHIModalClickAllowEvent</code>	If <code>kHIModalClickIsModal</code> is set, the <code>kHIModalClickAllowEvent</code> flag indicates whether the click event should be allowed to pass to the clicked window. If <code>kHIModalClickIsModal</code> is not set, the setting of <code>kHIModalClickAllowEvent</code> is ignored.
<code>kHIModalClickAnnounce</code>	If <code>kHIModalClickIsModal</code> is set and <code>kHIModalClickAllowEvent</code> is not set, <code>kHIModalClickAnnounce</code> indicates whether the caller should announce that the click has been blocked by a modal window using the appropriate UI (typically, by calling <code>SysBeep</code> ).
<code>kHIModalClickIsModal</code>	A modal window prevents the mouse event from being passed to the clicked window.
<code>kHIModalClickRaiseWindow</code>	If <code>kHIModalClickIsModal</code> and <code>kHIModalClickAllowEvent</code> are set, <code>kHIModalClickRaiseWindow</code> indicates whether the clicked window should be z-ordered to the top of its window group.
<code>kHIServicesMenuCharCode</code>	The character code of the menu item shortcut.
<code>kHIServicesMenuItemName</code>	The name of the menu item.
<code>kHIServicesMenuKeyModifiers</code>	The keyboard modifiers of the menu item shortcut in Menu Manager modifiers format.

<code>kHIServicesMenuProviderName</code>	The name of the service provider.
<code>typeHIShapeRef</code>	<code>HIShapeRef</code>
<code>typeModalClickResult</code>	<code>HIModalClickResult</code>
<code>typeWindowModality</code>	<code>WindowModality</code>

## ControlDefinitions.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>AXUIElementCreateWithDataBrowserAndItemInfo</code>	Creates an <code>AXUIElementRef</code> that represents some part of a data browser accessibility hierarchy.
<code>AXUIElementGetDataBrowserItemInfo</code>	Obtains a description of the part of a data browser represented by an <code>AXUIElementRef</code> .
<code>DataBrowserChangeAttributes</code>	Sets the attributes for a data browser.
<code>DataBrowserGetAttributes</code>	Gets the attributes of a data browser.
<code>DataBrowserGetMetric</code>	Gets the value of a specified data browser metric.
<code>DataBrowserSetMetric</code>	Sets the value of a specified data browser metric.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>columnIndex</code>	
<code>columnProperty</code>	
<code>DataBrowserAccessibilityItemInfo</code>	
<code>DataBrowserAccessibilityItemInfoV0</code>	
<code>DataBrowserAccessibilityItemInfoV1</code>	
<code>DataBrowserMetric</code>	Specifies constants used by <code>DataBrowserSetMetric</code> .
<code>kControlEditTextCharCount</code>	

<code>kDataBrowserAttributeColumnViewResizeWindow</code>	In column view, this data browser is allowed to resize the owning window whenever necessary.
<code>kDataBrowserAttributeListViewAlternatingRowColors</code>	In list view, this data browser should draw alternating row background colors. However, note that this attribute does not work with variable row heights as of Mac OS X v10.4.
<code>kDataBrowserAttributeListViewDrawColumnDividers</code>	In list view, this data browser should draw a vertical line between the columns.
<code>kDataBrowserAttributeNone</code>	The data browser has no attributes.
<code>kDataBrowserListViewNoGapForIconInHeaderButton</code>	Normally the text in a header button for a column of type <code>kDataBrowserIconAndTextType</code> is aligned as though it has an icon next to it even if no icon is specified for the header button.
<code>kDataBrowserMetricCellContentInset</code>	The content (icon, text, etc.) within a cell is drawn a certain amount in from the left and right edges of the cell. This metric governs the amount of inset.
<code>kDataBrowserMetricDisclosureColumnEdgeInset</code>	In list view only, this metric is used instead of (not in addition to) <code>DataBrowserMetricCellContentInset</code> for the side of the cell in the disclosure column that displays the disclosure triangle.
<code>kDataBrowserMetricDisclosureColumnPerDepthGap</code>	In list view only, this metric controls the amount of space in the disclosure column for each level of indentation in progressively deeper hierarchies of disclosed items.
<code>kDataBrowserMetricDisclosureTriangleAndContentGap</code>	In list view only, this metric controls the amount of space between the disclosure triangle and the cell's content.
<code>kDataBrowserMetricIconAndTextGap</code>	This metric controls the space between the icon and text within a column of type <code>kDataBrowserIconAndTextType</code> .
<code>kDataBrowserMetricLast</code>	Same as <code>kDataBrowserMetricDisclosureColumnPerDepthGap</code> .

<code>kDataBrowserPopupMenuButtonless</code>	This flag is only for use with columns of type <code>kDataBrowserPopupMenuType</code> and indicates that the popup is to be drawn in a sleek buttonless fashion. The text is drawn next to a popup glyph, and the whole cell is clickable.
<code>kHIDataBrowserClassID</code>	Defines the <code>HIObjecT</code> class ID for the <code>HIDataBrowser</code> class.
<code>kHIUserPaneClassID</code>	Defines the <code>HIObjecT</code> class ID for the <code>HIUserPane</code> class.
<code>propertyPart</code>	
<code>rowIndex</code>	
<code>v0</code>	

## Dialogs.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kStdAlertDoNotCloseOnHelp</code>	Specifies that the alert stay up even after the user clicks the Help button.
--	--

## Drag.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kDragDoNotScaleImage</code>	
-----------------------------------	--

## HIArchive.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HIArchiveCopyDecodedCFTYPE</code>	Retrieves a <code>CFTYPE</code> object from an archive.
<code>HIArchiveCopyEncodedData</code>	Compresses an archive for storage.



<code>HIArchiveCreateForDecoding</code>	Creates an <code>HIArchive</code> object to retrieve objects.
<code>HIArchiveCreateForEncoding</code>	Creates an <code>HIArchive</code> object to store objects.
<code>HIArchiveDecodeBoolean</code>	Retrieves a Boolean value from an archive.
<code>HIArchiveDecodeNumber</code>	Retrieves a number from an archive.
<code>HIArchiveEncodeBoolean</code>	Stores a Boolean value in an archive.
<code>HIArchiveEncodeCftype</code>	Stores a <code>Cftype</code> object in an archive.
<code>HIArchiveEncodeNumber</code>	Stores a number in an archive.
<code>HIArchiveGetTypeID</code>	Obtains the <code>Cftype</code> ID for <code>HIArchive</code> objects.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>hiArchiveEncodingCompleteErr</code>	<code>HIArchiveCopyEncodedData</code> was called on this archive, so no more data can be added.
<code>hiArchiveHIObjectIgnoresArchivingErr</code>	The <code>HIObject</code> you wanted to encode does not support the <code>HIArchive</code> protocol.
<code>hiArchiveKeyNotAvailableErr</code>	The requested key does not exist in the specified archive.
<code>HIArchiveRef</code>	Defines an uncompressed archive object.
<code>hiArchiveTypeMismatchErr</code>	The encoding or decoding archive was passed into a noncorresponding function. (For example, an archive created for encoding was passed into a decoding function.)
<code>kHIArchiveDecodeSuperclassForUnregisteredObjects</code>	If the class of the <code>HIObject</code> you are attempting to decode is not a registered subclass, this option allows <code>HIArchiveCopyDecodedCftype</code> to instantiate the object as its superclass, if it exists.

## HIGeometry.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIGetScaleFactor	Returns the scale factor of an application's user interface.
HIPointConvert	Converts a point from one coordinate space to another.
HIRectConvert	Converts a rectangle from one coordinate space to another.
HISizeConvert	Converts a size structure from one coordinate space to another.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HICoordinateSpace	Specify coordinate spaces used in HIToolbox.
kHICoordSpace72DPIGlobal	Specifies a global coordinate space that has been adjusted by the scale factor of your application's user interface.
kHICoordSpaceScreenPixel	Specifies a coordinate space defined by the screen size in pixels.
kHICoordSpaceView	Specifies the coordinate space of a view of type <code>HViewRef</code> .
kHICoordSpaceWindow	Specifies the coordinate space of a window of type <code>WindowRef</code> .

### HIOBJECT.H

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIObjectCopyCustomArchiveData	Copies custom archive data that is associated with an <code>HIObject</code> .
HIObjectIsArchivingIgnored	Obtains a Boolean value indicating whether an <code>HIObject</code> is marked as ignored for archiving.
HIObjectOverrideAccessibilityContainment	Overrides the <code>AXUIElement</code> references supplied by an <code>HIObject</code> .
HIObjectSetArchivingIgnored	Changes the state of archiving for an <code>HIObject</code> .
HIObjectSetAuxiliaryAccessibilityAttribute	Associates an additional accessibility attribute with an <code>HIObject</code> .
HIObjectSetCustomArchiveData	Associates custom archive data with an <code>HIObject</code> .

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kEventHIOBJECTEncode	HIArchiveEncodeHIOBJECT has been called on your HIOBJECT, and you are being asked to encode your object into an archive.
kEventParamHIArchive	An HIArchive used to store or retrieve the HIOBJECT.
kHIOBJECTCustomDataCDEFProcIDKey	The standard custom archive data dictionary key for ProcPointer-based CDEFs.
kHIOBJECTCustomDataClassIDKey	The class ID.
kHIOBJECTCustomDataParameterNamesKey	The value of this key is an array of strings. Each CFStringRef contains an OSType that is a Carbon event parameter name.
kHIOBJECTCustomDataParameterTypesKey	The value of this key is an array of strings. Each CFStringRef contains an OSType that is a Carbon event parameter type.
kHIOBJECTCustomDataParameterValuesKey	The value of this key is an array of strings. Each CFStringRef contains a representation of the value.
kHIOBJECTCustomDataSuperClassIDKey	The super-class ID key.

## HIShape.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIShapeCreateEmpty	Creates an empty immutable shape.
HIShapeIntersectsRect	Determines whether a given shape intersects a given rectangle.

## HITheme.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIThemeBrushCreateCGColor	Para
HIThemeDrawSegment	
HIThemeSetFill	Para
HIThemeSetStroke	Para
HIThemeSetTextFill	Para

**Data Types & Constants**

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIThemeSegmentAdornment	
HIThemeSegmentDrawInfo	
HIThemeSegmentDrawInfoPtr	
HIThemeSegmentKind	
HIThemeSegmentPosition	
HIThemeSegmentSize	
HIThemeTabDrawInfoVersionZero	
HIThemeTabKind	
HIThemeTabPaneAdornment	
HIThemeTabPaneDrawInfoVersionZero	
HIThemeTabPosition	
kHIThemeSegmentAdornmentFocus	
kHIThemeSegmentAdornmentLeadingSeparator	
kHIThemeSegmentAdornmentNone	
kHIThemeSegmentAdornmentTrailingSeparator	
kHIThemeSegmentKindInset	
kHIThemeSegmentKindNormal	
kHIThemeSegmentPositionFirst	
kHIThemeSegmentPositionLast	
kHIThemeSegmentPositionMiddle	
kHIThemeSegmentPositionOnly	
kHIThemeSegmentSizeMini	
kHIThemeSegmentSizeNormal	
kHIThemeSegmentSizeSmall	
kHIThemeTabAdornmentLeadingSeparator	
kHIThemeTabAdornmentTrailingSeparator	

kHIThemeTabKindNormal	
kHIThemeTabPaneAdornmentNormal	
kHIThemeTabPositionFirst	
kHIThemeTabPositionLast	
kHIThemeTabPositionMiddle	
kHIThemeTabPositionOnly	

## HIToolbar.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIToolbarGetSelectedItemInWindow	Obtains the toolbar item that is selected in a window.
HIToolbarItemChangeAttributesInWindow	Changes the attributes of a toolbar item in a specific window.
HIToolbarItemGetAttributesInWindow	Obtains the attributes of a toolbar item in the specified window.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kEventToolbarGetSelectableIdentifiers	This event is sent to the delegate after the user clicks a toolbar item in order to get a list of all the items that can acquire a selection highlight when clicked.
kEventToolbarItemSelectedStateChanged	This event is sent to the item itself when the selected state changes. Any interested parties can install handlers on the toolbar item to receive notifications. (Available in Mac OS X v10.4 and later.)
kHIToolbarItemDisabled	If this attribute bit is set, the item is disabled. Setting this attribute is the same as calling <code>HIToolbarItemSetEnabled</code> on the item with the <code>isEnabled</code> parameter set to false. (Available in Mac OS X v10.4 and later.)
kHIToolbarItemSelected	If this attribute bit is set, the item is drawn with a selected appearance. (Available in Mac OS X v10.4 and later.)

## HUIView.h

## Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIComboBoxIsListVisible	Determines whether a combo box disclosure list is visible.
HIComboBoxSetListVisible	Hides or shows a combo box disclosure list.
HICreateTransformedCGImage	Creates a new Core Graphics image with the standard selected or disabled appearance.
HIViewChangeTrackingArea	Changes the shape of a tracking area.
HIViewCopyShape	Copies the shape of a part of a view.
HIViewCopyText	Copies the text of a view.
HIViewCountSubviews	Returns the number of subviews embedded in a view.
HIViewDisposeTrackingArea	Disposes of an existing tracking area.
HIViewGetCommandID	Obtains the command ID of a view.
HIViewGetEventTarget	Returns the EventTargetRef for the specified view.
HIViewGetID	Obtains the HIViewID of a view.
HIViewGetIndexedSubview	Obtains the subview of a view by index.
HIViewGetKind	Obtains the signature and kind of a view.
HIViewGetMaximum	Obtains a view's maximum value.
HIViewGetMinimum	Obtains the minimum value of a view.
HIViewGetOptimalBounds	Obtains the optimal size and text placement of a view.
HIViewGetTrackingAreaID	Obtains the ID of a tracking area.
HIViewGetValue	Obtains the value of a view.
HIViewGetViewSize	Obtains the view size of a view.
HIViewIsActive	Determines whether a view is active.
HIViewIsCompositingEnabled	Determines whether compositing is enabled for a view.
HIViewIsEnabled	Determines whether a view is enabled.
HIViewIsLatentlyVisible	Determines whether a view is latently visible.

<code>HViewIsLayoutLatentlyActive</code>	Determines whether layout handling is latently active or suspended.
<code>HViewIsValid</code>	Determines whether the specified view is known to the HIToolbox.
<code>HViewNewTrackingArea</code>	Creates a new tracking area for a view.
<code>HViewSetActivated</code>	Sets a view to be active or inactive.
<code>HViewSetCommandID</code>	Sets the command ID of a view.
<code>HViewSetEnabled</code>	Enables or disables a view.
<code>HViewSetHilite</code>	Sets highlighting on a view.
<code>HViewSetID</code>	Sets the <code>HViewID</code> of a view.
<code>HViewSetMaximum</code>	Sets a view's maximum value.
<code>HViewSetMinimum</code>	Sets a view's minimum value.
<code>HViewSetNeedsDisplayInRect</code>	Uses an <code>HRect</code> to mark a portion of a view as needing or not needing to be redrawn.
<code>HViewSetNeedsDisplayInShape</code>	Uses a shape to mark a portion of a view as needing or not needing to be redrawn.
<code>HViewSetText</code>	Sets the text of a view to the specified string.
<code>HViewSetValue</code>	Sets the value of a view.
<code>HViewSetViewSize</code>	Sets the view size of a view.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HViewContentInfo</code>	Describes the content of a view.
<code>HViewContentInfoPtr</code>	
<code>HViewContentType</code>	
<code>HViewKind</code>	
<code>HViewTrackingAreaID</code>	Define an <code>HView</code> tracking area ID.
<code>HViewTrackingAreaRef</code>	Define an <code>HView</code> tracking area reference.
<code>kControlKindHIGrowBoxView</code>	Control kind for a grow box view.
<code>kControlKindHIImageView</code>	Control kind for an image view.

<code>kControlKindHIMenuView</code>	Control kind for a menu view.
<code>kControlKindHIScrollView</code>	Control kind for a scroll view.
<code>kControlKindHIStandardMenuView</code>	Control kind for a standard menu view.
<code>kEventClassClockView</code>	Event class for events related to a clock view.
<code>kEventClassHIComboBox</code>	Event class for events related to a combo box view.
<code>kEventClockDateOrTimeChanged</code>	An event sent by the clock control when the user changes the date or time.
<code>kEventComboBoxListItemSelected</code>	This event is sent as a notification when an item in the combo box disclosure list has been selected.
<code>kEventControlTrackingAreaEntered</code>	If you installed a mouse tracking area in your view, you will receive this event when the mouse enters that area. The tracking area reference is sent with the event.
<code>kEventControlTrackingAreaExited</code>	If you installed a mouse tracking area in your view, you will receive this event when the mouse leaves that area. The tracking area reference is sent with the event.
<code>kEventParamCandidateText</code>	The text that is to replace the selection.
<code>kEventParamComboBoxListSelectedItemIndex</code>	Indicates the index of a selected combo box list item.
<code>kEventParamHIViewTrackingArea</code>	An <code>HIViewTrackingAreaRef</code> for the tracking area that was entered.
<code>kEventParamReplacementText</code>	Optional replacement text.
<code>kEventParamTextSelection</code>	The range of the selection that is about to be changed. The units of the selection are in the same units that are returned in a <code>EditTextSelectionRec</code> , when called with <code>GetControlData</code> using <code>kControlEditTextSelectionTag</code> .
<code>kEventParamUnconfirmedRange</code>	If the text field currently has an open inline hole, this parameter contains the range of text inside the hole. This parameter is optional and is only present during inline editing.
<code>kEventParamUnconfirmedText</code>	If the text field currently has an open inline hole, this parameter contains the non-confirmed text currently being edited inside the hole.
<code>kEventSearchFieldSearchClicked</code>	
<code>kEventTextDidChange</code>	This event is sent to indicate that the contents of an editable text field have changed.



<code>kEventTextShouldChangeInRange</code>	This event is sent whenever the text is about to be modified in text field, either by user input or in other scenarios such as a paste from the clipboard, spell-checking word correction, or Mac OS X Service operation.
<code>kHISearchFieldAttributesSearchIcon</code>	Indicates that this view contains the Search icon in the text field.
<code>kHITransformDisabled</code>	The image should be transformed to use a disabled appearance. This transformation should not be combined with any other transformations.
<code>kHITransformNone</code>	No visual transformation should be applied.
<code>kHITransformSelected</code>	The image should be transformed to use a selected appearance. This transformation should not be combined with any other transformations.
<code>kHIViewAttributeIsFieldEditor</code>	Indicates that text editing controls should behave appropriately for editing fields in a dialog; specifically, the control should ignore the Return, Enter, Escape, and Tab keys, and allow them to be processed by other participants in the event flow.
<code>kHIViewAttributeSendCommandToUserFocus</code>	When set, the control sends the command it generates to the user focus; the command propagates as it would naturally from there.
<code>kHIViewClickableMetaPart</code>	Used for asynchronous window dragging only. The default is the structure region.
<code>kHIViewContentCGImageRef</code>	The view's content is an <code>CGImageRef</code> .
<code>kHIViewContentIconRef</code>	The view's content is an <code>IconRef</code> .
<code>kHIViewContentIconSuiteRef</code>	The view's content is an <code>IconSuiteRef</code> .
<code>kHIViewContentMetaPart</code>	The content region is only defined by views that can embed other views. It is the area that embedded content can live.
<code>kHIViewContentNone</code>	The view has no content.
<code>kHIViewContentTextOnly</code>	The view has no content other than text.
<code>kHIViewDisabledPart</code>	Disabled part.
<code>kHIViewEntireView</code>	The entire view; used when not referring to a specific part.
<code>kHIViewFeatureAllowsSubviews</code>	This view allows subviews to be embedded within it. Use this constant instead of the legacy constant, <code>kHIViewAllowsSubviews</code> .

<code>kHViewFeatureAutoToggles</code>	This view supports the auto-toggle protocol and should at the very least auto-toggle between off and on.
<code>kHViewFeatureDoesNotDraw</code>	This bit is an optimization that tells the Control Manager that a view is transparent and does not do any drawing, so the Control Manager doesn't have to invalidate the view and instead should invalidate views behind this view.
<code>kHViewFeatureDoesNotUseSpecialParts</code>	Indicates to the Control Manager that this view doesn't use the special part codes for indicator, inactive, and disabled. Use this constant instead of the legacy constant, <code>kHViewDoesNotUseSpecialParts</code> .
<code>kHViewFeatureGetsFocusOnClick</code>	If this view is clicked, the keyboard focus should be set to this view automatically; used primarily for edit text controls. Use this constant instead of the legacy constant, <code>kHViewGetsFocusOnClick</code> .
<code>kHViewFeatureIdlesWithTimer</code>	An informational bit.
<code>kHViewFeatureIgnoresClicks</code>	When set, this bit tells the Control Manager that it does not need to ask the control for its clickable region.
<code>kHViewFeatureInvertsUpDownValueMeaning</code>	This bit tells the Control Manager that when the Up button part of the control is clicked, the value of the control should increase.
<code>kHViewFeatureIsOpaque</code>	When set, the view's structure region is used to determine the view's opaque region, and calling the view can usually be avoided. Use this constant instead of the legacy constant, <code>kHViewIsOpaque</code> .
<code>kHViewFeatureSupportsGhosting</code>	This view supports using the ghosting protocol when live tracking is not enabled. Use this constant instead of the legacy constant, <code>kHViewSupportsGhosting</code> .
<code>kHViewFeatureSupportsLiveFeedback</code>	This view supports the live feedback protocol, which is necessary for implementing live scroll bar tracking.
<code>kHViewFeatureSupportsRadioBehavior</code>	This view can be placed in a radio group. Radio buttons and bevel buttons report this behavior. Use this constant instead of the legacy constant, <code>kHViewSupportsRadioBehavior</code> .
<code>kHViewFocusNextPart</code>	
<code>kHViewFocusNoPart</code>	
<code>kHViewFocusPrevPart</code>	

<code>kHIViewInactivePart</code>	Inactive part.
<code>kHIViewIndicatorPart</code>	Indicator part.
<code>kHIViewKindSignatureApple</code>	Signature of all toolbox views.
<code>kHIViewNoPart</code>	The entire view; used when not referring to a specific part.
<code>kHIViewOpaqueMetaPart</code>	The portion of the view that is opaque. No views behind this portion of the view will be asked to draw because their drawing would be completely overwritten by this view's drawing.
<code>kHIViewStructureMetaPart</code>	The structure region is the total area over which the view draws.
<code>typeHIViewTrackingAreaRef</code>	

## Keyboards.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kKLLanguageCode</code>	
------------------------------	--

## MacApplication.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HISearchWindowShow</code>	Displays a Spotlight search window.
---------------------------------	-------------------------------------

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kHIToolboxVersionNumber10_3</code>	
<code>kHIToolboxVersionNumber10_3_2</code>	
<code>kHIToolboxVersionNumber10_3_3</code>	
<code>kHIToolboxVersionNumber10_3_4</code>	

kHIToolboxVersionNumber10_3_5
-------------------------------

## MacHelp.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HMHideTagWithOptions
----------------------

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kHMHideTagFade
----------------

kHMHideTagImmediately
-----------------------

## MacTextEditor.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

DisposeTXNActionNameMapperUPP	Disposes of the universal procedure pointer (UPP) to your action name mapper callback function.
DisposeTXNContextualMenuSetupUPP	Disposes of the universal procedure pointer (UPP) to your contextual menu setup callback function.
HITextViewCopyBackgroundColor	Obtains the background color of the view.
HITextViewSetBackgroundColor	Sets the background color of the view.
InvokeTXNActionNameMapperUPP	Calls your action name mapper callback function.
InvokeTXNContextualMenuSetupUPP	Calls your contextual menu setup callback function.
NewTXNActionNameMapperUPP	Creates a new universal procedure pointer (UPP) to an action name mapper callback function.
NewTXNContextualMenuSetupUPP	Creates a new universal procedure pointer (UPP) to a contextual menu setup callback function.
TXNBeginActionGroup	Starts an action group.

TXNCanRedoAction	Indicates whether an action can be redone.
TXNCanUndoAction	Indicates whether an action can be undone.
TXNClearCountForActionType	Sets the counter for the specified action type to zero.
TXNCopyTypeIdentifiersForRange	Obtains an array of universal type identifiers for a TXNObject.
TXNEndActionGroup	Ends an action group.
TXNGetAccessibilityHIObject	Obtains an HIObjectRef representing the MLTE object for accessibility purposes.
TXNGetCommandEventSupport	Obtains the command event support that is currently set for an MLTE object.
TXNGetCountForActionType	Gets the number of times a given type of action has occurred.
TXNGetEventTarget	Obtains the current event target for a TXNObject.
TXNGetSpellCheckAsYouType	Determines whether the “Spell Check as You Type” feature is enabled.
TXNReadFromCFURL	Reads data from a CFURLRef into a TXNObject.
TXNSetActionNameMapper	Sets a callback that MLTE uses to obtain the localized string representing an action or an action group.
TXNSetCommandEventSupport	Enables and disables support for menu commands in MLTE.
TXNSetContextualMenuSetup	Provides a callback function that is called before MLTE displays its contextual menu.
TXNSetEventTarget	Sets a Carbon Event target for MLTE Carbon Event handlers.
TXNSetSpellCheckAsYouType	Enables and disables the “Spell Check as You Type” feature.
TXNWriteRangeToCFURL	Writes a range of a text object to a file or to a special file bundle.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kTXNActionAlignCenter	An align center action.
kTXNActionAlignLeft	An align left action.
kTXNActionAlignRight	An align right action.
kTXNActionChangeColor	A color change action.
kTXNActionChangeFont	A font change action.

<code>kTXNActionChangeFontFeature</code>	A change font feature action.
<code>kTXNActionChangeFontVariation</code>	A change in font variation action.
<code>kTXNActionChangeGlyphVariation</code>	A change glyph variation action.
<code>kTXNActionChangeSize</code>	A size change action.
<code>kTXNActionChangeStyle</code>	A change in style action.
<code>kTXNActionChangeTextPosition</code>	A change text position action; includes changing the space before and after characters and shifting the text's baseline.
<code>kTXNActionClear</code>	A clear action.
<code>kTXNActionCountOfAllChanges</code>	Total count of actions including all text and style changes, as well as custom defined actions.
<code>kTXNActionCountOfStyleChanges</code>	Count of text style changes. Style changes include changing font, font face, font size, font feature, font variation, font color, glyph variation, and text position. Undo or redo events of these kinds are also included in this action count.
<code>kTXNActionCountOfTextChanges</code>	Count of text changes.
<code>kTXNActionCut</code>	A cut action.
<code>kTXNActionDrop</code>	A drop action.
<code>kTXNActionMove</code>	A move action.
<code>kTXNActionNameMapperKey</code>	Indicates an action key mapper callback function. Available in Mac OS X v10.4; use instead of <code>kTXNActionKeyMapperKey</code> .
<code>kTXNActionPaste</code>	A paste action.
<code>kTXNActionTyping</code>	A typing action.
<code>kTXNActionUndoLast</code>	Used in undo and redo functions if none of the other constants apply.
<code>kTXNAutoScrollBehaviorTag</code>	Specifies autoscroll behavior. Available in Mac OS X v10.4 and later. For constants that represent the various types of autoscrolling, see Automatic Scrolling Behavior.
<code>kTXNAutoScrollInsertionIntoView</code>	The default auto scrolling behavior. When text is inserted, the document is scrolled to show the new insertion. This was the only type of autoscrolling prior to Mac OS X v10.4.

<code>kTXNAutoScrollNever</code>	Never autoscroll, even when dragging the mouse or inserting text. The only way to scroll the document is for the user to use the scrollbar or to scroll programatically.
<code>kTXNAutoScrollWhenInsertionVisible</code>	Autoscrolling only happens when the insertion offset is currently in the user's view.
<code>kTXNDataOptionCharacterEncodingKey</code>	CFNumber of type <code>kCFNumberSInt32Type</code> containing the character encoding as specified in <code>CFString.h</code> and <code>CFStringEncodingExt.h</code> .
<code>kTXNDataOptionDocumentTypeKey</code>	CFString specifying the document format.
<code>kTXNDefaultFontName</code>	The default font name.
<code>kTXNDisableLayoutAndDraw</code>	Layout and draw is disabled.
<code>kTXNDisableLayoutAndDrawTag</code>	Specifies visibility. Equivalent to <code>kTXNVisibilityTag</code> . Available in Mac OS X v10.4 and later. Use this tag to disable and re-enable layout and drawing. It optimizes performance when adding data incrementally to a text object.
<code>kTXNDocumentAttributeAuthorKey</code>	CFString containing the name of the document's author.
<code>kTXNDocumentAttributeCommentKey</code>	CFString containing comments.
<code>kTXNDocumentAttributeCompanyNameKey</code>	CFString containing the company name.
<code>kTXNDocumentAttributeCopyrightKey</code>	CFString containing the copyright of the document.
<code>kTXNDocumentAttributeCreationTimeKey</code>	CFAbsoluteTime containing document comments; note that this is not the file system creation date of the file, but of the document, as it is stored in the document.
<code>kTXNDocumentAttributeEditorKey</code>	CFString containing the name of the person who last edited the document.
<code>kTXNDocumentAttributeKeywordsKey</code>	CFArray of values of type CFString containing keywords.
<code>kTXNDocumentAttributeModificationTimeKey</code>	CFAbsoluteTime containing the last modification date of the document contents.
<code>kTXNDocumentAttributeSubjectKey</code>	CFString containing the document's subject.
<code>kTXNDocumentAttributeTitleKey</code>	CFString containing the document's title.
<code>kTXNEnableLayoutAndDraw</code>	Layout and draw is enabled.
<code>kTXNFontPanelEventHandlerKey</code>	Indicates the handler for Font Panel events.

<code>kTXNMLTEDocumentType</code>	Native MLTE document type.
<code>kTXNOperationNotAllowedErr</code>	Indicates that the function cannot be called in this context.
<code>kTXNPlainTextDocumentType</code>	Plain text document.
<code>kTXNQuickTimeDocumentType</code>	Multimedia file that can be opened by QuickTime importers. This document type is only supported for reading data, not for writing data.
<code>kTXNRTFDocumentType</code>	Rich text format (RTF) document type.
<code>kTXNSupportEditCommandProcessing</code>	Setting this bit when calling <code>TXNSetTXNObjectControls</code> enables support for processing the menu item associated with <code>kHICommandUndo</code> , <code>kHICommandRedo</code> , <code>kHICommandCut</code> , <code>kHICommandCopy</code> , <code>kHICommandPaste</code> , <code>kHICommandClear</code> , and <code>kHICommandSelectAll</code> .
<code>kTXNSupportEditCommandUpdating</code>	Setting this bit when calling <code>TXNSetTXNObjectControls</code> enables support for updating the menu item associated with <code>kHICommandUndo</code> , <code>kHICommandRedo</code> , <code>kHICommandCut</code> , <code>kHICommandCopy</code> , <code>kHICommandPaste</code> , <code>kHICommandClear</code> , and <code>kHICommandSelectAll</code> .
<code>kTXNSupportFontCommandProcessing</code>	Setting this bit enables Carbon Font Panel support.
<code>kTXNSupportFontCommandUpdating</code>	Setting this bit enables support for updating the selection in Carbon Font Panel when the current selection in an MLTE document is changed.
<code>kTXNSupportSpellCheckCommandProcessing</code>	Setting this bit when calling <code>TXNSetTXNObjectControls</code> enables support for spell checking.
<code>kTXNSupportSpellCheckCommandUpdating</code>	Enables support for updating the menu item associated with a given spell checking command.
<code>kTXNWheelMouseEventHandlerKey</code>	Indicates the handler for wheel mouse events.
<code>object</code>	
<code>TXNAutoScrollBehavior</code>	Specify automatic scrolling behavior for a text object.
<code>TXNCommandEventSupportOptions</code>	Specify options for enabling support for command events in a <code>TXNObject</code> .



## MacWindows.h

## Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

GetWindowGroupLevelOfType	Obtains the Core Graphics window level of a window group.
HIWindowChangeAvailability	Changes the availability of a window during Exposé or in Spaces.
HIWindowGetAvailability	Obtains the availability of a window during Exposé or in Spaces.
HIWindowGetProxyFSRef	Obtains the FSRef used to determine the proxy icon for a window.
HIWindowGetScaleMode	Obtains the window's scale mode and the application's display scale factor.
HIWindowInvalidateShadow	Recalculates a window's shadow.
HIWindowSetProxyFSRef	Sets the proxy icon for a window using an FSRef to a file system object.
SetWindowGroupLevelOfType	Sets the window level of a window group.

## Data Types &amp; Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIWindowAvailability	Define window availability constants for Exposé and Spaces.
HIWindowScaleMode	Define window scale mode constants.
kHIWindowExposeHidden	If this bit is set, the window is hidden during the "All Windows" and "Application windows" modes of Exposé. If this bit is not set, the window is visible during those modes.
kHIWindowScaleModeApplicationScaled	The window's contents are scaled manually by the application because the display scale factor is not equal to 1.0 and because the window was created with kWindowApplicationScaledAttribute.
kHIWindowScaleModeFrameworkScaled	The window's contents are scaled to match the display scale factor because the display scale factor is not equal to 1.0 and because the window was created with kWindowFrameworkScaledAttribute.

<code>kHIWindowScaleModeMagnified</code>	The window's backing store is being magnified by the Window Server because the display scale factor is not equal to 1.0 and because the window was not created with the <code>kWindowFrameworkScaledAttribute</code> .
<code>kHIWindowScaleModeUnscaled</code>	The window is not scaled at all because the display scale factor is 1.0.
<code>kWindowApplicationScaledAttribute</code>	If the bit specified by this mask is set, the details of resolution-independent scaling will be taken care of primarily by the application.
<code>kWindowFrameworkScaledAttribute</code>	If the bit specified by this mask is set, this window's content is scaled to match the display scale factor.
<code>kWindowGroupAttrFixedLevel</code>	If this attribute is specified, this window group's window level should be left unchanged.
<code>kWindowGroupLevelActive</code>	The window level that is nominally used for windows in the group when the application is active.
<code>kWindowGroupLevelInactive</code>	The window level for windows in the group when the application is inactive.
<code>kWindowGroupLevelPromoted</code>	The window level that is actually used for windows in the group when the application is active.
<code>kWindowMetalNoContentSeparatorAttribute</code>	If the bit specified by this mask is set, no border is drawn between the toolbar and window content. See <code>kHIWindowBitNoTexturedContentSeparator</code> .
<code>kWindowNoTitleBarAttribute</code>	If the bit specified by this mask is set, the window's title bar can be hidden.
<code>kWindowToolBarButtonRgn</code>	Bounds of the toolbar button area.

## Menus.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kMenuEisuGlyph</code>	
<code>kMenuKanaGlyph</code>	

## Scrap.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kScrapFlavorTypeUTF16External
-------------------------------

## TextServices.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

GetInputModePaletteMenu	Obtains from an input method the menu to display for a pull-down menu on the input mode palette.
InputModePaletteItemHit	Informs an input method that a function button on the input mode palette was pressed.
TSMInputModePaletteLoadButtons	Notifies the input mode palette of changes to the controls for an input method and replaces the current controls with the new control array.
TSMInputModePaletteUpdateButtons	Notifies the input mode palette of changes to the controls for an input method and updates the controls.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kCMGetInputModePaletteMenu	
kCMInputModePaletteItemHit	
kComponentBundleInvisibleInSystemUIKey	
kTSInputModePaletteItemAltIconKey	A CFString that specifies an alternate icon file name. The file should be located in the input method bundle resource directory, so this is just the file name, not full path.
kTSInputModePaletteItemEnabledKey	A CFBoolean that specifies the enabled state of the control.

<code>kTSInputModePaletteItemIconKey</code>	A CFString that specifies an icon file name. The file should be located in the input method bundle resource directory, so this is just the file name, not full path.
<code>kTSInputModePaletteItemIDKey</code>	A CFNumber that specifies a UInt32 tag ID for the control.
<code>kTSInputModePaletteItemKeyEquivalentKey</code>	A CFString that specifies a menu item keyboard shortcut .
<code>kTSInputModePaletteItemKeyEquivalentModifiersKey</code>	A CFNumber that specifies a menu item keyboard shortcut modifier (from Events.h).
<code>kTSInputModePaletteItemStateKey</code>	A CFNumber that specifies the state of the control (0: clear or unpressed, 1: checked or pressed, 2: mixed).
<code>kTSInputModePaletteItemTitleKey</code>	A CFString that specifies a menu item title. Use - for a separator.
<code>kTSInputModePaletteItemTypeKey</code>	A CFNumber that specifies the type of control (0: push button, 1: toggle button, 2: pull-down menu),
<code>kTSM23Version</code>	Specifies the version of the Text Services Manager is 2.3 (Mac OS X v10.4).
<code>kTSMDocumentWindowLevelPropertyTag</code>	

## Print

---

### PMApplication.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>PMSessionBeginCGDocument</code>	Begins a print job in which all drawing is to a Quartz graphics context. A printing status dialog informs the user of the job's progress.
---------------------------------------	---

### PMPrintingDialogExtensions.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

creator	
embedderUserPane	
GetSummaryText	
kPMInkPDEKindID	
printSession	
Prologue	
reinitializePlugIn	
summaryArray	
Sync	
Terminate	
titleArray	
userOptionKind	

## Ink

---

### Ink.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

InkStrokeGetPointCount	Returns the number of points in the specified InkStrokeRef.
InkStrokeGetPoints	Fills an array with the points belonging to the specified InkStrokeRef.
InkStrokeGetTypeID	Returns the CFTypeID of an InkStrokeRef object.
InkTextGetStroke	Returns a reference to the specified stroke in an InkTextRef.
InkTextGetStrokeCount	Returns the number of strokes in the specified InkTextRef.
InkTextGetTypeID	Returns the CFTypeID of an InkTextRef object.

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>InkStrokeRef</code>	Defines a data type for a reference to an opaque Ink stroke object.
<code>kInkPenLowerSideButtonMask</code>	The lower pen barrel button.
<code>kInkPenTipButtonMask</code>	The writing or eraser tip.
<code>kInkPenUpperSideButtonMask</code>	The upper pen barrel button.
<code>kInkTabletPointerCursor</code>	Any puck-like device.
<code>kInkTabletPointerEraser</code>	
<code>kInkTabletPointerPen</code>	The writing end of a stylus-like device.
<code>kInkTabletPointerUnknown</code>	The type of tablet pointer is unknown; having an unknown type of tablet pointer should not happen.
<code>kInkTerminationDefault</code>	Restores default phrase termination matching the current user settings ( <code>kInkTerminationTimeOut</code>   <code>kInkTerminationOutOfProximity</code>   <code>kInkTerminationRecognizerHorizontalBreak</code>   <code>kInkTerminationRecognizerVerticalBreak</code> ).
<code>kInkTerminationStroke</code>	Causes phrases to be terminated at the end of every stroke (whenever the pen is lifted from the tablet while writing). Only useful for single-stroke gesture input, not for text.

## ImageCapture

---

### ICAApplication.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>ICACloseSession</code>	
<code>ICALoadDeviceModule</code>	
<code>ICAOpenSession</code>	
<code>ICAUnloadDeviceModule</code>	
<code>ICAUploadFile</code>	

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

## 10.4 Symbol Changes

ICACloseSessionPB	
ICALoadDeviceModulePB	
ICAOpenSessionPB	
ICAUnloadDeviceModulePB	
ICAUploadFilePB	
kICABluetoothAddressKey	
kICABluetoothTransportType	
kICADeviceModulePathKey	
kICADeviceOther	
kICADevicePDA	
kICADevicePhone	
kICADownloadAndReturnPathArray	
kICAFireWireGUIDKey	
kICAFireWireTransportType	
kICAIOServicePathKey	
kICAIPAddressKey	
kICAIPGUIDKey	
kICAIPNameKey	
kICAIPPortKey	
kICAManualConnectDeviceList	
kICANetworkDeviceList	
kICASCSITransportType	
kICATCPIPTransportType	
kICATransportTypeKey	
kICATWAINDSPathKey	
kICATWAINTransportType	
kICAUploadFileAsIs	
kICAUploadFileScaleToFit	

kICAUSBLocationIDKey	
kICAUSBTransportType	
paramDictionary	

## NavigationServices

---

### Navigation.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

NavDialogSetFilterTypeIdentifiers	Sets UTI filtering criteria for “get file” and “choose file” dialogs.
-----------------------------------	---



# 10.3 Symbol Changes

---

This article lists the symbols added to `Carbon.framework` in Mac OS X v10.3.

## Classes

All of the classes with new symbols are listed alphabetically, with their new class, instance, and delegate methods described.

## C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

### CarbonSound

---

#### Sound.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kMicrosoftGSMCompression</code>	
<code>kMPEG4AudioFormat</code>	
<code>kSoundAudioCodecPropertyWritableFlag</code>	
<code>kSoundCodecInfoVariableDuration</code>	
<code>kVariableDurationDVAudioFormat</code>	
<code>propertyClass</code>	
<code>propertyData</code>	
<code>propertyDataSize</code>	
<code>propertyDataType</code>	

propertyRequestFlags	
propertyRequestResult	
siAudioCodecPropertyInfo	
siAudioCodecPropertyValue	
SoundAudioCodecPropertyRequestParams	

## CommonPanels

---

### FontPanel.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

FCAddCollection	
FCAddFontDescriptorToCollection	
FCCopyCollectionNames	
FCCopyFontDescriptorsInCollection	
FCFontDescriptorCreateWithFontAttributes	
FCFontDescriptorCreateWithName	
FCRemoveCollection	
FCRemoveFontDescriptorFromCollection	

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

FCFontDescriptorRef	
kEventParamDictionary	
kFCFontCGColorAttribute	
kFCFontFaceAttribute	
kFCFontFamilyAttribute	

kFCFontNameAttribute	
kFCFontSizeAttribute	
kFCFontVisibleNameAttribute	
kFontPanelATSUIFontIDKey	
kFontPanelAttributeSizesKey	
kFontPanelAttributesKey	
kFontPanelAttributeTagsKey	
kFontPanelAttributeValuesKey	
kFontPanelFeatureSelectorsKey	
kFontPanelFeatureTypesKey	
kFontPanelVariationAxesKey	
kFontPanelVariationValuesKey	

## OpenScripting

---

### OSA.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

OSADoScriptFile	Loads a script from the specified file, compiles the script if the file is a text file, executes the script, converts the resulting script value to text, and stores the script back into the file if the script has persistent properties and the file is not a text file.
OSALoadExecuteFile	Loads a script from the specified file into the specified scripting component, compiles the script if the file is a text file, and executes the script.
OSALoadFile	Loads a script from the specified file into the specified scripting component, compiling the script if the file is a text file.
OSAStoreFile	Stores a script into the specified file.

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kOSAModeFullyQualifyDescriptors</code>	This mode flag may be passed to <code>OSACoerceToDesc</code> to indicate that the resulting descriptor should be fully qualified (i.e. should include the root application reference).
--	--

## HI Toolbox

---

### Appearance.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>AppearancePartCode</code>	
<code>AppearanceRegionCode</code>	
<code>kAppearancePartDownButton</code>	
<code>kAppearancePartIndicator</code>	
<code>kAppearancePartLeftButton</code>	
<code>kAppearancePartMetaDisabled</code>	
<code>kAppearancePartMetaInactive</code>	
<code>kAppearancePartMetaNone</code>	
<code>kAppearancePartPageDownArea</code>	
<code>kAppearancePartPageLeftArea</code>	
<code>kAppearancePartPageRightArea</code>	
<code>kAppearancePartPageUpArea</code>	
<code>kAppearancePartRightButton</code>	
<code>kAppearancePartUpButton</code>	
<code>kAppearanceRegionCloseBox</code>	
<code>kAppearanceRegionCollapseBox</code>	
<code>kAppearanceRegionContent</code>	
<code>kAppearanceRegionDrag</code>	
<code>kAppearanceRegionGrow</code>	
<code>kAppearanceRegionStructure</code>	

### 10.3 Symbol Changes

kAppearanceRegionTitleBar	
kAppearanceRegionTitleProxyIcon	
kAppearanceRegionTitleText	
kAppearanceRegionZoomBox	
kThemeArrowButtonMini	
kThemeArrowButtonSmall	
kThemeBackgroundMetal	
kThemeBrushAlternatePrimaryHighlightColor	
kThemeComboBox	
kThemeComboBoxMini	
kThemeComboBoxSmall	
kThemeIncDecButtonMini	
kThemeIncDecButtonSmall	
kThemeMetricComboBoxLargeBottomShadowOffset	
kThemeMetricComboBoxLargeDisclosureWidth	
kThemeMetricComboBoxLargeRightShadowOffset	
kThemeMetricComboBoxMiniBottomShadowOffset	
kThemeMetricComboBoxMiniDisclosureWidth	
kThemeMetricComboBoxMiniRightShadowOffset	
kThemeMetricComboBoxSmallBottomShadowOffset	
kThemeMetricComboBoxSmallDisclosureWidth	
kThemeMetricComboBoxSmallRightShadowOffset	
kThemeMetricLittleArrowsMiniHeight	
kThemeMetricLittleArrowsMiniWidth	
kThemeMetricLittleArrowsSmallHeight	
kThemeMetricLittleArrowsSmallWidth	
kThemeMetricMiniCheckBoxHeight	
kThemeMetricMiniCheckBoxWidth	

### 10.3 Symbol Changes

kThemeMetricMiniDisclosureButtonHeight	
kThemeMetricMiniDisclosureButtonWidth	
kThemeMetricMiniHSliderHeight	
kThemeMetricMiniHSliderMinThumbWidth	
kThemeMetricMiniHSliderTickHeight	
kThemeMetricMiniHSliderTickOffset	
kThemeMetricMiniPopupButtonHeight	
kThemeMetricMiniPullDownHeight	
kThemeMetricMiniPushButtonHeight	
kThemeMetricMiniRadioButtonHeight	
kThemeMetricMiniRadioButtonWidth	
kThemeMetricMiniTabCapsWidth	
kThemeMetricMiniTabFrameOverlap	
kThemeMetricMiniTabHeight	
kThemeMetricMiniTabOverlap	
kThemeMetricMiniVSliderMinThumbHeight	
kThemeMetricMiniVSliderTickOffset	
kThemeMetricMiniVSliderTickWidth	
kThemeMetricMiniVSliderWidth	
kThemeMetricRoundTextFieldContentHeight	
kThemeMetricRoundTextFieldContentInsetBottom	
kThemeMetricRoundTextFieldContentInsetLeft	
kThemeMetricRoundTextFieldContentInsetRight	
kThemeMetricRoundTextFieldContentInsetTop	
kThemeMetricRoundTextFieldContentInsetWithIconLeft	
kThemeMetricRoundTextFieldContentInsetWithIconRight	
kThemeMetricRoundTextFieldMiniContentHeight	
kThemeMetricRoundTextFieldMiniContentInsetBottom	

## 10.3 Symbol Changes

kThemeMetricRoundTextFieldMiniContentInsetLeft	
kThemeMetricRoundTextFieldMiniContentInsetRight	
kThemeMetricRoundTextFieldMiniContentInsetTop	
kThemeMetricRoundTextFieldMiniContentInsetWithIconLeft	
kThemeMetricRoundTextFieldMiniContentInsetWithIconRight	
kThemeMetricRoundTextFieldSmallContentHeight	
kThemeMetricRoundTextFieldSmallContentInsetBottom	
kThemeMetricRoundTextFieldSmallContentInsetLeft	
kThemeMetricRoundTextFieldSmallContentInsetRight	
kThemeMetricRoundTextFieldSmallContentInsetTop	
kThemeMetricRoundTextFieldSmallContentInsetWithIconLeft	
kThemeMetricRoundTextFieldSmallContentInsetWithIconRight	
kThemeMetricSmallPaneSplitterHeight	The height of a small pane splitter. Should only be used in a window with thick borders, such as a textured window.
kThemeMetricSmallTabFrameOverlap	
kThemeMetricSmallTabOverlap	
kThemeMiniCheckBox	
kThemeMiniIndeterminateBar	
kThemeMiniProgressBar	
kThemeMiniRadioButton	
kThemeMiniScrollBar	
kThemeMiniSlider	
kThemeMiniSystemFont	
kThemePoofCursor	The cursor identified by this constant is typically used to indicate that a dragged object will go away if it is released at the current mouse position. When the object goes away, a poof cloud animation should occur.

kThemePopupButtonMini	
kThemePopupButtonNormal	
kThemePopupButtonSmall	
kThemePushButtonMini	
kThemePushButtonNormal	
kThemePushButtonSmall	
kThemeResizeDownCursor	The cursor identified by this constant is typically used to indicate that an object may be resized by dragging vertically in the down direction.
kThemeResizeUpCursor	The cursor identified by this constant is typically used to indicate that an object may be resized by dragging vertically in the up direction.
kThemeResizeUpDownCursor	The cursor identified by this constant is typically used to indicate that an object may be resized by dragging vertically in either direction.

## CarbonEvents.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CopySymbolicHotKeys	Obtains information about symbolic hot keys in the Keyboard preferences pane.
GetEventManagerTarget	Obtains an event monitor target.
HIMouseTrackingGetParameters	Obtains information about how mouse tracking loops should behave.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kAvailBoundsChangedForDisplay	
-------------------------------	--



kAvailBoundsChangedForDock	
kEventAccessibleGetAllParameterizedAttributeNames	A request sent by an assistive application to get the names of all parameterized attributes the given accessibility object supports.
kEventAppActiveWindowChanged	The active window in the current process has changed.
kEventAppAvailableWindowBoundsChanged	The available window positioning bounds have changed.
kEventAppIsEventInInstantMouser	The given event's global mouse location is over an "instant mousing" area.
kEventClassInk	Events related to ink.
kEventClassSystem	Events related to the system.
kEventClassToolbarItemView	Events related to toolbar item views.
kEventClassTSMDocumentAccess	Events related to Text Services Manager document access.
kEventControlGetFrameMetrics	
kEventControlInvalidateForSizeChange	
kEventControlLayoutInfoChanged	
kEventControlVisibilityChanged	
kEventInkGesture	The Ink Manager recognizes the current ink phrase as one of the known system gestures.
kEventInkPoint	A mouse event will be handled as an ink point and used for recognition.
kEventInkText	The Ink Manager recognizes a word. The kEventParamInkTextRef parameter contains the ink text reference with all the information about the word. For more information, see Ink.h.
kEventMenuBarHidden	Sent to all handlers registered for this event when the front process hides its menubar. This event is sent only to the application target.
kEventMenuBarShown	Sent to all handlers registered for this event when the front process shows its menubar. This event is sent only to the application target.

## 10.3 Symbol Changes

<code>kEventMenuBecomeScrollable</code>	Requests that a menu content view prepare to be scrollable, which it does by installing the appropriate event handlers, timers, and the like.
<code>kEventMenuCalculateSize</code>	Sent by <code>CalcMenuSize</code> to request that the menu calculate its size.
<code>kEventMenuCeaseToBeScrollable</code>	Requests that a menu content view cease to be scrollable.
<code>kEventMenuCreateFrameView</code>	Requests that a menu content view create the <code>HIView</code> that will be used to draw the menu window frame.
<code>kEventMenuGetFrameBounds</code>	Requests that a menu content view calculate the bounding rect, in global coordinates, of the menu window frame that should contain the menu.
<code>kEventParamAccessibilityEventQueued</code>	
<code>kEventParamAccessibleAttributeParameter</code>	Specifies the parameters of a parameterized attribute. The parameter type is <code>typeCFTyperef</code> .
<code>kEventParamBounds</code>	<code>typeQDRectangle</code>
<code>kEventParamControlFrameMetrics</code>	<code>typeControlFrameMetrics</code>
<code>kEventParamControlHit</code>	<code>typeBoolean</code>
<code>kEventParamControlPartAutoRepeats</code>	<code>typeBoolean</code>
<code>kEventParamControlWouldAcceptDrop</code>	<code>typeBoolean</code>
<code>kEventParamCurrentDockDevice</code>	<code>typeGDHandle</code>
<code>kEventParamCurrentWindow</code>	
<code>kEventParamGDevice</code>	<code>typeGDHandle</code>
<code>kEventParamIndex</code>	<code>typeCFIndex</code>
<code>kEventParamInkGestureBounds</code>	Bounds of the gesture in global coordinates. ( <code>typeHIRect</code> )
<code>kEventParamInkGestureHotspot</code>	Hotspot, in global coordinates, for the gesture. ( <code>typeHIPoint</code> )
<code>kEventParamInkGestureKind</code>	Kind of gesture. ( <code>typeUInt32</code> )

### 10.3 Symbol Changes

kEventParamInkKeyboardShortcut	A Boolean whose value indicates whether the word the Ink Manager recognized is a keyboard shortcut. That is, the Command or Control key was pressed and the top-choice alternate text is a single character. (typeBoolean)
kEventParamInkTextRef	The ink text reference containing the data for the word the Ink Manager recognized. (typePtr)
kEventParamIsInInstantMouser	
kEventParamMenuDirection	
kEventParamMenuDismissed	
kEventParamMenuFrameView	
kEventParamMenuIsPopup	
kEventParamMenuPopupItem	
kEventParamMenuType	
kEventParamParentMenu	
kEventParamParentMenuItem	
kEventParamPasteboardRef	
kEventParamPreviousDockDevice	typeGDHandle
kEventParamPreviousWindow	
kEventParamReason	typeUInt32
kEventParamTextInputReplyGlyphInfoArray	typeGlyphInfoArray
kEventParamTextInputReplyMacEncoding	
kEventParamTextInputSendGlyphInfoArray	
kEventParamTextInputSendReplaceRange	typeCFRange
kEventParamTransactionID	typeUInt32
kEventParamTSMDocAccessCharacterCount	typeCFIndex
kEventParamTSMDocAccessEffectiveRange	typeRange
kEventParamTSMDocAccessLockCount	typeCFIndex
kEventParamTSMDocAccessReplyATSTFont	typeATSTFontRef
kEventParamTSMDocAccessReplyATSUGlyphSelector	typeGlyphSelector

### 10.3 Symbol Changes

kEventParamTSMDocAccessReplyCharacterRange	typeCFRange
kEventParamTSMDocAccessReplyCharactersPtr	typePtr
kEventParamTSMDocAccessReplyFontSize	typeFloat
kEventParamTSMDocAccessRequestedCharacterAttributes	typeUInt32
kEventParamTSMDocAccessSendCharacterIndex	typeCFIndex
kEventParamTSMDocAccessSendCharacterRange	The parameter data type is typeCFRange.
kEventParamTSMDocAccessSendCharactersPtr	typePtr
kEventParamTSMDocAccessSendComponentInstance	typeComponentInstance
kEventParamTSMDocAccessSendRefCon	typeLongInteger
kEventParamTSMSendComponentInstance	This parameter is equivalent to the text input parameter kEventParamTextInputSendComponentInstance; the parameter data type is typeComponentInstance.
kEventParamTSMSendRefCon	This parameter is equivalent to the text input parameter kEventParamTextInputSendRefCon; the parameter data type is typeLongInteger.
kEventParamUserData	typeVoidPtr
kEventParamWindowPartCode	typeWindowPartCode
kEventParamWindowTransitionAction	typeWindowTransitionAction
kEventParamWindowTransitionEffect	typeWindowTransitionEffect
kEventSystemTimeDateChanged	
kEventSystemUserSessionActivated	
kEventSystemUserSessionDeactivated	
kEventTSMDocumentAccessGetCharacters	This fills a caller provided buffer with Unicode characters in the specified range.
kEventTSMDocumentAccessGetCharactersPtr	Returns a Unicode pointer to the entire document content.
kEventTSMDocumentAccessGetCharactersPtrForLargestBuffer	Returns a Unicode pointer to a portion of the document.

### 10.3 Symbol Changes

kEventTSMDocumentAccessGetFont	Returns font, font size, and the range over which these attributes are constant.
kEventTSMDocumentAccessGetGlyphInfo	Returns glyph info and the range covered by that glyph.
kEventTSMDocumentAccessGetLength	Returns the number of 16-bit Unicode characters in the document.
kEventTSMDocumentAccessGetSelectedRange	Returns the selection range in the document.
kEventTSMDocumentAccessLockDocument	Notifies the application that it should not change its document's text content (on its own) while a text service is involved in a transaction.
kEventTSMDocumentAccessUnlockDocument	Unlock the document so the application text engine is free to initiate changes again.
kEventWindowTransitionCompleted	Sent to all handlers registered for it.
kEventWindowTransitionStarted	Sent to all handlers registered for it.
kHICommandShowCharacterPalette	The character palette needs to be shown. Events with this command ID are only generated in Mac OS X v10.3 and later. The toolbox will respond to this event automatically; your application does not need to handle it.
kHIMenuCenterDirection	
kHIMenuLeftDirection	
kHIMenuRightDirection	
kMouseParamsSticky	Requests the time and distance for determining "sticky" mouse tracking.
kTSMDocAccessEffectiveRangeAttribute	Use to set or test for the kTSMDocAccessEffectiveRangeAttributeBit bit.
kTSMDocAccessEffectiveRangeAttributeBit	When this bit is set, indicates to obtain effective range information used in the Carbon events kEventTSMDocumentAccessGetFont and kEventTSMDocumentAccessGetGlyphInfo.
kTSMDocAccessFontSizeAttribute	Use to set or test for the kTSMDocAccessFontSizeAttributeBit bit.

kTSMDocAccessFontSizeAttributeBit	When this bit is set, indicates to obtain font size information; used in the Carbon event <code>kEventTSMDocumentAccessGetFont</code> .
TSMDocAccessAttributes	Used to set or test for document-access attributes.
typeATSTFontRef	ATSTFontRef
typeCFRange	
typeControlFrameMetrics	HIViewFrameMetrics
typeGDHandle	GDHandle
typeGlyphSelector	ATSUGlyphSelector
typeMenuDirection	
typePasteboardRef	
typeThemeMenuType	
typeWindowPartCode	WindowPartCode
typeWindowTransitionAction	WindowTransitionAction
typeWindowTransitionEffect	WindowTransitionEffect

## CarbonEventsCore.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

AcquireFirstMatchingEventInQueue	Obtains the first event that matches the specified list of event classes and kinds.
CopyEventAs	Copies an existing event, allowing you to change the class and kind of the event.
DisableSecureEventInput	Disables secure event input mode.
EnableSecureEventInput	Enables secure event input mode.
IsSecureEventInputEnabled	Determines whether secure event input mode is enabled.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

eventDeferAccessibilityEventErr	
eventPassToNextTargetErr	
kEventAttributeMonitored	An event that was not originally targeted to this process but has been provided to this process because an event handler for this event type has been installed on the event monitoring target.
kEventQueueOptionsNone	
typeCFArrayRef	A Core Foundation array.
typeCFDictionaryRef	A Core Foundation dictionary.
typeCFMutableDictionaryRef	A Core Foundation mutable dictionary.

## ControlDefinitions.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlBevelButtonIsMultiValueMenuTag	
kControlEditTextInsertCFStringRefTag	Gets or sets the control's text as a CFString.
kControlEditTextInsertTextBufferTag	Gets or sets the control's text as WorldScript encoded text.
kControlGroupBoxFrameRectTag	Gets the full rectangle that content is drawn in.
kControlLittleArrowsIncrementValueTag	Gets or sets the increment value of the control.
kControlStaticTextIsMultilineTag	Gets or sets a flag specifying whether the control draws its text in multiple lines if the text is too wide for the control bounds.
kControlTabSizeMini	
kControlWindowHeaderIsListHeaderTag	Set to true if the control is to draw as a list header.
kDataBrowserListViewTypeSelectColumn	The column is capable of being selected and having text entered.

## Controls.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kControlClickableMetaPart</code>	The area of the control that causes a mouse event to be captured by that control. If a mouse event falls inside the control bounds but outside of this area, then the Control Manager will allow the event to pass through the control to the next control behind it in z-order. This area is used to determine which parts of a window should allow async window dragging when clicked (the draggable area is computed by subtracting the clickable areas of controls in the window from the window's total area). You can also customize the clickable area of a control if you want the control to have an effectively transparent area (for example, a control that draws multiple tabs might want clicks in the space between the tabs to fall through to the next control rather than be captured by the tab-drawing control). The default handler for the <code>kEventControlGetPartRegion</code> event returns the control's bounds when this part is requested.
<code>kControlContentCGImageRef</code>	
<code>kControlFontMiniSystemFont</code>	
<code>kControlInvertsUpDownValueMeaning</code>	
<code>kControlOpaqueMetaPart</code>	The area of the control that, when drawn, is filled with opaque pixels. You may also return a subset of the opaque area if this is computationally easier to construct. If a control is contained in a composited window, the Control Manager will use this area to optimize drawing of other controls that intersect this area; controls that are entirely contained within the opaque area, and that are z-ordered underneath this control, will not be drawn at all, since any drawing would be completely overwritten by this control. The default handler for the <code>kEventControlGetPartRegion</code> event returns an empty area when this part is requested.
<code>kControlSizeMini</code>	

## Drag.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>GetDragPasteboard</code>	
<code>NewDragWithPasteboard</code>	



## Events.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kPencilLeftUnicode
--------------------

## HITheme.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIThemeApplyBackground	Para
HIThemeDrawBackground	Para
HIThemeDrawButton	Draws a theme-compliant button.
HIThemeDrawChasingArrows	Para
HIThemeDrawFocusRect	Para
HIThemeDrawFrame	Para
HIThemeDrawGenericWell	Para
HIThemeDrawGrabber	Para
HIThemeDrawGroupBox	Para
HIThemeDrawGrowBox	Para
HIThemeDrawHeader	Para
HIThemeDrawMenuBackground	Para
HIThemeDrawMenuBarBackground	Para
HIThemeDrawMenuItem	Para
HIThemeDrawMenuSeparator	Para
HIThemeDrawMenuTitle	Para
HIThemeDrawPaneSplitter	Para
HIThemeDrawPlacard	Para

### 10.3 Symbol Changes

HIThemeDrawPopupArrow	Para
HIThemeDrawScrollBarDelimiters	Para
HIThemeDrawSeparator	Para
HIThemeDrawTab	Para
HIThemeDrawTabPage	Para
HIThemeDrawTextBox	Para
HIThemeDrawTickMark	Para
HIThemeDrawTitleBarWidget	Para
HIThemeDrawTrack	Para
HIThemeDrawTrackTickMarks	Para
HIThemeDrawWindowFrame	Para
HIThemeGetButtonBackgroundBounds	Para
HIThemeGetButtonContentBounds	Para
HIThemeGetButtonShape	Para
HIThemeGetGrowBoxBounds	Para
HIThemeGetMenuBackgroundShape	Para
HIThemeGetScrollBarTrackRect	Para
HIThemeGetTabDrawShape	Para
HIThemeGetTabPageContentShape	Para
HIThemeGetTabPageDrawShape	Para
HIThemeGetTabShape	Para
HIThemeGetTextDimensions	Gets the text dimensions of a string.
HIThemeGetTrackBounds	Para
HIThemeGetTrackDragRect	Para
HIThemeGetTrackLiveValue	Para
HIThemeGetTrackPartBounds	Para
HIThemeGetTrackParts	Para
HIThemeGetTrackThumbPositionFromBounds	Para

## 10.3 Symbol Changes

<code>HIThemeGetTrackThumbPositionFromOffset</code>	Para
<code>HIThemeGetTrackThumbShape</code>	Para
<code>HIThemeGetWindowRegionHit</code>	Para
<code>HIThemeGetWindowShape</code>	Para
<code>HIThemeHitTestScrollBarArrows</code>	Para
<code>HIThemeHitTestTrack</code>	Para

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>animation</code>	
<code>condensedTitleExtra</code>	
<code>HIScrollBarTrackInfo</code>	Drawing parameters passed to scroll bar drawing and measuring functions.
<code>HIThemeAnimationFrameInfo</code>	
<code>HIThemeAnimationTimeInfo</code>	
<code>HIThemeBackgroundDrawInfo</code>	
<code>HIThemeBackgroundDrawInfoPtr</code>	
<code>HIThemeButtonDrawInfo</code>	
<code>HIThemeButtonDrawInfoPtr</code>	
<code>HIThemeChasingArrowsDrawInfo</code>	
<code>HIThemeChasingArrowsDrawInfoPtr</code>	
<code>HIThemeFrameDrawInfo</code>	
<code>HIThemeFrameDrawInfoPtr</code>	
<code>HIThemeFrameKind</code>	
<code>HIThemeGrabberDrawInfo</code>	
<code>HIThemeGrabberDrawInfoPtr</code>	
<code>HIThemeGroupBoxDrawInfo</code>	
<code>HIThemeGroupBoxDrawInfoPtr</code>	

### 10.3 Symbol Changes

HIThemeGroupBoxKind	
HIThemeGrowBoxDrawInfo	
HIThemeGrowBoxDrawInfoPtr	
HIThemeGrowBoxKind	
HIThemeGrowBoxSize	
HIThemeHeaderDrawInfo	
HIThemeHeaderDrawInfoPtr	
HIThemeHeaderKind	
HIThemeMenuBarDrawInfo	
HIThemeMenuBarDrawInfoPtr	
HIThemeMenuDrawInfo	
HIThemeMenuDrawInfoPtr	
HIThemeMenuItemDrawInfo	
HIThemeMenuItemDrawInfoPtr	
HIThemeMenuTitleDrawInfo	
HIThemeMenuTitleDrawInfoPtr	
HIThemeOrientation	Constants that indicate the orientation of a Quartz graphics context.
HIThemePlacardDrawInfo	
HIThemePlacardDrawInfoPtr	
HIThemePopupArrowDrawInfo	
HIThemePopupArrowDrawInfoPtr	
HIThemeScrollBarDelimitersDrawInfo	
HIThemeScrollBarDelimitersDrawInfoPtr	
HIThemeSeparatorDrawInfo	
HIThemeSeparatorDrawInfoPtr	
HIThemeSplitterAdornment	

### 10.3 Symbol Changes

HIThemeSplitterDrawInfo	
HIThemeSplitterDrawInfoPtr	
HIThemeTabAdornment	
HIThemeTabDrawInfo	
HIThemeTabPageDrawInfo	
HIThemeTabPageDrawInfoPtr	
HIThemeTabSize	
HIThemeTextBoxOptions	
HIThemeTextHorizontalFlush	
HIThemeTextInfo	
HIThemeTextTruncation	
HIThemeTextVerticalFlush	
HIThemeTickMarkDrawInfo	
HIThemeTickMarkDrawInfoPtr	
HIThemeTrackDrawInfo	
HIThemeWindowDrawInfo	
HIThemeWindowDrawInfoPtr	
HIThemeWindowWidgetDrawInfo	
HIThemeWindowWidgetDrawInfoPtr	
horizontalFlushness	
isFocused	
kHIThemeFrameListBox	
kHIThemeFrameTextFieldSquare	
kHIThemeGroupBoxKindPrimary	
kHIThemeGroupBoxKindPrimaryOpaque	
kHIThemeGroupBoxKindSecondary	
kHIThemeGroupBoxKindSecondaryOpaque	
kHIThemeGrowBoxKindNone	

### 10.3 Symbol Changes

<code>kHIThemeGrowBoxKindNormal</code>	
<code>kHIThemeGrowBoxSizeNormal</code>	
<code>kHIThemeGrowBoxSizeSmall</code>	
<code>kHIThemeHeaderKindList</code>	
<code>kHIThemeHeaderKindWindow</code>	
<code>kHIThemeMenuItemDrawCondensed</code>	
<code>kHIThemeOrientationInverted</code>	The graphics context has an origin at the bottom left.
<code>kHIThemeOrientationNormal</code>	The graphics context has an origin at the top left.
<code>kHIThemeSplitterAdornmentMetal</code>	
<code>kHIThemeSplitterAdornmentNone</code>	
<code>kHIThemeTabAdornmentFocus</code>	
<code>kHIThemeTabAdornmentNone</code>	
<code>kHIThemeTabSizeMini</code>	
<code>kHIThemeTabSizeNormal</code>	
<code>kHIThemeTabSizeSmall</code>	
<code>kHIThemeTextBoxOptionNone</code>	
<code>kHIThemeTextBoxOptionStronglyVertical</code>	
<code>kHIThemeTextHorizontalFlushCenter</code>	
<code>kHIThemeTextHorizontalFlushLeft</code>	
<code>kHIThemeTextHorizontalFlushRight</code>	
<code>kHIThemeTextInfoVersionZero</code>	
<code>kHIThemeTextTruncationEnd</code>	
<code>kHIThemeTextTruncationMiddle</code>	
<code>kHIThemeTextTruncationNone</code>	
<code>kHIThemeTextVerticalFlushBottom</code>	
<code>kHIThemeTextVerticalFlushCenter</code>	

kHIThemeTextVerticalFlushTop	
truncationHappened	
truncationMaxLines	
truncationPosition	
verticalFlushness	
widgetState	
widgetType	
windowState	
windowType	

## HIToolbar.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIToolbarItemConfigDataChanged	Tells the toolbar that the configuration for a toolbar item has changed.
HIToolbarSetItemsWithIdentifiers	Sets a toolbar's items all at once.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kEventParamToolbarDisplayMode	Indicates that the display mode changed.
kEventParamToolbarDisplaySize	Indicates that the display size changed.
kEventToolbarBeginMultiChange	This event is sent when multiple attributes are going to be changed at the same time.
kEventToolbarDisplayModeChanged	This event is sent when the display mode is changed for a toolbar. The toolbar object sends this event to itself, so you need to install a handler on the toolbar to receive this event. (Available in Mac OS X v10.2 and later.)
kEventToolbarDisplaySizeChanged	This event is sent when the display size is changed for a toolbar. The toolbar object sends this event to itself, so you need to install a handler on the toolbar to receive this event. (Available in Mac OS X v10.2 and later.)

<code>kEventToolbarEndMultiChange</code>	This event is sent when multiple changes in the toolbar have been made.
<code>kEventToolbarItemAcceptDrop</code>	This event is sent when a drag enters a toolbar item.
<code>kEventToolbarItemAdded</code>	This event is sent when an item is added to the toolbar. The toolbar object sends this event to itself, so you need to install a handler on the toolbar to receive this event. (Available in Mac OS X v10.2 and later.)
<code>kEventToolbarItemRemoved</code>	This event is sent when an item is removed from toolbar.
<code>kEventToolbarItemViewConfigForMode</code>	This event notifies a toolbar item view that the toolbar's display mode has changed.
<code>kEventToolbarItemViewConfigForSize</code>	This event notifies a toolbar item view that the toolbar's display size has changed.
<code>kEventToolbarItemViewEnterConfigMode</code>	This event notifies a toolbar item view that configure mode has been entered.
<code>kEventToolbarItemViewExitConfigMode</code>	This event notifies a toolbar item view that configure mode has been exited. A custom toolbar item view can respond to this event in any way it sees fit. (Available in Mac OS X v10.3 and later.)
<code>kEventToolbarItemWouldAcceptDrop</code>	This event is sent when a toolbar item is clicked.
<code>kEventToolbarLayoutChanged</code>	Sent when the layout of a toolbar changes (either an item has been moved, or the entire contents have been replaced).
<code>kHIToolbarCommandPressAction</code>	This command, when specified as a toolbar's command ID, causes a <code>kEventToolbarItemPerformAction</code> event to be generated when the toolbar item's menu item in the toolbar overflow menu is selected.
<code>kHIToolbarItemLabelDisabled</code>	If this attribute bit is set, clicking on the label of an item does nothing.
<code>typeHIToolbarDisplayMode</code>	<code>HIToolbarDisplayMode</code> .
<code>typeHIToolbarDisplaySize</code>	<code>HIToolbarDisplaySize</code> .

## HIView.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HIMenuGetContentView</code>	Obtains an <code>HIViewRef</code> that can be used to draw menu content for a menu.
-----------------------------------	---



### 10.3 Symbol Changes

<code>HIMenuViewGetMenu</code>	Returns the <code>MenuRef</code> associated with a view that is a subclass of <code>HIMenuView</code> .
<code>HIScrollViewCanNavigate</code>	Determines whether it is possible to navigate in a scroll view.
<code>HIScrollViewNavigate</code>	Changes the portion of a view's target.
<code>HISearchFieldChangeAttributes</code>	Sets the attributes of a search field.
<code>HISearchFieldCopyDescriptiveText</code>	Obtains the description associated with a search field.
<code>HISearchFieldCreate</code>	Creates a Search field control.
<code>HISearchFieldGetAttributes</code>	Obtains the attributes for a search field.
<code>HISearchFieldGetSearchMenu</code>	Obtains the search menu associated with a search field.
<code>HISearchFieldSetDescriptiveText</code>	Sets the description of the search action for a search field.
<code>HISearchFieldSetSearchMenu</code>	Sets the search menu associated with a search field view.
<code>HISegmentedViewChangeSegmentAttributes</code>	Changes the attributes of an individual segment of a segmented view.
<code>HISegmentedViewCopySegmentImage</code>	Copies the image drawn by a segment.
<code>HISegmentedViewCopySegmentLabel</code>	Obtains a copy of the label string associated with a segment.
<code>HISegmentedViewCreate</code>	Creates a segmented view.
<code>HISegmentedViewGetSegmentAttributes</code>	Returns the attributes of an individual segment of a segmented view.
<code>HISegmentedViewGetSegmentBehavior</code>	Obtains the behavior of an individual segment of a segmented view.
<code>HISegmentedViewGetSegmentCommand</code>	Obtains the command ID associated with a segment.
<code>HISegmentedViewGetSegmentContentWidth</code>	Obtains the content width of a segment.
<code>HISegmentedViewGetSegmentCount</code>	Obtains the number of segments for a segmented view.
<code>HISegmentedViewGetSegmentImageContentType</code>	Obtains the type of image content drawn by a segment.
<code>HISegmentedViewGetSegmentValue</code>	Returns the value of an individual segment of a segmented view.

## 10.3 Symbol Changes

<code>HISegmentedViewIsSegmentEnabled</code>	Determines whether an individual segment of a segmented view is enabled.
<code>HISegmentedViewSetSegmentBehavior</code>	Changes the behavior of an individual segment of a segmented view.
<code>HISegmentedViewSetSegmentCommand</code>	Sets the command ID for a segment.
<code>HISegmentedViewSetSegmentContentWidth</code>	Specifies how the content width of segment is to be calculated.
<code>HISegmentedViewSetSegmentCount</code>	Sets the number of segments for a segmented view.
<code>HISegmentedViewSetSegmentEnabled</code>	Enables or disables an individual segment of a segmented view.
<code>HISegmentedViewSetSegmentImage</code>	Sets or clears the image associated with a segment.
<code>HISegmentedViewSetSegmentLabel</code>	Sets the label string for a segment.
<code>HISegmentedViewSetSegmentValue</code>	Changes the value of an individual segment of a segmented view.
<code>HIViewApplyLayout</code>	Applies the current layout to the specified view.
<code>HIViewChangeFeatures</code>	Changes the features of a view.
<code>HIViewGetFeatures</code>	Obtains the features of the specified view.
<code>HIViewGetLayoutInfo</code>	Obtains the layout information of an view.
<code>HIViewGetWindow</code>	Obtains a reference to the window to which the specified view is bound.
<code>HIViewIsLayoutActive</code>	Determines whether layout handling is active or suspended.
<code>HIViewRegionChanged</code>	Informs the system that a region of the view has changed.
<code>HIViewRender</code>	Renders the invalid portions of a view.
<code>HIViewResumeLayout</code>	Resumes layout handling for a view and its children.
<code>HIViewSetLayoutInfo</code>	Sets the layout information of an HIView.
<code>HIViewSuspendLayout</code>	Suspends layout handling for a view and its children.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

## 10.3 Symbol Changes

binding	
errNeedsCompositedWindow	This result code is returned by an HView or a Control Manager function when an action that requires a compositing window is attempted on a non-compositing window. It may also be returned when the value of a parameter is not valid for the requested action, even though the window is a compositing window.
HIAxisPosition	
HIAxisScale	
HIBinding	
HIBindingKind	Specify binding constants used by the HView layout engine.
HILayoutInfo	
HIPositioning	
HIPositionKind	Specify position constants used by the HView layout engine.
HIScaleKind	Specify a constant that indicates the scale is determined from the axis size.
HIScaling	
HIScrollViewAction	Specify constants for scroll view navigation actions.
HISegmentBehavior	
HISideBinding	
HIViewFeatures	Specify constants for view features.
HIViewFrameMetrics	
HIViewImageContentInfo	
HIViewImageContentType	
kControlKindHISearchField	Control kind for a search field.
kControlSearchFieldCancelPart	Cancel part.
kControlSearchFieldMenuPart	Menu part.
kEventClassSearchField	Event calls for events related to a search field view.
kEventClassTextField	Event class for events related to text field views.
kEventSearchFieldCancelClicked	

### 10.3 Symbol Changes

kEventTextAccepted	This event is sent as a notification when the text contained in a control's editable text field has been accepted by the user.
kHILayoutBindBottom	Synonym for kHILayoutBindMax.
kHILayoutBindLeft	Synonym for kHILayoutBindMin.
kHILayoutBindMax	Bind to the maximum of the axis.
kHILayoutBindMin	Bind to the minimum of the axis.
kHILayoutBindNone	No binding is to occur.
kHILayoutBindRight	Synonym for kHILayoutBindMax.
kHILayoutBindTop	Synonym for kHILayoutBindMin.
kHILayoutInfoVersionZero	The version of the HILayoutInfo structure is 0.
kHILayoutPositionBottom	Synonym for kHILayoutPositionMax.
kHILayoutPositionCenter	Bind to the center.
kHILayoutPositionLeft	Synonym for kHILayoutPositionMin.
kHILayoutPositionMax	Bind to the maximum of the axis.
kHILayoutPositionMin	Bind to the minimum of the axis.
kHILayoutPositionNone	No positioning is to occur.
kHILayoutPositionRight	Synonym for kHILayoutPositionMax.
kHILayoutPositionTop	Synonym for kHILayoutPositiondMin.
kHILayoutScaleAbsolute	Indicates that the scale is determined from the axis size.
kHIScrollViewPageDown	The scroll view should page down.
kHIScrollViewPageLeft	The scroll view should page left.
kHIScrollViewPageRight	The scroll view should page right.
kHIScrollViewPageUp	The scroll view should page up.
kHIScrollViewScrollToBottom	The scroll view should move to the bottom of the content.
kHIScrollViewScrollToLeft	The scroll view should move to the left of the content.
kHIScrollViewScrollToRight	The scroll view should move to the right of the content.
kHIScrollViewScrollToTop	The scroll view should move to the top of the content.
kHISearchFieldAttributesCancel	Indicates that this view contains a Cancel button.

## 10.3 Symbol Changes

<code>kHISearchFieldNoAttributes</code>	Indicates that this view does not have any attributes.
<code>kHISegmentBehaviorMomentary</code>	Pops back up after being pressed, just like a push button.
<code>kHISegmentBehaviorRadio</code>	Stays pressed until another segment with the radio behavior is pressed. This makes the segment behave like a radio button. After this segment is clicked, the segmented view's value is changed to this segment's one-based index.
<code>kHISegmentBehaviorSticky</code>	After being pressed, this type of segment stays pressed until it is unpressed programmatically. Currently, this constant should not be used; if you use it, you get the same effect as if you used <code>kHISegmentBehaviorMomentary</code> .
<code>kHISegmentBehaviorToggles</code>	Behaves like a check box. When clicked, it toggles back and forth between checked and unchecked states. Currently, this constant should not be used; if you use it, you get the same effect as if you used <code>kHISegmentBehaviorMomentary</code> .
<code>kHISegmentedViewKind</code>	Control kind for a segmented view.
<code>kHISegmentNoAttributes</code>	Indicates no attributes.
<code>kHISegmentSendCmdToUserFocus</code>	If this attribute bit is set, the command that is sent when the segment is clicked will be directed at the user focus instead of up the segmented view's containment hierarchy.
<code>kHIViewAllowsSubviews</code>	
<code>kHIViewAutoToggles</code>	
<code>kHIViewDoesNotDraw</code>	
<code>kHIViewDoesNotUseSpecialParts</code>	
<code>kHIViewGetsFocusOnClick</code>	
<code>kHIViewIdlesWithTimer</code>	
<code>kHIViewIgnoresClicks</code>	
<code>kHIViewInvertsUpDownValueMeaning</code>	
<code>kHIViewIsOpaque</code>	
<code>kHIViewMenuContentID</code>	The standard view ID for the content view of a menu. The Menu Manager assigns this view ID to all menu content views.
<code>kHIViewSupportsGhosting</code>	
<code>kHIViewSupportsLiveFeedback</code>	
<code>kHIViewSupportsRadioBehavior</code>	

kHIViewValidFeaturesForPanther	
ratio	
toView	

## MacApplication.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIAboutBox	Displays a generic, HI-compliant about box.
------------	---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kHIToolboxVersionNumber	The current HIToolbox version number, which is incremented each time that HIToolbox is rebuilt during the course of a Mac OS X release.
kUIModeAllSuppressed	All system UI elements (including the menu bar) are hidden.
kUIOptionDisableHide	This flag disables the Hide menu item in the Application menu.

## MacTextView.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HITextViewCreate	Creates an HITextView that is initially invisible.
HITextViewGetTXNObject	Obtains the text object associated with an HITextView.
TXNAttachObjectToWindowRef	Attaches a text object to a window.
TXNCreateObject	Creates a new MLTE text object which is an opaque structure that handles text formatting at the document level.
TXNDrawObject	Draws a text object in the last window set by your application.
TXNFlattenObjectToCFDataRef	Flattens a text object so it can be saved to disk or embedded with other data.

TXNGetHIRect	Obtains the values for the current view, destination, or text rectangle.
TXNGetWindowRef	Returns a reference to the window to which the specified text object is attached.
TXNHIPointToOffset	Gets the offset value that corresponds to a point in local coordinates.
TXNOffsetToHIPoint	Obtains the local coordinates of the point that corresponds to a specified offset of a text object.
TXNSetDataFromCFURLRef	Replaces a range of data with the contents of a file.
TXNSetHIRectBounds	Sets the view rectangle and/or the destination rectangle.
TXNSetScrollbarState	Sets the state of the scroll bars so they are drawn correctly in response to activate events.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlKindHITextView	The control kind for the HITextView class.
kTXNATSUIStyle	Specifies that the data field of the TXNTypeAttributes structure contains an ATSUI style.
kTXNATSUIStyleContinuousBit	
kTXNATSUIStyleContinuousMask	
kTXNATSUIStyleSize	The size of the ATSUI style attribute.
kTXNDestinationRectKey	Specifies to use the destination rectangle.
kTXNDisabledFunctionalityErr	Indicates the function has been disabled.
kTXNDoFontSubstitutionBit	
kTXNDoFontSubstitutionMask	
kTXNDrawItemAllMask	Used to set all draw-items bits or test to see whether all draw-items bits are set. Setting all bits specifies to draw the scroll bars, text, and the current selection.
kTXNDrawItemScrollbarsBit	Specifies to draw the scroll bars.
kTXNDrawItemScrollbarsMask	Use to set or test for the kTXNDrawItemScrollbarsBit.
kTXNDrawItemTextAndSelectionBit	Specifies to render the text and the current selection.
kTXNDrawItemTextAndSelectionMask	Use to set or text for the kTXNDrawItemTextAndSelectionBit.

kTXNDrawItemTextBit	Specifies to render the text.
kTXNDrawItemTextMask	Used to set or test for the kTXNDrawItemTextBit.
kTXNHorizontalScrollBarRectKey	Specifies to include the horizontal scroll bar in the rectangle.
kTXNRichTextFormatData	
kTXNTextAndMultimediaData	
kTXNTextRectKey	Specifies to use the text rectangle.
kTXNVerticalScrollBarRectKey	Specifies to include the vertical scroll bar in the rectangle.
kTXNViewRectKey	Specifies to use the view rectangle.
TXNDrawItems	Test for draw-items bits.
TXNRectKey	Specifies the bounds to use for a text object.

## MacWindows.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

DetachSheetWindow	Detaches a sheet window from its parent window.
HIWindowChangeClass	Changes the appearance and behavior of a window.
HIWindowChangeFeatures	Changes a window's features.
HIWindowFlush	Flushes any dirty areas a window might have.
HIWindowIsDocumentModalTarget	Determines if a window is currently the target window of another document modal window, such as a sheet.
TransitionWindowWithOptions	Transitions a window from one state to another with appropriate animation and sound.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIWindowRef	Represents a window.
kSimpleWindowClass	



<code>kWindowAsyncDragAttribute</code>	If the bit specified by this mask is set, the window server drags the window automatically.
<code>kWindowDoesNotCycleAttribute</code>	If the bit specified by this mask is set, the window does not participate in window cycling.
<code>kWindowFadeTransitionEffect</code>	Fade the window into or out of visibility. Use with the Show or Hide transition action. (Available in Mac OS X v10.3 and later.)
<code>kWindowGenieTransitionEffect</code>	Use the Genie effect that the Dock uses to minimize or maximize a window to show or hide the window. Use with the Show or Hide transition action. (Available in Mac OS X v10.3 and later.)
<code>TransitionWindowOptions</code>	Defines transition options used when calling <code>TransitionWindowWithOptions</code> .

## Menus.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>CancelMenuTracking</code>	Cancels menu tracking.
---------------------------------	------------------------

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>classID</code>	
<code>initEvent</code>	
<code>kHIMenuAppendItem</code>	Pass to <code>InsertMenuItem</code> , <code>InsertMenuItemText</code> , or <code>InsertMenuItemTextWithCFString</code> to indicate that the new menu item should be added to the end of the menu.
<code>kHIMenuDismissedByActivationChange</code>	The active window changed.
<code>kHIMenuDismissedByAppSwitch</code>	The application with the menu is no longer frontmost.
<code>kHIMenuDismissedByCancelMenuTracking</code>	The application called <code>CancelMenuTracking</code> .
<code>kHIMenuDismissedByFocusChange</code>	The user focus window changed, or the keyboard focus was removed from the current process.
<code>kHIMenuDismissedByKeyEvent</code>	A keyboard event occurred.
<code>kHIMenuDismissedByMouseDown</code>	The user pressed the mouse someplace that did not result in a menu item selection.

<code>kHIMenuDismissedByMouseUp</code>	The user released the mouse someplace that did not result in a menu item selection.
<code>kHIMenuDismissedBySelection</code>	The user selected a menu item.
<code>kHIMenuDismissedByTimeout</code>	The menu tracking mode timed out.
<code>kHIMenuDismissedByUserCancel</code>	The user cancelled menu tracking.
<code>kMenuAttrCondenseSeparators</code>	Hides extra separators to avoid blank spaces in a menu.
<code>kMenuAttrDoNotCacheImage</code>	Disables automatic caching of the menu image.
<code>kMenuAttrDoNotUseUserCommandKeys</code>	Disables substitution of command key equivalents from the <code>NSUserKeysEquivalents</code> dictionary.
<code>kMenuDefClassID</code>	A custom menu definition using an <code>HView</code> subclass.
<code>kMenuItemAttrAutoDisable</code>	Disables the menu item if it does not respond to the <code>kEventCommandUpdateStatus</code> event .
<code>kMenuItemAttrUpdateSingleItem</code>	Update only the menu item that matches when searching available command keys.
<code>view</code>	

## TextServices.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>CopyTextServiceInputModeList</code>	Obtains a copy of the set of input modes supported by a keyboard-class input method.
<code>IsTextServiceSelected</code>	Determines if a text service component is selected.
<code>TSMCopyInputMethodEnabledInputModes</code>	Obtain the array of the enabled (and visible) input modes for a component.
<code>TSMSelectInputMode</code>	Sets the specified input method input mode as the current input source.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kCMCopyTextServiceInputModeList</code>	
<code>kCMTextServiceEventRef</code>	

<code>kOCRInputMethodClass</code>	Specifies a text service class for Optical Character Recognition input methods. Behavior is additive. Similar to Character palette class.
<code>kSpeechInputMethodClass</code>	Specifies a text service class for Speech input methods. Behavior is additive. Similar to Character palette class.
<code>kTextServiceDocumentInterfaceType</code>	A four-character code identifying a TSM document type for traditional (non-Unicode) TSM documents.
<code>kTextServiceInputModePropertyTag</code>	Specifies the input mode property for input methods.
<code>kTextServiceJaTypingMethodPropertyTag</code>	Deprecated. Use the tag <code>kTextServiceInputModePropertyTag</code> instead.
<code>kTSM15Version</code>	Specifies the version of the Text Services Manager is 1.5
<code>kTSM20Version</code>	Specifies the version of the Text Services Manager is 2.0 (Mac OS X v10.0).
<code>kTSM22Version</code>	Specifies the version of the Text Services Manager is 2.2 (Mac OS X 1v0.3).
<code>kTSMDocumentInputModePropertyTag</code>	The property value indicates which input mode should be used by the current keyboard-class input method.
<code>kTSMDocumentPropertyUnicodeInputWindow</code>	You should no longer use this property.
<code>kTSMDocumentRefconPropertyTag</code>	The property value initially contains the refcon value passed to the function <code>NewTSMDocument</code> .
<code>kTSMDocumentSupportDocumentAccessPropertyTag</code>	The presence of this property tag indicates that the event handlers associated with this TSM document support the TSM document access event suite (see “Carbon Events for TSM Document Access.”) This property also indicates that the handler for the <code>TSMEvent kEventTextInputUpdateActiveInputArea</code> supports the <code>replaceRange</code> parameter and that the handler is a Carbon event handler, not an <code>AppleEvent</code> handler.
<code>kTSMDocumentSupportGlyphInfoPropertyTag</code>	The existence of this property in a TSM document indicates that the event handlers associated with the TSM document are aware of the TSM <code>GlyphInfo</code> data structure.

<code>kTSMDocumentTextServicePropertyTag</code>	Specifies a non-Unicode savvy document. This property is equivalent to a pre-existing document interface type.
<code>kTSMDocumentTSMTEPropertyTag</code>	This property is equivalent to the TSMTE document interface type.
<code>kTSMDocumentUnicodeInputWindowPropertyTag</code>	The presence of this property tag indicates that although the TSM document has been told to use the TSM floating input window to handle input from input methods, the floating window is to support Unicode input.
<code>kTSMDocumentUnicodePropertyTag</code>	This property is equivalent to the Unicode document interface type.
<code>kTSMDocumentUseFloatingWindowPropertyTag</code>	The presence of this property tag indicates that the TSM document should use the TSM floating input window to handle input from input methods.
<code>kTSMTEDocumentInterfaceType</code>	Deprecated.
<code>kUnicodeDocumentInterfaceType</code>	Specifies a TSM document type for Unicode-savvy applications. TSM pass through all Unicode text unchanged. When this TSM document is active, the full range of input sources is available to the user, such as Unicode keyboard layouts.
<code>TextServicePropertyTag</code>	Specify a feature or functionality of a component.
<code>TextServicePropertyValue</code>	Defines a data type for text service property values.
<code>TSMDocumentInterfaceType</code>	Defines an interface type for a TSM document.
<code>TSMDocumentPropertyTag</code>	Specify property tags for a TSM document.

## Print

---

### PMPrintingDialogExtensions.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kEventUpdatePrintButton</code>	
--------------------------------------	--

## Ink

---

### Ink.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>InkAddStrokeToCurrentPhrase</code>	Adds a stroke to the current Ink phrase.
<code>InkIsPhraseInProgress</code>	Returns whether Ink Services has initiated and is currently maintaining an Ink phrase whose source is user input.
<code>InkSetApplicationRecognitionMode</code>	Specifies whether Ink input should be interpreted as text, gestures, both, or neither.
<code>InkSetApplicationWritingMode</code>	Controls where the user is allowed to write in the current application.
<code>InkSetDrawingMode</code>	Controls what is drawn when the user writes.
<code>InkSetPhraseTerminationMode</code>	Sets the conditions that define a phrase termination.
<code>InkTerminateCurrentPhrase</code>	Terminates the current phrase.
<code>InkTextAlternatesCount</code>	Returns the number of alternate text interpretations available for an Ink phrase.
<code>InkTextBounds</code>	Returns the bounds of an Ink text object.
<code>InkTextCopy</code>	Copies an existing Ink text object.
<code>InkTextCreateCFString</code>	Obtains the string associated with a text interpretation of an Ink phrase.
<code>InkTextCreateFromCFData</code>	Creates an Ink text object from a previously-flattened Ink text object.
<code>InkTextDraw</code>	Rescales and draws Ink text into the specified bounds.
<code>InkTextFlatten</code>	Flattens an Ink text object for archiving.
<code>InkTextInsertAlternatesInMenu</code>	Inserts a list of alternate text interpretations into a menu.
<code>InkTextKeyModifiers</code>	Returns a value that specifies the key modifiers applied to an Ink phrase.
<code>InkUserWritingMode</code>	Returns the Ink writing mode set by the user in the Ink preferences pane.

**Data Types & Constants**

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

InkAlternateCount	Defines a data type that specifies the number of alternate text interpretations of an Ink phrase.
InkApplicationWritingModeType	
InkDrawingModeType	Specify what Ink Services should draw.
InkGestureKind	Define editing actions.
InkPoint	Contains data that describes an Ink point.
InkPointPtr	
InkRecognitionType	Specify how to interpret Ink input for an application.
InkSourceType	
InkTerminationType	Defines the conditions under which an Ink phrase should be terminated.
InkTextDrawFlagsType	
InkTextRef	Defines a data type for a reference to an opaque Ink text object.
InkUserWritingModeType	Specify the Ink writing mode set by the user in the Ink pane of System Preferences.
keyModifiers	
kInkAlternateCommand	Specifies the menu command ID assigned to menu items inserted by the function <code>InkTextInsertAlternatesInMenu</code> . You can use this constant to determine which menu items in a menu are supplied by Ink Services.
kInkDrawingCommand	Specifies the menu command ID assigned to the menu item containing the ink drawing.
kInkDrawInkAndWritingGuides	Specifies to draw both the Ink and the writing guides. This is the default.
kInkDrawInkOnly	Specifies to draw Ink but not the writing guides.
kInkDrawNothing	Specifies not to draw Ink or the writing guides.
kInkGestureClear	Specifies to clear.
kInkGestureCopy	Specifies to copy.

## 10.3 Symbol Changes

<code>kInkGestureCut</code>	Specifies to cut.
<code>kInkGestureDelete</code>	Specifies to delete. This corresponds to pressing the Delete key.
<code>kInkGestureEscape</code>	This corresponds to pressing the Escape key.
<code>kInkGestureJoin</code>	Specifies to join two words into a single word, eliding the space between them, and may be applied to editable objects other than text.
<code>kInkGestureLeftReturn</code>	Specifies to insert a return (new line) character. The “left” distinction indicates that the gesture is drawn with the small angle-bracket pointing to the left side.
<code>kInkGestureLeftSpace</code>	Specifies to insert a single space character. The “left” distinction indicates that the gesture is drawn with the long, horizontal tail is on the left side.
<code>kInkGesturePaste</code>	Specifies to paste.
<code>kInkGestureRightReturn</code>	Specifies to insert a return (new line) character. The “right” distinction indicates that the gesture is drawn with the small angle-bracket pointing to the right side.
<code>kInkGestureRightSpace</code>	Specifies to insert a single space character. The “right” distinction indicates that the gesture is drawn with the long, horizontal tail is on the right side.
<code>kInkGestureSelectAll</code>	Specifies to select all items in the area that has user focus.
<code>kInkGestureTab</code>	Specifies to insert a tab character.
<code>kInkGestureUndo</code>	Specifies to undo the last action.
<code>kInkRecognitionDefault</code>	Specifies the default setting, which is to interpret Ink input as text or gestures.
<code>kInkRecognitionGesture</code>	Specifies to allow interpretation of Ink input as gestures.
<code>kInkRecognitionNone</code>	Specifies to turn off Ink recognition.
<code>kInkRecognitionText</code>	Specifies to allow interpretation of Ink input as text.
<code>kInkSeparatorCommand</code>	Specifies the menu command ID assigned to the separator item between the alternates and the Ink drawing.
<code>kInkSourceApplication</code>	Specifies the Ink source from the application.
<code>kInkSourceUser</code>	Specifies the Ink source from direct user input.

<code>kInkTerminationAll</code>	Specifies to restore automatic phrase termination by Ink Services.
<code>kInkTerminationNone</code>	Specifies to inhibit automatic phrase termination by Ink Services.
<code>kInkTerminationOutOfProximity</code>	Specifies to terminate a phrase when the user stops writing and lifts the stylus entirely out of the proximity range of the tablet.
<code>kInkTerminationRecognizerHorizontalBreak</code>	Specifies to terminate a phrase when the user leaves a large horizontal space between words (approximately two character widths or more).
<code>kInkTerminationRecognizerVerticalBreak</code>	Specifies to terminate a phrase when the user finishes one line and begins writing on the next.
<code>kInkTerminationTimeOut</code>	Specifies to terminate a phrase when all of the following are true:
<code>kInkTextDrawDefault</code>	Specifies to use the default system settings when drawing. By default, Ink is drawn with pressure sensitive gradients, and the Quartz context settings are overridden for line color and width.
<code>kInkTextDrawHonorContext</code>	Specifies to use the current Quartz context settings for line color and width.
<code>kInkTextDrawIgnorePressure</code>	Specifies not to use pressure sensitive gradients when drawing.
<code>kInkWriteAnywhere</code>	Specifies the user has enabled Ink to allow writing anywhere on the screen. Ink Services flows ink points and recognition results to the frontmost application. This is the default situation when the user enables Ink.
<code>kInkWriteAnywhereInApp</code>	Specifies to allow Ink input anywhere onscreen for your application.
<code>kInkWriteInInkAwareAppsOnly</code>	Specifies the user has enabled Ink only to allow writing in an application that has enabled Ink Services by calling the function <code>InkSetApplicationWritingMode</code> with the <code>kInkWriteAnywhereInApp</code> parameter.
<code>kInkWriteNowhere</code>	Specifies the user has disabled Ink or that Ink Services are not available (for example, a tablet is not attached).
<code>kInkWriteNowhereInApp</code>	Specifies not to allow Ink input in your application.
<code>tabletPointData</code>	



## ImageCapture

---

### ICAApplication.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

ICACopyObjectThumbnail	
ICAImportImage	

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

ICACopyObjectThumbnailPB	
ICAImportImagePB	
importedImages	
kDontEmbedColorSyncProfile	
kICAAllowMultipleImages	
kICAButtonCopy	
kICAButtonEMail	
kICAButtonScan	
kICAButtonWeb	
kICACannotYieldDevice	
kICADatatypeNotFoundErr	
kICADeviceAlreadyOpenErr	
kICADeviceGUIDNotFoundErr	
kICADeviceInternalErr	
kICADeviceInvalidParamErr	
kICADeviceIOServicePathNotFoundErr	
kICADeviceLocationIDNotFoundErr	

### 10.3 Symbol Changes

kICADeviceMemoryAllocationErr	
kICAMessageCameraReadClock	
kICAMessageDeviceYield	
kICAMessageGetLastButtonPressed	
kICAThumbnailFormatICA	
kICAThumbnailFormatJPEG	
kICAThumbnailFormatTIFF	
supportedFileTypes	
thumbnailData	
thumbnailFormat	

# 10.2 Symbol Changes

---

This article lists the symbols added to `Carbon.framework` in Mac OS X v10.2.

## Classes

All of the classes with new symbols are listed alphabetically, with their new class, instance, and delegate methods described.

## C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

### CarbonSound

---

#### Sound.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>bufferFlags</code>	
<code>bufferFlagsMask</code>	
<code>commonFrameSize</code>	
<code>extensionsPtr</code>	
<code>extensionsSize</code>	
<code>frameSizesArray</code>	
<code>kExtendedSoundBufferFlagsValid</code>	
<code>kExtendedSoundBufferIsDiscontinuous</code>	
<code>kExtendedSoundBufferIsFirstBuffer</code>	

kExtendedSoundCommonFrameSizeValid	
kExtendedSoundExtensionsValid	
kExtendedSoundFrameSizesValid	
kHALCustomComponentSubType	
kSoundCodecInfoCanChangeOutputRate	
kSoundCodecInfoFixedCompression	
kSoundCodecInfoHasRestrictedInputRates	
kSoundCodecInfoRequiresExternalFraming	
kSoundCodecInfoResourceType	
kSoundCodecInfoVariableCompression	
siClientAcceptsVBR	
siCompressionConfiguration	
siCompressionInputRateList	
siCompressionMaxPacketSize	
siCompressionOutputSampleRate	
siDecompressionConfiguration	
siHALAudioDeviceUniqueID	
siMediaContextID	
siSourceIsExhausted	

## Sound.k.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

AddSource	
GetDeviceInfo	
GetSource	
GetSourceData	
InitHardware	

## 10.2 Symbol Changes

InitOutputDevice	
PauseRecording	
PauseSource	
PlaySourceBuffer	
ReadSync	
RemoveSource	
ResumeRecording	
SetDeviceInfo	
SetOutput	
SetSource	
StartSource	
StopRecording	
StopSource	

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

uppSndInputGetDeviceInfoProcInfo	
uppSndInputGetStatusProcInfo	
uppSndInputInitHardwareProcInfo	
uppSndInputPauseRecordingProcInfo	
uppSndInputReadAsyncProcInfo	
uppSndInputReadSyncProcInfo	
uppSndInputResumeRecordingProcInfo	
uppSndInputSetDeviceInfoProcInfo	
uppSndInputStopRecordingProcInfo	
uppSoundComponentAddSourceProcInfo	
uppSoundComponentGetInfoProcInfo	
uppSoundComponentGetSourceDataProcInfo	

uppSoundComponentGetSourceProcInfo	
uppSoundComponentInitOutputDeviceProcInfo	
uppSoundComponentPauseSourceProcInfo	
uppSoundComponentPlaySourceBufferProcInfo	
uppSoundComponentRemoveSourceProcInfo	
uppSoundComponentSetInfoProcInfo	
uppSoundComponentSetOutputProcInfo	
uppSoundComponentSetSourceProcInfo	
uppSoundComponentStartSourceProcInfo	
uppSoundComponentStopSourceProcInfo	

## CommonPanels

---

### FontPanel.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

FPIsFontPanelVisible	Checks whether the Fonts panel is visible.
FPShowHideFontPanel	Shows or hides the Fonts panel.
SetFontInfoForSelection	Sets the selections that appear in the Fonts panel.

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

fontPanelFontSelectionQDStyleVersionErr	Your application specified an invalid version in the version field of the FontSelectionQDStyle data structure.
fontPanelSelectionStyleErr	Your application specified an invalid style type in the iStyleType parameter of the SetFontInfoForSelection function.
fontPanelShowErr	The Fonts panel could not be opened.

FontSelectionQDStyle	Contains QuickDraw style font information.
FontSelectionQDStylePtr	A pointer to a data structure that contains QuickDraw style font information.
hasColor	
kEventClassFont	Specifies a Carbon event related to font selection or handling.
kEventFontPanelClosed	Specifies a close event in the Fonts panel. Your application should update the user interface as necessary. For example, you may need to change a Hide Fonts Panel menu command to Show Fonts Panel.
kEventFontSelection	Specifies a selection event in the Fonts panel. Your application can obtain the selections from the event parameters associated with this event. Your application must check for all those parameters it handles and apply the selections to the text.
kEventParamATSUIFontID	Specifies the four-character code associated with an ATSUIFontID selection in the Fonts panel.
kEventParamATSUIFontSize	Specifies the four-character code associated with an ATSUIFontSize selection in the Fonts panel.
kEventParamFMFontFamily	Specifies the four-character code associated with an FMFontFamily selection in the Fonts panel.
kEventParamFMFontSize	Specifies the four-character code associated with an FMFontSize selection in the Fonts panel.
kEventParamFMFontStyle	Specifies the four-character code associated with an FMFontStyle selection in the Fonts panel.
kEventParamFontColor	Specifies the four-character code associated with a font color selection in the Fonts panel.
kFontSelectionATSUIType	Specifies ATSUI data (ATSUIStyle).
kFontSelectionQDStyleVersionZero	Specifies version 0 of a QuickDraw font selection.
kFontSelectionQDType	Specifies QuickDraw data (FontSelectionQDStyle).
kHICommandShowHideFontPanel	Specifies the four-character code for the show/hide Fonts panel command.
typeATSUIFontID	Specifies that an ATSUIFontID is of type UInt32.
typeATSUSize	Specifies that an ATSUSize is of type Fixed.
typeFMFontFamily	Specifies that an FMFontFamily is of type SInt16.

<code>typeFMFontSize</code>	Specifies that an <code>FMFontSize</code> is of type <code>Slnt16</code> .
<code>typeFMFontStyle</code>	Specifies that an <code>FMFontStyle</code> is of type <code>Slnt16</code> .
<code>typeFontColor</code>	Specifies that a font color is of type <code>RGBColor</code> .

## SecurityHI

---

### SecCertificateSupport.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>SecChooseIdentity</code>	
<code>SecChooseIdentityAsSheet</code>	
<code>SecDisplayCertificate</code>	
<code>SecDisplayCertificateGroup</code>	
<code>SecEditTrust</code>	
<code>SecEditTrustAsSheet</code>	

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>typeSecIdentityRef</code>	
---------------------------------	--

## OpenScripting

---

### DigitalHubRegistry.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kDigiHubBlankCD</code>	
<code>kDigiHubBlankDVD</code>	



kDigiHubEventClass	
kDigiHubMusicCD	
kDigiHubPictureCD	
kDigiHubVideoDVD	

## HI Toolbox

---

### Appearance.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kCenterOnScreen	The picture is its actual size, or clipped if necessary, with the desktop pattern showing to the side of the picture if it is smaller than the desktop
kFillScreen	The picture's aspect ratio is altered if necessary.
kFitToScreen	The picture is reduced if necessary.
kThemeBrushMenuBackground	
kThemeBrushMenuBackgroundSelected	
kThemeBrushPrimaryHighlightColor	
kThemeBrushSecondaryHighlightColor	
kThemeDrawerWindow	
kThemeNotAllowedCursor	The cursor identified by this constant is typically used to indicate that the current action is not allowed. For example, you could use this cursor to indicate that an object being dragged cannot be dropped at the current mouse position.
kThemeSystemFontDetail	
kThemeSystemFontDetailEmphasized	
kThemeTextColorSystemDetail	
kThemeToolbarFont	The font used to draw the label of a toolbar item.
kThemeTrackHasFocus	If the bit specified by this mask is set, the thumb has focus. This attribute currently has effect only on sliders.

kTiledOnScreen	The picture draws repeatedly.
kUseBestGuess	The picture is automatically positioned, based on picture and monitor sizes.

## CarbonEvents.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

AXNotificationHIOBJECTNotify	Posts a notification for an accessibility object.
AXUIElementCreateWithHIOBJECTAndIdentifier	Creates an accessibility object that represents a user interface element.
AXUIElementGetHIOBJECT	Gets the user interface element the given accessibility object represents.
AXUIElementGetIdentifier	Gets the unique identifier associated with an accessibility object.
ChangeMouseTrackingRegion	
ClipMouseTrackingRegion	
ClipWindowMouseTrackingRegions	
CreateMouseTrackingRegion	Creates a mouse tracking region.
GetMouseTrackingRegionID	
GetMouseTrackingRegionRefCon	Obtains the reference constant for a mouse tracking region.
MoveMouseTrackingRegion	
MoveWindowMouseTrackingRegions	
ReleaseMouseTrackingRegion	Releases a mouse tracking region.
ReleaseWindowMouseTrackingRegions	
RetainMouseTrackingRegion	Retains a mouse tracking region.
SetMouseTrackingRegionEnabled	
SetWindowMouseTrackingRegionsEnabled	

**Data Types & Constants**

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HICCommandExtended</code>	Represents an extended command event.
<code>kEventAccessibleGetAllActionNames</code>	Sent by an assistive application to find out which actions the given accessibility object supports.
<code>kEventAccessibleGetAllAttributeNames</code>	A request sent by an assistive application to get the names of all attributes the given accessibility object supports.
<code>kEventAccessibleGetChildAtPoint</code>	A request sent by an assistive application to get the accessible child of the given accessibility object that contains the given point.
<code>kEventAccessibleGetFocusedChild</code>	A request sent by an assistive application to get the accessible child of the given accessibility object that is part of the focus chain.
<code>kEventAccessibleGetNamedActionDescription</code>	Sent by an assistive application to get the human-intelligible name of the given action.
<code>kEventAccessibleGetNamedAttribute</code>	A request sent by an assistive application to get the value of the given attribute.
<code>kEventAccessibleIsNamedAttributeSettable</code>	A request sent by an assistive application to find out if the given attribute's value can be changed.
<code>kEventAccessiblePerformNamedAction</code>	Sent by an assistive application when it wants the given accessibility object to perform the given action.
<code>kEventAccessibleSetNamedAttribute</code>	A request sent by an assistive application to set the value of the given attribute to the passed-in value.
<code>kEventAppFocusMenuBar</code>	Request to switch the keyboard focus to the menu bar. The Carbon Event Manager handles this event by default.
<code>kEventAppFocusNextDocumentWindow</code>	Request to shift keyboard focus to the next or previous document window (depending on the state of the Shift key).
<code>kEventAppFocusNextFloatingWindow</code>	Request to shift keyboard focus to the next or previous floating window (depending on the state of the Shift key).
<code>kEventAppFocusToolbar</code>	Request to shift keyboard focus to the toolbar.
<code>kEventAppHidden</code>	The application was hidden.

<code>kEventAppShown</code>	The application was shown.
<code>kEventAppSystemUIModeChanged</code>	The system user interface mode of the frontmost application has changed.
<code>kEventClassAccessibility</code>	Events related to application accessibility features.
<code>kEventClassToolbar</code>	Events related to the toolbar (not the toolbar window class).
<code>kEventClassToolbarItem</code>	Events related to toolbar items.
<code>kEventControlDragEnter</code>	Sent when a drag item enters a view's bounds.
<code>kEventControlDragLeave</code>	Sent when a drag item leaves your view. You can use this event to unhighlight your view, and so on. (Available in Mac OS X v10.2 and later.)
<code>kEventControlDragReceive</code>	Sent when a drag item is dropped within your view.
<code>kEventControlDragWithin</code>	Sent when a drag item has moved while in the view's bounds (but not within any of its subviews).
<code>kEventControlEnabledStateChanged</code>	Sent when a control's enabled state changes (that is, when a control is enabled or disabled).
<code>kEventControlGetAutoToggleValue</code>	Sent when the system wants to auto-toggle a control.
<code>kEventControlGetClickActivation</code>	Sent when a mouse click occurs in a background (inactive) control.
<code>kEventControlGetNextFocusCandidate</code>	Sent to allow a control to customize the focus order of its subcontrols.
<code>kEventControlGetSizeConstraints</code>	Sent when the <code>HViewGetSizeConstraints</code> function is called. You use this to let your custom view indicate its maximum and minimum size. A parent view can use this information to help it lay out subviews.
<code>kEventControlGetSubviewForMouseEvent</code>	
<code>kEventControlHighlightChanged</code>	Sent when a control's highlight state changes.
<code>kEventControlInterceptSubviewClick</code>	Sent when the <code>HViewGetViewForMouseEventClick</code> function is called (typically by the Control Manager before it descends into subviews).
<code>kEventControlTitleChanged</code>	Sent when your control's title changes.
<code>kEventMouseEntered</code>	The mouse entered a tracking region. Used with mouse tracking regions. See <code>CreateMouseTrackingRegion</code> for more information.

kEventMouseExited	The mouse left a tracking region. Used with mouse tracking regions. See <code>CreateMouseTrackingRegion</code> for more information.
kEventParamAccessibleActionDescription	Specifies the description of an action. The parameter type is <code>typeCFMutableStringRef</code> .
kEventParamAccessibleActionName	Specifies an action name. The parameter type is <code>typeCFStringRef</code> .
kEventParamAccessibleActionNames	Specifies an array of action names (each of type <code>CFStringRef</code> ) . The parameter type is <code>typeCFMutableArrayRef</code> .
kEventParamAccessibleAttributeName	Specifies an attribute name. The parameter type is <code>typeCFStringRef</code> .
kEventParamAccessibleAttributeNames	Specifies an array of attribute names (each of type <code>CFStringRef</code> ). The parameter type is <code>typeCFMutableArrayRef</code> .
kEventParamAccessibleAttributeSettable	Specifies whether an attribute is settable. The parameter type is <code>typeBoolean</code> .
kEventParamAccessibleAttributeValue	Specifies the value of an attribute.
kEventParamAccessibleChild	Specifies the child accessibility object. The parameter type is <code>typeCFTyperef</code> , and the data must be of type <code>AXUIElementRef</code> .
kEventParamAccessibleObject	Specifies an accessibility object. The parameter type is <code>typeCFTyperef</code> , and the data must be of type <code>AXUIElementRef</code> .
kEventParamControlCurrentPart	<code>typeControlPartCode</code>
kEventParamControlFocusEverything	<code>typeBoolean</code>
kEventParamControlInvalidRgn	<code>typeQDRgnHandle</code>
kEventParamControlPreviousPart	<code>typeControlPartCode</code>
kEventParamControlSubview	<code>typeControlRef</code>
kEventParamControlValue	<code>typeLongInteger</code>
kEventParamCurrentDockRect	<code>typeHIRect</code>
kEventParamMaximumSize	<code>typeHISize</code>
kEventParamMinimumSize	<code>typeHISize</code>
kEventParamMouseTrackingRef	
kEventParamMutableArray	<code>typeCFMutableArrayRef</code>

kEventParamNextControl	typeControlRef
kEventParamPostTarget	Specifies the target the event should be sent to. Instead of sending an event directly to a given target, you can set this parameter and post the event onto the event queue.
kEventParamPreviousDockRect	typeHIRect
kEventParamResult	Any type, depending on the event
kEventParamStartControl	typeControlRef
kEventParamSystemUIMode	typeUInt32
kEventParamTextInputGlyphInfoArray	
kEventParamToolbar	typeHIToolbarRef
kEventParamToolbarItem	typeHIToolbarItemRef
kEventParamToolbarItemConfigData	typeCFStringRef
kEventParamToolbarItemIdentifier	typeHIToolbarRef
kEventParamWindowMouseLocation	
kEventTextInputUnicodeText	Produced only by input methods or other text services and is delivered to the Text Services Manager by SendTextInputEvent.
kEventWindowCollapsing	The window is collapsing.
kEventWindowDrawerClosed	Sent to the drawer and its parent when the drawer has fully closed.
kEventWindowDrawerClosing	Sent to the drawer and its parent window when the drawer is closing. If you don't want the drawer to close, your handler should return <code>userCanceledErr</code> .
kEventWindowDrawerOpened	Sent to the drawer and its parent window when the drawer has fully opened.
kEventWindowDrawerOpening	Sent to the drawer and its parent window when the drawer is opening. If you don't want the drawer to open, your handler should return <code>userCanceledErr</code> .
kEventWindowExpanding	The window is expanding.
kEventWindowFocusContent	Focus should be shifted to the main content area of your window.
kEventWindowFocusToolbar	Focus should be shifted to the window's toolbar.

<code>kEventWindowGetDockTileMenu</code>	Sent when a dock tile wants to display a menu.
<code>kHICommandFromControl</code>	The command event originated from a control. This bit was introduced in Mac OS X v10.2 and CarbonLib 1.6; it is never set in earlier versions of Mac OS X or CarbonLib.
<code>kHICommandFromWindow</code>	The command event originated from a window. This bit was introduced in Mac OS X v10.2 and CarbonLib 1.6; it is never set in earlier versions of Mac OS X or CarbonLib.
<code>kHICommandRotateFloatingWindowsBackward</code>	The floating window focus hot key (ctl-F6 by default) has been pressed, and floating windows should be rotated so that the window before the focused window is activated.
<code>kHICommandRotateFloatingWindowsForward</code>	The floating window focus hot key (ctl-F6 by default) has been pressed, and floating windows should be rotated so that the window after the focused window is activated.
<code>kHICommandRotateWindowsBackward</code>	The Rotate Windows hot key (cmd-~ by default) has been pressed.
<code>kHICommandRotateWindowsForward</code>	The Rotate Windows hot key (cmd-~ by default) has been pressed.
<code>kMenuContextCommandIDSearch</code>	Sent while trying to match a command ID using the Menu Manager function <code>CountMenuItemsWithCommandID</code> or <code>GetIndMenuItemWithCommandID</code> .
<code>kMouseTrackingOptionsGlobalClip</code>	The region passed to <code>CreateMouseTrackingRegion</code> is defined in global coordinates and that the region is clipped to the owning window's structure region.
<code>kMouseTrackingOptionsLocalClip</code>	The region passed to <code>CreateMouseTrackingRegion</code> is defined in local coordinates, and that the region is clipped to the owning window's content region.
<code>kMouseTrackingOptionsStandard</code>	Same as <code>kMouseTrackingOptionsLocalClip</code> .
<code>kWindowBoundsChangeZoom</code>	The bounds are changing as a result of the user clicking the zoom button.
<code>MouseTrackingOptions</code>	Define constants used by the <code>CreateMouseTrackingRegion</code> function.
<code>MouseTrackingRef</code>	Represents a mouse tracking region
<code>MouseTrackingRegionID</code>	Represents a mouse tracking region identifier.
<code>typeCFIndex</code>	<code>CFIndex</code>

typeCFMutableStringRef	A Core Foundation mutable string.
typeCFTyperef	A Core Foundation type.
typeEventTargetRef	The parameter type for kEventParamPostTarget.
typeHIRect	HIRect
typeHISize	HISize
typeHIToolbarItemRef	HIToolbarItemRef
typeHIToolbarRef	HIToolbarRef
typeMouseTrackingRef	
typeVoidPtr	Void

## CarbonEventsCore.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

DisposeEventLoopIdleTimerUPP	Disposes of an event loop idle timer.
GetCurrentButtonState	
GetCurrentEvent	
GetCurrentEventButtonState	
GetCurrentEventKeyModifiers	Obtains the queue-synchronized keyboard modifier state.
InstallEventLoopIdleTimer	Installs a timer that fires only when there is no user activity.
InstallIdleTimer	
InvokeEventLoopIdleTimerUPP	Calls an event loop idle timer through a UPP.
NewEventLoopIdleTimerUPP	Creates an event loop idle timer UPP.
SendEventToEventTargetWithOptions	Sends an event to the specified event target with propagation options.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.



EventLoopIdleTimerMessage	Represents an idle timer message.
inState	
kEventLoopIdleTimerIdling	The idle period is continuing.
kEventLoopIdleTimerStarted	The idle period has just begun (and this is the first time your callback is being called for this idle period).
kEventLoopIdleTimerStopped	The idle period has just stopped (a user event occurred). Your callback should do any necessary cleanup of the idle process now that a user event has occurred.
kEventTargetDontPropagate	Do not propagate this event to any other event target.
kEventTargetSendToAllHandlers	Send this event to all event targets in the handler chain, regardless of any handler's return value.

## ControlDefinitions.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

GetDataBrowserListViewHeaderDesc	Obtains a header description for a column in list view.
SetDataBrowserListViewHeaderDesc	Provides a description for a column title in list view.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlEditTextSingleLineTag	Indicates whether the control should always be single line.
-------------------------------	---

## Drag.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

GetStandardDropLocation	Gets the standard drop location set by the receiver of a drag.
SetDragImageWithCGImage	Associates a Core Graphics image with a drag reference.
SetStandardDropLocation	Used by the receiver of a drag to set the standard drop location for a drag.

**Data Types & Constants**

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIRect	Defines the position and size of a rectangle using floating-point coordinates.
HISize	Defines the width and height of an object using floating-point coordinates.
kDragStandardDropLocationTrash	Set when a drag is dropped on the trash icon. Setting this standard drop location automatically sets the traditional drop location to an alias to the trash folder.
kDragStandardDropLocationUnknown	The receiver did not specify a drop location. This is the default.
kFlavorTypeUnicodeClippingFilename	The flavor of the unicode name of a clipping file.
kFlavorTypeUnicodeClippingName	The flavor of a hint for the unicode name of a clipping file. This flavor type is preferred over the kFlavorTypeUnicodeClippingFilename type.
StandardDropLocation	Define common drop locations.

**HIObject.h****Functions**

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIObjectCopyClassID	Obtains the class ID of a given HIObject.
HIObjectCreate	Creates an object derived from HIObject.
HIObjectCreateFromBundle	Obtains the HIObject for the given bundle.
HIObjectDynamicCast	Obtains the instance data for a specific class of an HIObject.
HIObjectGetEventTarget	Obtains the event target of an HIObjectRef.
HIObjectIsAccessibilityIgnored	Reports whether the given HIObject is marked as ignored for accessibility.
HIObjectIsOfClass	Determines whether an object is of a certain class.
HIObjectPrintDebugInfo	Prints the internal information of an HIObject for debugging purposes.
HIObjectRegisterSubclass	Registers an HIObject subclass.

<code>HIObjectSetAccessibilityIgnored</code>	Marks an <code>HIObject</code> as ignored (or not) for the purposes of the accessibility APIs.
<code>HIObjectUnregisterClass</code>	Unregisters a previously registered subclass of <code>HIObject</code> .

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>hiObjectClassExistsErr</code>	You are trying to register a class ID that already exists.
<code>hiObjectClassHasInstancesErr</code>	You are trying to unregister a class which has instances that still exist.
<code>hiObjectClassHasSubclassesErr</code>	You are trying to unregister a class which has subclasses registered. They must be unregistered before this class can be unregistered.
<code>hiObjectClassIsAbstractErr</code>	You are trying to create an <code>HIObject</code> class that is defined as being abstract. You must subclass it instead.
<code>HIObjectClassRef</code>	
<code>HIObjectRef</code>	
<code>kEventClassHIObject</code>	The event class for <code>HIObject</code> events.
<code>kEventHIObjectConstruct</code>	Your object is being constructed.
<code>kEventHIObjectDestruct</code>	Your object is being destroyed. This is your chance to dispose of anything you might have allocated for your object. Do not call through with <code>CallNextEventHandler</code> . (Available in Mac OS X v10.2 and later.)
<code>kEventHIObjectInitialize</code>	Your object is being initialized.
<code>kEventHIObjectIsEqual</code>	<code>HIObjectIsEqual</code> has been called, and you are being asked to determine whether your object is equivalent to the one being passed to your handler.
<code>kEventHIObjectPrintDebugInfo</code>	<code>HIObjectPrintDebugInfo</code> has been called, and you are being asked to print your information to <code>stdout</code> . This event is sent to all handlers and you should not call <code>CallNextEventHandler</code> . (Available in Mac OS X v10.2 and later.)
<code>kEventParamHIObjectInstance</code>	On entry, the <code>HIObject</code> reference for your object.
<code>kHIObjectCFArrayCallbacks</code>	
<code>kHIObjectCFDictKeyCallbacks</code>	
<code>kHIObjectCFDictValueCallbacks</code>	

typeHIObjectRef	
-----------------	--

## HIShape.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIShapeContainsPoint	Determines whether a shape contains the given point.
HIShapeCreateCopy	Creates an immutable copy of a mutable or immutable shape.
HIShapeCreateDifference	Creates an immutable shape that is the difference of two shapes.
HIShapeCreateIntersection	Creates an immutable shape that is the intersection of two shapes.
HIShapeCreateMutable	Creates an empty mutable shape.
HIShapeCreateMutableCopy	Creates a mutable copy of an existing shape.
HIShapeCreateUnion	Creates an immutable shape that is the union of two shapes.
HIShapeCreateWithQDRgn	Creates an immutable shape from a QuickDraw region.
HIShapeCreateWithRect	Creates an immutable shape from a Quartz rectangle.
HIShapeDifference	Sets a shape to be the difference of two other shapes.
HIShapeGetAsQDRgn	Creates a QuickDraw region with the same shape as the specified HIShape.
HIShapeGetBounds	Obtains the bounding rectangle for the given shape.
HIShapeGetTypeID	Obtains the Core Foundation type ID for the HIShape class.
HIShapeIntersect	Sets a shape to be the intersection of two other shapes.
HIShapeIsEmpty	Determines whether a given shape is empty.
HIShapeIsRectangular	Determines whether a given shape is rectangular.
HIShapeOffset	Moves a shape by a specified offset.
HIShapeReplacePathInCGContext	Make the current path in a graphics context match a shape.
HIShapeSetEmpty	Sets the specified mutable shape to be empty.
HIShapeSetQDClip	Sets the clip region in the current QuickDraw port to be the specified shape.
HIShapeUnion	Sets a shape to be the union of two other shapes.

**Data Types & Constants**

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIMutableShapeRef	Defines a mutable shape object.
HIShapeRef	Defines a shape object.

**HIToolbar.h**

---

**Functions**

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIToolbarAppendItem	Appends an item to the toolbar.
HIToolbarChangeAttributes	Changes the attributes of a toolbar.
HIToolbarCopyIdentifier	Obtains the identifier for a toolbar.
HIToolbarCopyItems	Obtains the array of toolbar items for a toolbar.
HIToolbarCreate	Creates a toolbar.
HIToolbarCreateItemWithIdentifier	Creates a toolbar item.
HIToolbarGetAttributes	Obtains the attributes for the given toolbar.
HIToolbarGetDelegate	Returns the current delegate in use by a toolbar.
HIToolbarGetDisplayMode	Obtains the current display mode of a toolbar.
HIToolbarGetDisplaySize	Obtains the current display size of a toolbar.
HIToolbarInsertItemAtIndex	Inserts a toolbar item at a given index into a toolbar.
HIToolbarItemChangeAttributes	Changes the attributes of a toolbar item.
HIToolbarItemCopyHelpText	Obtains the help tag text for a toolbar item.
HIToolbarItemCopyIdentifier	Obtains the identifier for a given toolbar item.
HIToolbarItemCopyImage	Obtains the image for a toolbar item.
HIToolbarItemCopyLabel	Obtains the label for a toolbar item.
HIToolbarItemCopyMenu	Obtains the submenu for a toolbar item.
HIToolbarItemCreate	Creates a toolbar item.
HIToolbarItemGetAttributes	Obtains the attributes of a toolbar item.

<code>HIToolbarItemGetCommandID</code>	Gets the command ID of a toolbar item.
<code>HIToolbarItemGetToolbar</code>	Obtains the toolbar associated with a toolbar item.
<code>HIToolbarItemIsEnabled</code>	Determines if a toolbar item is enabled.
<code>HIToolbarItemSetCommandID</code>	Sets the command ID of a toolbar item.
<code>HIToolbarItemSetEnabled</code>	Enables or disables a toolbar item.
<code>HIToolbarItemSetHelpText</code>	Sets the help tag text for a toolbar item.
<code>HIToolbarItemSetIconRef</code>	Sets the icon for a toolbar item.
<code>HIToolbarItemSetImage</code>	Sets the image for a toolbar item.
<code>HIToolbarItemSetLabel</code>	Sets the label of a toolbar item.
<code>HIToolbarItemSetMenu</code>	Sets the submenu for a toolbar item.
<code>HIToolbarRemoveItemAtIndex</code>	Removes an item at a given index from a toolbar.
<code>HIToolbarSetDelegate</code>	Sets the delegate object for a toolbar.
<code>HIToolbarSetDisplayMode</code>	Sets the current display mode of a toolbar.
<code>HIToolbarSetDisplaySize</code>	Sets the current display size of a toolbar.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HIToolbarDisplayMode</code>	
<code>HIToolbarDisplaySize</code>	
<code>HIToolbarItemRef</code>	
<code>HIToolbarRef</code>	
<code>kEventToolbarCreateItemFromDrag</code>	This event is sent to the delegate to when the toolbar needs to create an item from a drag.
<code>kEventToolbarCreateItemWithIdentifier</code>	This event is sent to the delegate to when the toolbar needs to create an item from an identifier. (
<code>kEventToolbarGetAllowedIdentifiers</code>	This event is sent to the delegate to get a list of all the items which could possibly be added to the toolbar. This is sent out when the configuration sheet is about to be displayed. You are passed a mutable array to fill in with the identifiers. (

<code>kEventToolbarGetDefaultIdentifiers</code>	This event is sent to the delegate to get a list of all of the default item identifiers that should be created for a toolbar. You are passed a mutable array to fill in with the identifiers. (
<code>kEventToolbarItemCommandIDChanged</code>	This event is sent to the item when the command ID changes. Any interested parties can install handlers on the toolbar item to receive notifications. (Available in Mac OS X v10.2 and later.)
<code>kEventToolbarItemCreateCustomView</code>	This event is sent to the toolbar item when it is time to create a view for it to display its contents.
<code>kEventToolbarItemEnabledStateChanged</code>	This event is sent to the item when the enabled state changes. Any interested parties can install handlers on the toolbar item to receive notifications. (Available in Mac OS X v10.3 and later.)
<code>kEventToolbarItemGetPersistentData</code>	This event is sent to the item when the toolbar is going to write out the configuration information for the item.
<code>kEventToolbarItemHelpTextChanged</code>	This event is sent to the item when the help text changes. Any interested parties can install handlers on the toolbar item to receive notifications. (Available in Mac OS X v10.2 and later.)
<code>kEventToolbarItemImageChanged</code>	This event is sent to the item when the image changes. Any interested parties can install handlers on the toolbar item to receive notifications. (Available in Mac OS X v10.2 and later.)
<code>kEventToolbarItemLabelChanged</code>	This event is sent to the item when the label changes. Any interested parties can install handlers on the toolbar item to receive notifications. (Available in Mac OS X v10.2 and later.)
<code>kEventToolbarItemPerformAction</code>	This event is sent when a toolbar item is clicked.
<code>kHICommandCustomizeToolbar</code>	When sent to a window with a toolbar, this command causes the configuration sheet to appear.
<code>kHICommandHideToolbar</code>	Sending this command causes a window's toolbar to be hidden. You can set a menu item's command to this ID and it will be handled and updated automatically for you. (Available in Mac OS X v10.2 and later.)
<code>kHICommandShowToolbar</code>	Sending this command causes a window's toolbar to be shown. You can set a menu item's command to this ID and it will be handled and updated automatically for you. (Available in Mac OS X v10.2 and later.)
<code>kHIToolbarAutoSavesConfig</code>	Pass this attribute to allow the toolbar to save its configuration automatically to your application's preferences.

<code>kHIToolbarDisplayModeDefault</code>	Use the default display mode. Currently, this is defined as being both icon and label, but could change in the future.
<code>kHIToolbarDisplayModeIconAndLabel</code>	Display the image as well as the label of the toolbar items.
<code>kHIToolbarDisplayModeIconOnly</code>	Display only the image.
<code>kHIToolbarDisplayModeLabelOnly</code>	Display only the label.
<code>kHIToolbarDisplaySizeDefault</code>	This indicates to use the default display size. Currently, this is defined as using 32 x 32 icons (“normal” size).
<code>kHIToolbarDisplaySizeNormal</code>	This size uses a larger text and icon size.
<code>kHIToolbarDisplaySizeSmall</code>	This size uses a smaller text and icon size.
<code>kHIToolbarIsConfigurable</code>	This attribute indicates that the toolbar is configurable, that is, the user can drag items around and bring up the configuration palette, etc.
<code>kHIToolbarItemAllowDuplicates</code>	This indicates that an item can have more than one instance of itself in the toolbar. If this is not set, only one can be present. By default, the determining factor for what determines if two items are identical is the toolbar identifier.
<code>kHIToolbarItemAnchoredLeft</code>	This item cannot be moved at all by the user.
<code>kHIToolbarItemCantBeRemoved</code>	This item can be rearranged, but it cannot be removed from the toolbar by the user.
<code>kHIToolbarItemIsSeparator</code>	This indicates the item acts as a separator.
<code>kHIToolbarItemMutableAttrs</code>	The set of toolbar item attributes that can be changed by <code>HIToolbarItemChangeAttributes</code> and <code>HIToolbarItemChangeAttributesInWindow</code> . Any other attribute must be specified when it is created.
<code>kHIToolbarItemNoAttributes</code>	Pass this to indicate no attributes at all.
<code>kHIToolbarItemSendCmdToUserFocus</code>	If this attribute bit is set, the command that gets sent out will be directed at the user focus instead of at the window the toolbar is attached to.
<code>kHIToolbarItemValidAttrs</code>	The set of all valid toolbar item attributes.
<code>kHIToolbarNoAttributes</code>	Pass this to indicate no attributes at all.
<code>kHIToolbarValidAttrs</code>	The set of all valid toolbar attributes.



## HIView.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HIComboBoxAppendTextItem	Appends a text item to the combo box disclosure list.
HIComboBoxChangeAttributes	Changes the attributes of a combo box.
HIComboBoxCopyTextItemAtIndex	Copy a text item from a combo box disclosure list
HIComboBoxCreate	Creates a combo box control.
HIComboBoxGetAttributes	Gets the attributes of a combo box.
HIComboBoxGetItemCount	Gets the number of items in the combo box disclosure list.
HIComboBoxInsertTextItemAtIndex	Inserts a CFString in a combo box disclosure list.
HIComboBoxRemoveItemAtIndex	Removes an item from a combo box disclosure list.
HIGrowBoxViewIsTransparent	Determines whether a grow box view is transparent.
HIGrowBoxViewSetTransparent	Makes a grow box view transparent or opaque.
HIImageViewCopyImage	Obtains the image for an image view.
HIImageViewCreate	Creates an image view.
HIImageViewGetAlpha	Obtains the alpha value for a view.
HIImageViewGetScaleToFit	Determines whether an image will scale or clip to the view bounds.
HIImageViewIsOpaque	Determines whether an image view is opaque.
HIImageViewSetAlpha	Sets the alpha value for an image view.
HIImageViewSetImage	Sets the image to display in an image view.
HIImageViewSetOpaque	Sets the opacity of an image view.
HIImageViewSetScaleToFit	Specifies whether an image should scale or clip to the view's bounds.
HIScrollViewCreate	Creates a scroll view.
HIScrollViewGetScrollBarAutoHide	Obtains current setting of a scroll view's scroll bar auto-hide setting.
HIScrollViewSetScrollBarAutoHide	Sets a scroll view's auto-hide setting.
HIViewAddSubview	Adds a subview to the given parent view.

## 10.2 Symbol Changes

<code>HIViewAdvanceFocus</code>	Advances the keyboard focus to the next most appropriate view.
<code>HIViewChangeAttributes</code>	Changes the attributes of a view.
<code>HIViewClick</code>	Passes a mouse-down event to a view.
<code>HIViewConvertPoint</code>	Converts a point's coordinates from one view to another.
<code>HIViewConvertRect</code>	Converts a rectangle from one view to another.
<code>HIViewConvertRegion</code>	Converts a region from one view to another.
<code>HIViewCreateOffscreenImage</code>	Creates a Core Graphics offscreen image of a view.
<code>HIViewDrawCGImage</code>	Draws a Core Graphics image appropriately for a view.
<code>HIViewFindByID</code>	Obtains a view by its ID.
<code>HIViewFlashDirtyArea</code>	Flashes a window's dirty area.
<code>HIViewGetAttributes</code>	Obtains the attributes for a view.
<code>HIViewGetBounds</code>	Obtains the local bounds of a view.
<code>HIViewGetFirstSubview</code>	Returns the first subview of a parent view.
<code>HIViewGetFocusPart</code>	Obtains the part in the specified view that currently has focus.
<code>HIViewGetFrame</code>	Obtains the frame bounds of a view.
<code>HIViewGetLastSubview</code>	Returns the last subview in a parent view.
<code>HIViewGetNeedsDisplay</code>	Determines whether a view needs to be redrawn.
<code>HIViewGetNextView</code>	Returns the view behind the specified view.
<code>HIViewGetPartHit</code>	Determines the part hit for a given point.
<code>HIViewGetPreviousView</code>	Returns the view above the specified view.
<code>HIViewGetRoot</code>	Obtains the root view for a window.
<code>HIViewGetSizeConstraints</code>	Returns the minimum and maximum size for a control.
<code>HIViewGetSubviewHit</code>	Returns the child of the given view hit by the point passed in.
<code>HIViewGetSuperview</code>	Returns a view's parent view.
<code>HIViewGetViewForMouseEvent</code>	Returns the appropriate view to handle a mouse event.
<code>HIViewIsDrawingEnabled</code>	Determines if drawing is currently enabled for a view.
<code>HIViewIsVisible</code>	Determines whether a view is visible.

<code>HIViewMoveBy</code>	Move a view by the specified distance relative to its current location.
<code>HIViewPlaceInSuperviewAt</code>	Places a view at an absolute location within its parent.
<code>HIViewRemoveFromSuperview</code>	Removes a view from its parent.
<code>HIViewReshapeStructure</code>	Informs the system that the structure region of the given view has changed shape.
<code>HIViewScrollRect</code>	Scrolls a view's contents, or a portion thereof.
<code>HIViewSetBoundsOrigin</code>	Sets the origin of the view.
<code>HIViewSetDrawingEnabled</code>	Turns control drawing on or off.
<code>HIViewSetFirstSubviewFocus</code>	Sets the subview that is first to receive keyboard focus.
<code>HIView setFrame</code>	Sets the frame of a view.
<code>HIViewSetNeedsDisplay</code>	Marks a view as needing or not needing to be redrawn.
<code>HIViewSetNeedsDisplayInRegion</code>	Uses a region to mark a portion of a view as needing or not needing to be redrawn.
<code>HIViewSetNextFocus</code>	Sets the view that is to receive keyboard focus when keyboard focus advances from the specified view.
<code>HIViewSetVisible</code>	Hides or shows a view.
<code>HIViewSetZOrder</code>	Changes the front-to-back ordering of sibling views.
<code>HIViewSimulateClick</code>	Simulates a mouse click on a given view.
<code>HIViewSubtreeContainsFocus</code>	Determines whether a view or any subviews have keyboard focus.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>HIViewID</code>	Defines the HIView ID.
<code>HIViewPartCode</code>	Specify view parts constants.
<code>HIViewRef</code>	Define an HIView reference.
<code>HIViewZOrderOp</code>	Specify constants that set a view's z-order.
<code>kControlKindHIComboBox</code>	Control kind for a combo box.
<code>kEventClassScrollable</code>	Event class for events related to a scrollable view.
<code>kEventParamImageSize</code>	A value of type <code>typeHISize</code> representing the image size.

<code>kEventParamLineSize</code>	A value of type <code>typeHISize</code> representing the line size. The line size is the distance that the <code>HIScrollView</code> should scroll its subview when the user clicks a scroll bar arrow. For example, this might be 10 pixels vertically and 20 pixels horizontally.
<code>kEventParamOrigin</code>	A value of type <code>typeHIPoint</code> representing the origin.
<code>kEventParamViewSize</code>	A value of type <code>typeHISize</code> representing the view size. The view size is the current size of the scrollable view. Typically, this is the same as the view's bounds and can be acquired by calling <code>HIViewGetBounds</code> .
<code>kEventScrollableGetInfo</code>	This event is sent by an <code>HIScrollView</code> to its scrollable view to determine the current size and origin of the scrollable view.
<code>kEventScrollableInfoChanged</code>	This event is not sent by an <code>HIScrollView</code> itself.
<code>kEventScrollableScrollTo</code>	This event is sent by an <code>HIScrollView</code> to its scrollable view to request that the scrollable view update its current origin and redraw.
<code>kHIComboBoxAutoCompletionAttribute</code>	The control will attempt to auto complete the text the user is typing with an item in the combo box list that is the closest appropriate match.
<code>kHIComboBoxAutoDisclosureAttribute</code>	The control will disclose the combo box list after the user enters text.
<code>kHIComboBoxAutoSizeListAttribute</code>	The combo box list will be automatically sized to fit the Human Interface Guidelines.
<code>kHIComboBoxAutoSortAttribute</code>	The items in the combo box list will be automatically sorted in alphabetical order.
<code>kHIComboBoxDisclosurePart</code>	Disclosure part.
<code>kHIComboBoxEditTextPart</code>	Edit text part.
<code>kHIComboBoxListPixelHeightTag</code>	A <code>UInt32</code> containing the height of the combo box list. The height can be customized. This tag disables the auto size attribute.
<code>kHIComboBoxListPixelWidthTag</code>	A <code>UInt32</code> containing the width of the combo box list. The width can be customized. This tag disables the auto size attribute.
<code>kHIComboBoxListTag</code>	Extract the contents of the combo box list as a <code>CFArray</code> .
<code>kHIComboBoxNoAttributes</code>	Indicates the lack of any attributes.
<code>kHIComboBoxNumVisibleItemsTag</code>	A <code>UInt32</code> containing the number of visible items in the combo box list. The number can be customized. This tag disables the auto size attribute.

<code>kHIComboBoxStandardAttributes</code>	The minimum set of commonly used combo box attributes.
<code>kHIImageViewImageTag</code>	
<code>kHIScrollViewOptionsAllowGrow</code>	Indicates that space for a grow box should be taken into account when laying out scroll bars.
<code>kHIScrollViewOptionsHorizScroll</code>	Indicates that a horizontal scroll bar is desired.
<code>kHIScrollViewOptionsVertScroll</code>	Indicates that a vertical scroll bar is desired.
<code>kHIScrollViewValidOptions</code>	All valid scroll view options.
<code>kHIViewSendCommandToUserFocus</code>	Legacy constant. Use <code>kHIViewAttributeSendCommandToUserFocus</code> instead.
<code>kHIViewWindowContentID</code>	The standard view ID for the content view of a window.
<code>kHIViewWindowGrowBoxID</code>	The standard view ID for the grow box view of a window. Not all windows have grow boxes, so you might not find this view if you look for it.
<code>kHIViewZOrderAbove</code>	Order the view above another view.
<code>kHIViewZOrderBelow</code>	Order the view below another view.

## Keyboards.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>KLGetCurrentKeyboardLayout</code>	Obtains the keyboard layout for the layout currently selected in the Keyboards menu.
<code>KLGetIndexedKeyboardLayout</code>	Obtains the keyboard layout reference associated with the specified index.
<code>KLGetKeyboardLayoutAtIndex</code>	Obtains the keyboard layout reference associated with the specified index.
<code>KLGetKeyboardLayoutCount</code>	Returns the number of keyboard layouts.
<code>KLGetKeyboardLayoutProperty</code>	Obtains the value associated with the specified property tag for a keyboard layout.
<code>KLGetKeyboardLayoutWithIdentifier</code>	Obtains the keyboard layout reference associated with the specified identifier.
<code>KLGetKeyboardLayoutWithName</code>	Obtains the keyboard layout associated with the specified name.

<code>KLSetCurrentKeyboardLayout</code>	Sets the keyboard layout.
---	---------------------------

### Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>KeyboardLayoutIdentifier</code>	Specifies a layout identifier for a keyboard.
<code>KeyboardLayoutKind</code>	Specify the layout format used for a keyboard.
<code>KeyboardLayoutPropertyTag</code>	Specify the property tag for a keyboard layout.
<code>KeyboardLayoutRef</code>	Specifies a reference to an opaque keyboard layout data structure.
<code>kKLGroupIdentifier</code>	Specifies the keyboard layout group identifier. The data associated with this tag is an <code>Slnt32</code> value that specifies the grouping of a keyboard layout in the Keyboard menu.
<code>kKLIcon</code>	Specifies a keyboard layout icon. The data associated with this tag is an icon reference ( <code>IconRef</code> ) that specifies the icon that appears in the Keyboard menu.
<code>kKLIdentifier</code>	Specifies a keyboard layout identifier. The data associated with this tag is a “Keyboard Layout Identifier” constant.
<code>kCLKCHRData</code>	Specifies 'KCHR' layout format. The data associated with this tag is a pointer ( <code>const void *</code> ) to keyboard layout data formatted as 'KCHR' data. You can use 'KCHR' data with the Event Manager function <code>KeyTranslate</code> .
<code>kCLKCHRKind</code>	Specifies that only the 'KCHR' format is available.
<code>kCLKCHRuchrKind</code>	Specifies that both 'KCHR' and 'uchr' formats are available.
<code>kKLKind</code>	Specifies the keyboard layout format. The data associated with this tag is a “Keyboard Layout Formats” constant.
<code>kKLLocalizedName</code>	Specifies the localized keyboard layout name. The data associated with this tag is a <code>CFStringRef</code> that specifies the localized name for the keyboard layout.
<code>kKLName</code>	Specifies the keyboard layout name. The data associated with this tag is a <code>CFStringRef</code> that specifies the name for the keyboard layout.
<code>kKLuchrData</code>	Specifies 'uchr' layout format. The data associated with this tag is a pointer ( <code>const void *</code> ) to keyboard layout data formatted as 'uchr' data. You can use 'uchr' data with the Unicode Utilities function <code>UKeyTranslate</code> .
<code>kKLuchrKind</code>	Specifies that only the 'uchr' format is available.
<code>kKLUSKeyboard</code>	Specifies a US keyboard layout. All systems support this keyboard layout.

<code>PhysicalKeyboardLayoutType</code>	Specify the layout type associated with a physical keyboard.
---	--

## MacApplication.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>GetSystemUIMode</code>	Gets the presentation mode of the calling application.
<code>HIApplicationGetCurrent</code>	Returns the currently running Carbon application object.
<code>SetSystemUIMode</code>	Sets the presentation mode of the calling application.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kUIModeAllHidden</code>	All system UI elements (including the menu bar) are hidden.
<code>kUIModeContentHidden</code>	System UI elements positioned in the content area of the screen (the area other than the menu bar) are hidden.
<code>kUIModeContentSuppressed</code>	System UI elements positioned in the content area of the screen (the area other than the menu bar) are hidden.
<code>kUIModeNormal</code>	All standard system UI elements are visible.
<code>kUIOptionAutoShowMenuBar</code>	This flag specifies that the menu bar automatically shows itself when the user moves the mouse into the screen area that would ordinarily be occupied by the menu bar.
<code>kUIOptionDisableAppleMenu</code>	This flag disables all items in the Apple menu.
<code>kUIOptionDisableForceQuit</code>	This flag disables the Command-Option-Escape key sequence and the Force Quit menu item in the Apple menu to open the Force Quit window.
<code>kUIOptionDisableProcessSwitch</code>	This flag disables the Command-Tab and Command-Shift-Tab key sequences to switch the active process, and the global window rotation key sequence selected by the user in the Keyboard preference pane.
<code>kUIOptionDisableSessionTerminate</code>	This flag disables the Power key (if present) and the Restart, Shut Down, and Log Out menu items in the Apple menu.
<code>SystemUIMode</code>	Constants used to control the presentation of user interface elements provided by Mac OS X, such as the menu bar and Dock.

SystemUIOptions	Flags used to control optional behavior of system-provided user interface elements and features.
-----------------	--

## MacHelp.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kHMCStringLocalizedContent	The help tag content is a localized string contained in the Localizeable.strings file.
----------------------------	--

## MacTextEditor.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

DisposeTXNScrollInfoUPP	Disposes of the universal procedure pointer (UPP) to your scrolling callback function.
InvokeTXNScrollInfoUPP	Calls your scrolling callback function.
NewTXNScrollInfoUPP	Creates a new universal procedure pointer (UPP) to a scrolling callback function.
TXNClearUndo	
TXNRegisterScrollInfoProc	Installs or uninstalls a scrolling callback function on a text object.
TXNScroll	Scrolls the text within a view rectangle of the specified text object.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

iMaximumValue	
iScrollBarOrientation	
kTXNHorizontal	Specifies a horizontal scroll bar.
kTXNLinkNotPressed	
kTXNLinkTracking	



kTXNLinkWasPressed	
kTXNMonostyledTextBit	When this bit is set, it indicates that the text object will have a single style no matter what kind of changes are made to the object.
kTXNMonostyledTextMask	Use to set or test for the kTXNMonostyledTextBit bit.
kTXNScrollUnitsInLines	Specifies line count as the scrolling unit. Scrolling is slower when you use this unit because each line must be measured by MLTE before the text scrolls.
kTXNScrollUnitsInPixels	Specifies pixels as the scrolling unit.
kTXNScrollUnitsInViewRects	Specifies the height of the current view rectangle as the scrolling unit.
kTXNSingleLevelUndoTag	Specifies to use a single level of undo.
kTXNURLAttribute	Specifies that the data field of the TXNTypeAttributes structure contains a URL.
kTXNVertical	Specifies a vertical scroll bar.
kTXNVisibilityTag	Specifies visibility.
TXNHyperLinkState	
TXNScrollBarOrientation	Specifies the orientation of a text window's scroll bar.
TXNScrollUnit	Specify the unit by which scrolling should occur.
urlReference	

## MacWindows.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CloseDrawer	Closes a drawer.
GetAvailableWindowPositioningRegion	Obtains the available window positioning region.
GetDrawerCurrentEdge	Obtains the current window edge from which the drawer appears.
GetDrawerOffsets	Obtains the positioning offsets of a drawer.
GetDrawerParent	Obtains the parent window of a drawer.
GetDrawerPreferredEdge	Obtains the preferred opening edge for a drawer.

<code>GetDrawerState</code>	Determines the current state of the drawer.
<code>GetPreviousWindow</code>	Returns the window above the specified window in the window list.
<code>GetWindowAlpha</code>	Returns the current alpha channel value for the window.
<code>GetWindowDockTileMenu</code>	Returns the menu to be displayed by a window's dock tile.
<code>GetWindowResizeLimits</code>	Returns the minimum and maximum content sizes for a window.
<code>GetWindowToolbar</code>	Obtains the toolbar associated with a window.
<code>IsWindowToolbarVisible</code>	Determines whether a window's toolbar is visible.
<code>OpenDrawer</code>	Opens a drawer.
<code>SetDrawerOffsets</code>	Sets the positioning offsets for the drawer with respect to its parent window.
<code>SetDrawerParent</code>	Sets the parent window for a drawer.
<code>SetDrawerPreferredEdge</code>	Set the preferred window edge from which the drawer should appear.
<code>SetWindowAlpha</code>	Sets the window's alpha channel value.
<code>SetWindowDockTileMenu</code>	Associates a pop-up menu with a window.
<code>SetWindowResizeLimits</code>	Sets the maximum and minimum resize limits for windows.
<code>SetWindowToolbar</code>	Associates a toolbar with a window.
<code>ShowHideWindowToolbar</code>	Shows or hides the toolbar.
<code>ToggleDrawer</code>	Toggles the drawer state.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kDrawerWindowClass</code>	Identifies a drawer.
<code>kWindowCascadeStartAtParentWindowScreen</code>	Cascade the window on the screen containing the largest portion of its parent window, starting below and to the right of its parent window.
<code>kWindowCompositingAttribute</code>	If the bit specified by this mask is set, the window uses <code>HView</code> -based compositing.

<code>kWindowConstrainUseTransitionWindow</code>	Use <code>TransitionWindow</code> with <code>kWindowSlideTransitionEffect</code> to move windows onscreen.
<code>kWindowDefHIView</code>	The definition is an <code>HIView</code> -based object.
<code>kWindowDrawerClosed</code>	The drawer is closed.
<code>kWindowDrawerClosing</code>	The drawer is closing.
<code>kWindowDrawerOpen</code>	The drawer is open.
<code>kWindowDrawerOpening</code>	The drawer is opening.
<code>kWindowEdgeBottom</code>	The drawer should slide out from the bottom edge.
<code>kWindowEdgeDefault</code>	The drawer should be opened on whatever edge of the parent window has previously been set as the drawer's preferred edge.
<code>kWindowEdgeLeft</code>	The drawer should slide out from the left edge.
<code>kWindowEdgeRight</code>	The drawer should slide out from the right edge.
<code>kWindowEdgeTop</code>	The drawer should slide out from the top edge.
<code>kWindowIgnoreClicksAttribute</code>	If the bit specified by this mask is set, the window never receives mouse events, even in areas that are opaque.
<code>kWindowMenuIncludeRotate</code>	Requests that the standard window menu include a Rotate Windows menu item.
<code>kWindowMetalAttribute</code>	If the bit specified by this mask is set, the window has a textured or brushed-metal appearance.
<code>rootView</code>	
<code>WindowDrawerState</code>	Define constants that indicate the current drawer state.

## Menus.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>gestaltContextualMenuHasAttributeAndModifierKeys</code>	The contextual menu supports attributes and modifier keys.
<code>gestaltContextualMenuHasUnicodeSupport</code>	The contextual menu supports Unicode text.

HMenuBarHeader	Defines a list of hierarchical menus that have been inserted into a menu bar.
HMenuBarMenu	Defines a hierarchical menu.
kCMHelpItemRemoveHelp	The application does not support any help.
keyContextualMenuAttributes	Specifies the menu item attributes. The data format for this parameter must be typeLongInteger.
keyContextualMenuCommandID	The menu item command ID. The data format for this parameter must be typeLongInteger.
keyContextualMenuModifiers	Specifies the modifier keys for the menu item.
keyContextualMenuName	The menu item text.
keyContextualMenuSubmenu	The submenu of the menu item. You typically use this with the Apple Event Manager function AEPutDesc to add an entire AEDesc record (which contains the submenu) as the parameter data.
kMenuAttrHidden	Do not draw the menu title, even when the menu is inserted in the menu bar.
kMenuEjectGlyph	The Eject key (available in Mac OS X v10.2 and later).
kMenuItemAttrIncludeInCmdKeyMatching	If this attribute is set, functions such as MenuKey, MenuEvent and IsMenuKeyEvent examine this menu item during command key matching.
lastHMenu	
lastMenu	
lastRight	
mbResID	
MenuBarHeader	Defines a list of nonhierarchical menus that that have been placed in the menu bar.
MenuBarMenu	Defines a nonhierarchical menu.
menuLeft	
menuTitleBits	

## TextServices.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

DeselectTextService	Notifies TSM that an input method has been closed.
GetDefaultInputMethodOfClass	Obtains the default input method text service component for a given text service class.
SelectTextService	Selects a text service.
SetDefaultInputMethodOfClass	Sets the default input method text service component for a given text service class.
TSMGetDocumentProperty	Obtains a TSM document property.
TSMRemoveDocumentProperty	Removes a property from a TSM document.
TSMSetDocumentProperty	Sets a property for a TSM document.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

fontRef	
glyphInfo	
kCharacterPaletteInputMethodClass	Specifies a text service class for Character Palette input methods.
kInkInputMethodClass	Specifies a text service class for Ink input methods.
kKeyboardInputMethodClass	Specifies a text service class for keyboard input methods.
kTSClassForceSetting	
kTSClassForceToHonorUserSetting	
kTSClassHonorUserSetting	
kTSMDocumentPropertySupportGlyphInfo	You should no longer use this property.
numGlyphInfo	
TextServiceClass	Specify text service classes supported by TSM.
TSClassEnablingForceLevel	

TSMGlyphInfo	Describes one glyph embedded in a run of text.
TSMGlyphInfoArray	Contains an array of glyph information structures.

## Print

---

### PMApplication.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

PMSessionDisablePrinterPresets	Disables the use of printer presets in the Print dialog.
PMSessionEnablePrinterPresets	Enables the use of printer presets in the Print dialog.

### PMPrinterBrowsers.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kPMPrBrowserPCGetTitle	
------------------------	--

### PMPrintingDialogExtensions.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

PMUpdatePrintButton	
---------------------	--

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kEventClassPrinting	
kEventParamPDEHeight	
kEventPrintingPDEResize	

kPMSyncPaneFromTicket	
kPMSyncTicketFromPane	

## ImageCapture

---

### ICAApplication.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

ICAScannerCloseSession	
ICAScannerGetParameters	
ICAScannerInitialize	
ICAScannerOpenSession	
ICAScannerSetParameters	
ICAScannerStart	
ICAScannerStatus	

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

deviceObject	
eventDataCookie	
eventDataSize	
extd	
ICAEventDataCookie	
ICAExtendedRegisterEventNotificationPB	
ICAScannerCloseSessionPB	
ICAScannerGetParametersPB	
ICAScannerInitializePB	

## 10.2 Symbol Changes

ICAScannerOpenSessionPB	
ICAScannerSessionID	
ICAScannerSetParametersPB	
ICAScannerStartPB	
ICAScannerStatusPB	
kEventClassPTPStandard	
kEventClassPTPVendor	
kExtendedNotificationPB	
kMessageGetEventData	
rawEventType	
sessionID	



# 10.1 Symbol Changes

---

This article lists the symbols added to `Carbon.framework` in Mac OS X v10.1.

## Classes

All of the classes with new symbols are listed alphabetically, with their new class, instance, and delegate methods described.

## C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

### CarbonSound

---

Sound.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>siHALAudioDeviceID</code>
---------------------------------

### CommonPanels

---

NSL.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kNSLClientHandlesRecents</code>
---------------------------------------

## OpenScripting

---

### ASRegistry.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

enumConsidsAndIgnores	
kAECASEConsiderMask	
kAECASEIgnoreMask	
kAEDIACRITICConsiderMask	
kAEDIACRITICIgnoreMask	
kAEEXPANSIONConsiderMask	
kAEEXPANSIONIgnoreMask	
kAEHYPHENSConsiderMask	
kAEHYPHENSIgnoreMask	
kAEPUNCTUATIONConsiderMask	
kAEPUNCTUATIONIgnoreMask	
kAEWHITESPACEConsiderMask	
kAEWHITESPACEIgnoreMask	
kASCONSIDERREPLIESConsiderMask	
kASCONSIDERREPLIESIgnoreMask	

## HI Toolbox

---

### Appearance.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CopyThemeIdentifier	Retrieves a string identifying the current theme variant.
---------------------	---

**Data Types & Constants**

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kThemeBrushSheetBackgroundOpaque</code>	
<code>kThemeBrushSheetBackgroundTransparent</code>	
<code>kThemeBrushToolBarBackground</code>	
<code>kThemeMenuItemNoBackground</code>	This value may be added to other <code>ThemeMenuItemType</code> constants to specify that the menu background should not be drawn along with the menu item. Available in Mac OS X.
<code>kThemeMetricMenuExcludedMarkColumnWidth</code>	The width allocated for the mark character in a menu item when the menu has the attribute <code>kMenuAttrExcludesMarkColumn</code> .
<code>kThemeMetricMenuItemTrailingEdgeMargin</code>	The whitespace at the trailing edge of a menu icon, if the item also has text.
<code>kThemeMetricMenuItemIndentWidth</code>	The width per indent level of a menu item. This indent is set by the <code>SetMenuItemIndent</code> function.
<code>kThemeMetricMenuMarkColumnWidth</code>	The width allocated to draw the mark character in a menu.
<code>kThemeMetricMenuMarkIndent</code>	The indent into the interior of the mark column at which the mark character is drawn.
<code>kThemeMetricMenuItemTextLeadingEdgeMargin</code>	The whitespace at the leading edge of menu item text.
<code>kThemeMetricMenuItemTextTrailingEdgeMargin</code>	The whitespace at the trailing edge of menu item text.
<code>kThemeVariantBaseTintTag</code>	

**CarbonEvents.h****Functions**

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>CreateTypeStringWithOSType</code>	Converts an <code>OSType</code> string to a Core Foundation string.
<code>GetCFRunLoopFromEventLoop</code>	Obtains a Core Foundation <code>CFRunLoop</code> from an Carbon event loop reference.
<code>IsMouseCoalescingEnabled</code>	Indicates whether mouse coalescing is enabled.

SetMouseCoalescingEnabled	Turns mouse coalescing on or off.
---------------------------	-----------------------------------

### Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

eventTargetBusyErr	The event target you are trying to modify is busy (for example, dispatching an event).
HIPoint	Defines the position of a point using floating-point coordinates.
kEventAppearanceScrollBarVariantChanged	The scroll bar variant has changed.
kEventAppGetDockTileMenu	Request to display a pop-up menu by the application's dock tile.
kEventClassAppearance	Events related to the Appearance Manager.
kEventClassService	Events related to the Services Manager.
kEventMenuDrawItem	Sent by the standard window definition if a menu item has the kMenuItemAttrCustomDraw attribute set.
kEventMenuDrawItemContent	Sent by the standard window definition if a menu item has the kMenuItemAttrCustomDraw attribute set.
kEventMenuMeasureItemHeight	Sent by the standard window definition if a menu item has the kMenuItemAttrCustomDraw attribute set.
kEventMenuMeasureItemWidth	Sent by the standard window definition if a menu item has the kMenuItemAttrCustomDraw attribute set.
kEventMenuPopulate	Sent when an application should dynamically create a menu.
kEventParamDeviceColor	typeBoolean
kEventParamDeviceDepth	typeShortInteger
kEventParamKeyboardType	
kEventParamMenuCommandKeyBounds	
kEventParamMenuContext	
kEventParamMenuDrawState	
kEventParamMenuItemIconBounds	

kEventParamMenuItemBounds	
kEventParamMenuItemHeight	
kEventParamMenuItemType	
kEventParamMenuItemWidth	
kEventParamMenuMarkBounds	
kEventParamMenuTextBaseline	
kEventParamMenuTextBounds	
kEventParamMenuVirtualBottom	
kEventParamMenuVirtualTop	
kEventParamMouseChord	typeUInt32
kEventParamNewScrollBarVariant	typeShortInteger
kEventParamScrapRef	When provided as a parameter to the kEventServicePaste event, the current selection should be replaced by data from this scrap.
kEventParamServiceCopyTypes	When provided as a parameter to the kEventServiceGetTypes event, add CFString references to this array to report the types that can be pasted from the current selection.
kEventParamServiceMessageName	When provided as a parameter to the kEventServicePerform event, contains the name of the advertised service that was invoked.
kEventParamServicePasteTypes	When provided as a parameter to the kEventServiceGetTypes event, add CFString references to this array to report the types that can be copied from the current selection.
kEventParamServiceUserData	When provided as a parameter to the kEventServicePerform event, contains extra data provided by the requestor.
kEventParamTabletEventType	
kEventParamTabletPointRec	typeTabletPointRec
kEventParamTextInputReplyFMFont	typeUInt32
kEventServiceCopy	The user wants to invoke a service that requires your application to provide data.
kEventServiceGetTypes	Sent when the Services Manager needs to know what types of data it can cut-and-paste into the scrap.

## 10.1 Symbol Changes

kEventServicePaste	The user has invoked a service that requires your application to receive data.
kEventServicePerform	Sent when your application must perform a service.
kEventTabletPoint	Indicates that the pen has moved on a tablet. (Mac OS X only)
kEventWindowClickStructureRgn	
kEventWindowClickToolBarButtonRgn	Sent when the mouse is down in the toolbar button. The default behavior is to call TrackBox and then generate a kEventWindowToolBarSwitchMode event.
kEventWindowToolBarSwitchMode	The toolbar button (that is, the oblong button used to show and hide the toolbar) was successfully clicked.
kHICommandAppHelp	The application's help book should be displayed.
kHICommandBringAllToFront	All windows of this application should be brought in front of windows from other applications. Sent only on Mac OS X. The default application handler responds to this event automatically; your application does not need to handle it.
kHICommandClose	The active window should be closed.
kHICommandHideOthers	Other applications should be hidden. The Menu Manager responds to this command automatically; your application does not need to handle it.
kHICommandMaximizeAll	All collapsible windows should be maximized. This event is not sent or handled on Mac OS X.
kHICommandMaximizeWindow	The active window should be maximized. Sent only on Mac OS 9. The default application handler will respond to this event automatically; your application does not need to handle it.
kHICommandMinimizeAll	All collapsible windows should be minimized. The default application handler responds to this event automatically; your application does not need to handle it.
kHICommandNew	A new document or item should be created.
kHICommandOpen	The user wants to open an existing document.
kHICommandPageSetup	The user wants to configure the current page margins, formatting, and print options.
kHICommandPrint	The active window should be printed.

## 10.1 Symbol Changes

<code>kHICommandRevert</code>	The contents of the active document should be reverted to the last saved version.
<code>kHICommandSave</code>	The active document should be saved.
<code>kHICommandSaveAs</code>	The user wants to save the active document under a new name.
<code>kHICommandSelectWindow</code>	A window in the standard Window menu has been selected and should be activated.
<code>kHICommandShowAll</code>	All applications should become visible. The Menu Manager responds to this command automatically; your application does not need to handle it.
<code>kHICommandWindowListSeparator</code>	A placeholder to mark the separator item dividing the Zoom/Minimize/Maximize/Arrange menu items in the standard Window menu from the menu items listing the visible windows.
<code>kHICommandWindowListTerminator</code>	Used as a placeholder to mark the end of the window list section of the standard Window menu.
<code>kMenuContextKeyMatching</code>	This event is being sent while trying to match a command-key equivalent to a menu item.
<code>kMenuContextMenuBar</code>	The menu associated with this event is in the menu bar or is a submenu of a menu in the menubar.
<code>kMenuContextMenuBarTracking</code>	This event is being sent while a menu is being tracked in the menu bar.
<code>kMenuContextMenuEnabling</code>	Sent at idle time to update the enabled state of the menus.
<code>kMenuContextPopUp</code>	The menu associated with this event is a popup menu displayed by the Menu Manager function <code>PopUpMenuSelect</code> .
<code>kMenuContextPopUpTracking</code>	This event is being sent while a popup menu is being tracked.
<code>kMenuContextPullDown</code>	
<code>kMenuContextSubmenu</code>	The menu associated with this event is a submenu of a pulldown or popup menu.
<code>kMouseTrackingMouseDown</code>	The user pressed any mouse button.
<code>kMouseTrackingMouseDragged</code>	The mouse moved while the mouse button was down.
<code>kMouseTrackingMouseUp</code>	The user released the mouse button.
<code>TabletPointRec</code>	Defines a tablet point structure.

typeCFMutableArrayRef	A Core Foundation mutable array.
typeCFStringRef	A Core Foundation string.
typeHIPoint	HIPoint
typeScrapRef	
typeTabletPointRec	kEventParamTabletPointRec
typeThemeMenuItemType	
typeThemeMenuState	

## ControlDefinitions.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

DisposeDataBrowserDrawItemUPP	Disposes of a universal procedure pointer to a draw-item callback function.
DisposeDataBrowserEditItemUPP	Disposes of a universal procedure pointer to an edit-item callback function.
DisposeDataBrowserHitTestUPP	Disposes of a universal procedure pointer to a hit-test callback function.
DisposeDataBrowserItemAcceptDragUPP	Disposes of a universal procedure pointer to an item-accept-drag callback function.
DisposeDataBrowserItemDragRgnUPP	Disposes of a universal procedure pointer to an item-drag-region callback function.
DisposeDataBrowserItemNotificationWithItemUPP	Disposes of a universal procedure pointer to an item-notification-with-data callback function.
DisposeDataBrowserItemReceiveDragUPP	Disposes of a universal procedure pointer to an item-receive-drag callback function.
DisposeDataBrowserTrackingUPP	Disposes of a universal procedure pointer to a tracking callback function.
GetDataBrowserItemDataMenuRef	Obtains the pop-up menu displayed.
InvokeDataBrowserDrawItemUPP	Calls a draw-item callback function.
InvokeDataBrowserEditItemUPP	Calls an edit-item callback function.
InvokeDataBrowserHitTestUPP	Calls a hit-test callback function.



InvokeDataBrowserItemAcceptDragUPP	Calls an item-accept-drag callback function.
InvokeDataBrowserItemDragRgnUPP	Calls an item-drag-region callback function.
InvokeDataBrowserItemNotificationWithItemUPP	Calls an item-notification-with-data callback function.
InvokeDataBrowserItemReceiveDragUPP	Calls an item-receive-drag callback function.
InvokeDataBrowserTrackingUPP	Calls a tracking callback function.
NewDataBrowserDrawItemUPP	Creates a universal procedure pointer to a draw-item callback function.
NewDataBrowserEditItemUPP	Creates a universal procedure pointer to an edit-item callback function.
NewDataBrowserHitTestUPP	Creates a universal procedure pointer to a hit-test callback function.
NewDataBrowserItemAcceptDragUPP	Creates a universal procedure pointer to an item-accept-drag callback function.
NewDataBrowserItemDragRgnUPP	Creates a universal procedure pointer to an item-drag-region callback function.
NewDataBrowserItemNotificationWithItemUPP	Creates a universal procedure pointer to an item-notification-with-data callback function.
NewDataBrowserItemReceiveDragUPP	Creates a universal procedure pointer to an item-receive-drag callback function.
NewDataBrowserTrackingUPP	Creates a universal procedure pointer to a tracking callback function.
SetDataBrowserItemDataMenuRef	Sets the pop-up menu to display.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kControlClockAnimatingTag	
kControlEditTextPasswordCFStringRefTag	Extract the content of the edit text field if it is a password field.
kControlKindEditUnicodeText	
kControlScrollTextBoxAnimatingTag	
kControlTabImageContentTag	

## Dialogs.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kStdAlertDoNotAnimateOnCancel</code>	Do not animate hiding the sheet window when the user presses the Cancel button.
<code>kStdAlertDoNotAnimateOnDefault</code>	Do not animate hiding the sheet window when the user presses the default button.
<code>kStdAlertDoNotAnimateOnOther</code>	Do not animate hiding the sheet window when the user presses the other button.

## Drag.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>GetDragAllowableActions</code>	Returns the actions that the drag receiver may take on the data within a drag.
<code>GetDragDropAction</code>	Returns the action performed by the receiver of the drag.
<code>SetDragAllowableActions</code>	Sets the actions that the drag receiver may take on the data within a drag.
<code>SetDragDropAction</code>	Sets the action performed by the receiver of the drag.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>DragActions</code>	Specify the actions that should be or have been performed on the data in a drag.
<code>flavorDataPromised</code>	Set by the sender if the flavor data is promised at a later time.
<code>kDragActionAlias</code>	The data contained in the drag can be or has been shared.
<code>kDragActionAll</code>	Indicates that all of the above drag actions are allowed.
<code>kDragActionCopy</code>	The data contained in the drag can be or has been copied.
<code>kDragActionDelete</code>	The data contained in the drag can be or has been deleted.

## 10.1 Symbol Changes

<code>kDragActionGeneric</code>	When set by the drag sender, suggests that the drag receiver can determine the drag action. When returned by the drag receiver, indicates that the receiver did not define a drag action.
<code>kDragActionMove</code>	The data contained in the drag can be or has been moved.
<code>kDragActionNothing</code>	Nothing should be or has been done with the data in the drag. When set as an allowable action for remote drags, the drag is not sent to applications other than the drag sender.
<code>kDragActionPrivate</code>	Suggests that the drag action should be negotiated privately between the drag source and destination.

## Events.h

---

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kAppleLogoCharCode</code>	Available in Mac OS X v10.1 and later.
<code>kAppleLogoUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kBulletCharCode</code>	Available in Mac OS X v10.1 and later.
<code>kBulletUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kCheckCharCode</code>	Available in Mac OS X v10.1 and later.
<code>kCheckUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kCommandCharCode</code>	Available in Mac OS X v10.1 and later.
<code>kCommandUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kControlUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kDiamondCharCode</code>	Available in Mac OS X v10.1 and later.
<code>kDiamondUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kOptionUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kPencilUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kShiftUnicode</code>	Available in Mac OS X v10.1 and later.
<code>kSpaceCharCode</code>	Available in Mac OS X v10.1 and later.

## Icons.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

GetIconRefFromFileInfo	Provides an IconRef object for a file with minimal file I/O.
PlotIconRefInContext	Plots an IconRef using Quartz.
ReadIconFromFSRef	Reads an icon ('icns') file into memory.
RegisterIconRefFromFSRef	Registers an IconRef from a .icns file and associates it with a creator and type pair.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kAlertCautionBadgeIcon	
kBurningIcon	
kDeleteAliasIcon	
kEjectMediaIcon	
kGenericURLIcon	
kGenericWindowIcon	
kIconServicesCatalogInfoMask	Use this mask with the File Manager function FSGetCatalogInfo before calling GetIconRefFromFileInfo.
kInternationalResourcesIcon	
kPlotIconRefNoImage	
kPlotIconRefNoMask	
kPlotIconRefNormalFlags	
kPublicFolderIcon	
kQuestionMarkIcon	
kRightContainerArrowIcon	
kToolbarCustomizeIcon	

kToolbarDeleteIcon	
kToolbarFavoritesIcon	
kToolbarHomeIcon	
PlotIconRefFlags	

## MacApplication.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

GetApplicationDockTileMenu	Returns the menu containing items added to the contextual menu for your application Dock tile.
SetApplicationDockTileMenu	Adds items to the contextual menu for your application Dock tile.

## MacHelp.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

HMHideTag	Hides the most recently displayed help tag.
-----------	---

## MacTextEditor.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

TXNClearActionChangeCount	Resets the specified action counters to zero.
TXNGetActionChangeCount	Retrieves the number of times the specified action or actions have occurred since the count was initialized or cleared.
TXNGetRectBounds	Obtains the values for the current view, destination, and text rectangles.
TXNRecalcTextLayout	Recalculates the text layout based on new view and destination rectangles.

TXNSetRectBounds	Set the view rectangle and/or the destination rectangle.
------------------	--

### Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kTXNAllCountMask	Use to set or test for both kTXNTextInputCountBit and kTXNRunCountBit.
kTXNDisableDragAndDrop	Disables drag and drop when passed as a parameter to the function TXNSetTXNObjectControls. Indicates drag and drop is enabled when returned from the function TXNGetTXNObjectControls.
kTXNDisableDragAndDropBit	When this bit is set, it indicates that drag and drop will not be allowed in the text object.
kTXNDisableDragAndDropMask	Use to set or test for the kTXNDisableDragAndDropBit bit.
kTXNDisableDragAndDropTag	Specifies settings for drag and drop support.
kTXNDontWrapTextBit	When this bit is set, indicates MLTE should not wrap text.
kTXNDontWrapTextMask	Use to set or test for kTXNDontWrapTextBit bit.
kTXNEnableDragAndDrop	Enables drag and drop when passed as a parameter to the function TXNSetTXNObjectControls. Indicates drag and drop is disabled when returned from the function TXNGetTXNObjectControls.
kTXNImageWithQDBit	When this bit is set, indicates MLTE should use QuickDraw for imaging text.
kTXNImageWithQDMask	Use to set or test for kTXNImageWithQDBit bit.
kTXNRunCountBit	When this bit is set, general style changes to the text are included in the action count. Style changes include changes to the text face, font, font size and so forth. Undo and redo events of style changes are also included in the action count.
kTXNRunCountMask	Used to set or test for the kTXNRunCountBit.
kTXNSingleLineOnlyBit	When this bit is set, it indicates that the text object will not scroll vertically.
kTXNSingleLineOnlyMask	Use to set or test for the kTXNSingleLineOnlyBit bit.
kTXNTextInputCountBit	When this bit is set, general text input events that affect the content of the document are included in the action count.
kTXNTextInputCountMask	Use to set or test for the kTXNTextInputCountBit.

## 10.1 Symbol Changes

<code>kTXNUseCGContextRefBit</code>	When this bit is set, indicates MLTE should the Quartz context (CGContext) you provide instead of the temporary CGContextRef created internally by MLTE.
<code>kTXNUseCGContextRefMask</code>	Use to set or test for <code>kTXNUseCGContextRefBit</code> bit.
<code>kTXNUseQDforImagingBit</code>	When this bit is set, it indicates that QuickDraw will be used for imaging instead of Quartz, which is the default. Available in Mac OS X only.
<code>kTXNUseQDforImagingMask</code>	Use to set or test for the <code>kTXNUseQDforImagingBit</code> bit.
<code>kTXNWillDefaultToCarbonEventBit</code>	When this bit is set, indicates MLTE uses Carbon events by default.
<code>kTXNWillDefaultToCarbonEventMask</code>	Test for Carbon events as the default event handling mechanism.
<code>TXNCountOptions</code>	Set or test action count bits for use with <code>TXNGetActionChangeCount</code> and <code>TXNClearActionChangeCount</code> .
<code>TXNLongRect</code>	Contains coordinates for a view or text rectangle.

## MacWindows.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>GetAvailableWindowAttributes</code>	Returns the window attributes that are valid for a window class
<code>GetWindowGroupRetainCount</code>	Determines the current reference count for a window group.
<code>GetWindowStructurePort</code>	Obtains a graphics port that is used when drawing a window's structure.
<code>IsValidWindowClass</code>	Determines whether a given window class is valid.
<code>IsWindowLatentVisible</code>	Indicates whether a window is visible onscreen or is latently visible but not currently onscreen.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>inStructure</code>	The user has pressed the mouse button while the cursor is in the window's structure region.
--------------------------	---

<code>inToolBarButton</code>	The user has pressed the mouse button while the cursor is in the toolbar button. (Mac OS X only.)
<code>kAltPlainWindowClass</code>	Identifies an alternate plain window, which is similar to a plain window but has a solid black shadow on its right and bottom sides.
<code>kSheetAlertWindowClass</code>	Identifies an alert sheet.
<code>kWindowGroupAttrHideOnCollapse</code>	When any window in this group is collapsed, all other windows in this group are hidden.
<code>kWindowGroupAttrLayerTogether</code>	The z-order of the contents of this group with respect to each other cannot be changed.
<code>kWindowGroupAttrMoveTogether</code>	The positions of the contents of this group with respect to each other cannot be changed. When one item moves, all other items are moved simultaneously. Note that if one window's position is changed by calling a Window Manager function in Mac OS X v10.4 and later, the positions of the other windows in the group are updated asynchronously—that is, their bounds are not necessarily updated during the function call itself, even though visually the windows move together.
<code>kWindowGroupAttrSelectAsLayer</code>	Makes the group behave somewhat as a layer of windows that move together.
<code>kWindowIsOpaque</code>	Indicates that the window is entirely opaque. If this feature bit is set, the window will use less memory because no alpha channel information will be stored for the window's pixels.
<code>kWindowLatentVisibleAppHidden</code>	The window's process is hidden.
<code>kWindowLatentVisibleCollapsedGroup</code>	The window is in a group for which <code>kWindowGroupAttrHideOnCollapse</code> is set, and another window in the group was collapsed.
<code>kWindowLatentVisibleCollapsedOwner</code>	The window is in an owned group, and the owner was collapsed.
<code>kWindowLatentVisibleFloater</code>	The window is a floating window, and floating windows are hidden.
<code>kWindowLatentVisibleFullScreen</code>	The window has <code>kWindowHideOnFullScreenAttribute</code> set and the mode is full-screen.
<code>kWindowLatentVisibleSuspend</code>	The window has <code>kWindowHideOnSuspendAttribute</code> set and the application is suspended.
<code>kWindowNoConstrainAttribute</code>	If the bit specified by this mask is set, the window is not repositioned by the default <code>kEventWindowConstrain</code> handler.



<code>kWindowSheetAlertDefProcResID</code>	Defines a sheet window that is displayed as an alert (rather than a dialog) on Mac OS 9.
<code>kWindowSheetAlertProc</code>	An alert sheet.
<code>kWindowToolbarButtonAttribute</code>	If the bit specified by this mask is set, the window has a toolbar button.
<code>WindowLatentVisibility</code>	Defines window latent visibility constants.
<code>wInStructure</code>	The mouse-down event occurred in the window's structure region.
<code>wInToolbarButton</code>	The mouse-down event occurred in the toolbar button.

## Menus.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>CMPluginExamineContext</code>	An application-defined callback function that examines the context in a contextual menu CFPlugin.
<code>CMPluginHandleSelection</code>	An application-defined callback function that handles menu item selection in a contextual menu CFPlugin.
<code>CMPluginPostMenuCleanup</code>	An application-defined callback function that handles any post-selection cleanup in a contextual menu CFPlugin.
<code>EraseMenuBackground</code>	Erases the menu background to prepare for additional drawing.
<code>ScrollMenuImage</code>	Scrolls a portion of the menu image.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>CM_IUNKNOWN_C_GUTS</code>	
<code>ContextualMenuInterfaceStruct</code>	
<code>ExamineContext</code>	
<code>HandleSelection</code>	
<code>kMenuItemResourceType</code>	Identifies a CFString that names an icon resource in the main bundle of the application.

kMenuItemAttrCustomDraw	This is a custom menu item. Setting this attribute causes custom menu item drawing Carbon events to be sent to your application. Available in CarbonLib 1.4 and Mac OS X v10.1 and later.
kMenuSystemIconSelectorType	Identifies an OSType value that corresponds to an icon (type IconRef) registered with Icon Services under kSystemIconsCreator.
MDEFDrawData	Contains information needed to draw a menu.
MDEFDrawDataPtr	
MDEFFindItemData	Contains information used to determine which item the user has currently selected.
MDEFFindItemDataPtr	
PostMenuCleanup	

## Processes.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

IsProcessVisible	Determines the visibility of the user interface for a process.
ShowHideProcess	Shows or hides a given process.

## Scrap.h

---

### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

ClearScrap	Clears a given scrap.
GetScrapByName	Obtains a reference to a named scrap.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kScrapClearNamedScrap	Acquire the named scrap, bumping and clearing the scrap.
-----------------------	--

kScrapGetNamedScrap	Get current named scrap without bumping or clearing the scrap.
---------------------	--

## Print

---

### PMPrintingDialogExtensions.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

PMCreatePaperSizeCFString	Returns a text description of the physical dimensions of the paper specified in a paper information ticket. The unit of measure is based on the current language and script system.
---------------------------	---

## ImageCapture

---

### ICAApplication.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

ICACopyObjectPropertyDictionary	
ICADownloadFile	

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

dirFSRef	
fileFSRef	
ICACopyObjectPropertyDictionaryPB	
ICADownloadFilePB	
kAddMetaDataToFinderComment	
kAdjustCreationDate	

## 10.1 Symbol Changes

kCreateCustomIcon	
kDeleteAfterDownload	
kEmbedColorSyncProfile	
kICAPropertyColorSpace	
kICAPropertyImageAperture	
kICAPropertyImageDateDigitized	
kICAPropertyImageDateOriginal	
kICAPropertyImageExposureTime	
kICAPropertyImageFlash	
kICAPropertyImageFNumber	
kICAPropertyImageShutterSpeed	
kRotateImage	
kSetFileTypeAndCreator	
rotationAngle	
theDict	

### ICACamera.h

---

#### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kICAMessageCameraUploadData	
kICAPropertyCameraSupportedMessages	

## NavigationServices

---

### Navigation.h

---

#### Functions

---

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>NavCreateAskReviewDocumentsDialog</code>	Creates a Review Changes dialog, which notifies the user of multiple unsaved documents and gives the user the option to review them.
<code>NavDialogGetSaveFileExtensionHidden</code>	Gets the current state of extension hiding in a Save dialog.
<code>NavDialogSetSaveFileExtensionHidden</code>	Sets the current state of extension hiding in a Save dialog.

### Data Types & Constants

---

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

<code>kNavPreserveSaveFileExtension</code>	(Available in Navigation Services 3.1 and later.) Tells Navigation Services that the extension in the default filename should be preserved between dialog invocations and is initially hidden.
<code>kNavUserActionDiscardDocuments</code>	The user clicked the Discard Changes button in the Review Documents dialog (used only on Mac OS X).
<code>kNavUserActionReviewDocuments</code>	The user clicked the Review Unsaved button in the Review Documents dialog (used only on Mac OS X).
<code>saveFileExtensionHidden</code>	



# Document Revision History

---

This table describes the changes to *Carbon Reference Update*.

Date	Notes
2007-07-18	Updated with the symbols added to the Carbon framework in Mac OS X v10.5.
2005-06-04	New document that summarizes the symbols added to the Carbon framework in Mac OS X v10.4.

