

HIToolbar Classes for REALbasic 2007r2

Introduction

Since the addition of the `ToolbarItem` and `StandardToolbarItem` classes to REALbasic, developers have had mixed feelings. Sure, they are toolbars that aren't too difficult to manage, but other 3rd-party solutions existed that were not only more complete, but cross-platform. What REAL Software provided us was a set of classes that were pretty much half-baked.

The classes are built upon the HIToolbar in Mac OS X's Human Interface Toolbox. These are the same toolbars as Finder, Mail, and System Preferences. Except they are missing most of the functionality.

To help correct this mess, the HIToolbar and HIToolbarItem classes were created. They also just "wrapper" classes to the Mac OS X functionality of the same name, but they function just like every other toolbar in Mac OS X.

Requirements

As much as we tried, we found it impossible to create custom toolbar images without PNG conversion – which could not be done without a plugin. So these classes require the PNG Utilities Plugin, which is included in the archive. Also, because of linking to Quartz, all applications that use these classes must be built in Mach-O format. Applications built with these toolbars require Mac OS X 10.2.

Concepts

Unlike the Toolbars built-into REALbasic, you do not drag every single item you want to support into your window. You only need to drag the HIToolbar class into your window. Because of the customization, you actually will only define the identifiers of the items you want to support. When the OS needs to create the item for any reason, the toolbar is asked to create it, providing the identifier as the basis. This is a strange design for a REALbasic application, but it's the way Mac OS X handles it.

Identifiers

Mac OS X uses a lot of reverse-style-domain-identifiers. The HIToolbar uses an absurd amount of them. Every item must have its own unique identifier. Every toolbar must do the same. Even your application must have one.

Let's say a The ZAZ is developing a product known as "Stumbler!" for Mac OS X. The best identifier for "Stumbler!" would be `com.thezaz.stumbler`. For the main toolbar in the application, we'd want to give it an identifier of `com.thezaz.stumbler.maintoolbar` or something like that. And for each item in the toolbar, like a Stumble button, the identifier should be `com.thezaz.stumbler.maintoolbar.stumble`.

Every identifier should get more and more specific, as you can tell from example.

HIToolbar

Events

DefaultItems(Identifiers() As String)

Mac OS X fires this event to ask the toolbar to generate the default toolbar items seen at the bottom of a customization sheet.

DisplayModeChanged()

Fired because the user has changed the display mode between "Icon & Text", "Icon Only", and "Text Only".

DisplaySizeChanged()

Fired because the user has changed the display size between "Normal Icons" and "Small Icons".

Error(ErrorCode As Integer)

The toolbar has encountered an error of some sort. ErrorCode is a Mac OS result code.

ItemAdded()

The user has added an item to the toolbar.

ItemClicked(Identifier As String)

The user has clicked an item in the toolbar. You are provided with the item's identifier to determine what action to take.

ItemCreated(Item As HIToolbarItem) As Boolean

Mac OS X is asking your toolbar to populate the HIToolbarItem provided. This event is called a lot, so don't do anything too time-consuming. The provided item already has its identifier set, but everything else needs to be set. Do not store or save item anywhere. The OS needs to release the object itself. If you have made changes to Item, return true. If you do not want to support the item, or if the item is an OS standard item, return false.

ItemRemoved()

The user has removed an item from the toolbar.

LayoutChanged()

The user has changed the layout of the toolbar.

Methods

AppendItem(ItemIdentifier As String)

Apple strongly discourages adding items to the toolbar for the user, but if you feel it necessary, you can add an item to the end of toolbar using this method. Only pass the identifier. The ItemCreated event will be fired.

Create(InWindow As Window, ToolbarIdentifier As String, ApplicationIdentifier As String) As Boolean

Before the toolbar can function, this method needs to be called in the Window's open event. This method installs the Carbon Event handlers, creates the toolbar, and sets Window attributes. If anything fails, it will return false. Otherwise it returns true.

Destroy()

This method MUST be called in the Window close event. It saves the toolbar configuration for later retrieval (which your users will like) and removes the toolbar from memory.

IndexOf(ItemIdentifier As String) As Integer

This does not return the location of an item in the toolbar. It only returns the location of the identifier in the internal identifier array.

InsertItem(IntoSpace As Integer, ItemIdentifier As String)

Apple strongly discourages adding items to the toolbar for the user, but if you feel it necessary, you can add an item into the space provided using this method. Only pass the identifier. The ItemCreated event will be fired.

Item(Index As Integer) As String

Returns the identifier at the index provided. This is not the index of an item in the toolbar, but the index of an identifier in the internal identifier array.

Item(ItemIdentifier As String) As HIToolBarItem

Returns an HIToolBarItem from the toolbar with the identifier provided. If the item is not in the toolbar, Nil will be returned.

Items As HIToolBarItem()

Returns all the items in the toolbar as an array.

RegisterToolbarItem(ItemIdentifier As String)

Usually called after the Create function, you need to register each of the item identifiers with the toolbar using this method.

RemoveItem(Index As Integer)

Apple strongly discourages removing items from the toolbar for the user, but if you feel it necessary, you can remove an item using this method. Index is a zero-based index of an item in the toolbar.

ShowCustomizationSheet()

Displays the customization sheet for the toolbar.

Properties**Count As Integer [Read-Only]**

The number of identifiers in the internal identifier array.

DisplayMode As Integer

The current display mode of the toolbar. Setting this will change the toolbar. Use the HIToolbar.kDisplayMode* constants.

DisplaySize As Integer

The current icon size for the toolbar. Setting this will change the toolbar. Use the HIToolbar.kDisplaySize* constants.

Handle As Integer [Read-Only]

The Mac OS X handle.

IsCustomizable As Boolean

Used to get or set the customization ability of the toolbar.

Parent As Window [Read-Only]

The owner window of the toolbar.

Visible As Boolean

Used to get or set the visibility of the toolbar.

HIToolBarItem**Methods****SetImage(Image As Picture)**

Used to set the icon of the toolbar item.

SetImageFromIcon(Creator As String, Type As String)

Used exactly like the ToolbarItem function of the same name. For example, creator to "macs" and type to "FNDR" to set the icon to the Finder icon.

Properties**AllowDuplicates As Boolean**

Although implemented, Mac OS X doesn't appear to actually support this attribute. Setting this will cause an error, as of Mac OS X 10.4.

AnchoredLeft As Boolean

Causes the toolbar item to be anchored to its leftmost position.

CantBeRemoved As Boolean

Locks the item into the toolbar.

Caption As String

The caption of the toolbar item.

CommandID As String

Used heavily in Nib-Based Carbon applications, the CommandID will be sent to the application when pressed. Since REALbasic does not support custom CommandID handling out-of-the-box, this is fundamentally useless. Certain commands, such as "tcfg" are automatically handled by the OS.

Enabled As String

The toolbar item's enabled state.

Handle As Integer [Read-Only]

The Mac OS X handle to the toolbar item.

HelpTag As String

Get or set the standard help tag.

Identifier As String [Read-Only]

The identifier of the toolbar item.

Selected As Boolean

Sets the selected state of the toolbar item. Only supported on Mac OS X 10.4.

Toolbar As HIToolbar [Read-Only]

Gets the toolbar the item is contained in.

ToolbarHandle As Integer [Read-Only]

Gets the handle of the toolbar the item is contained in.

Changes**1.1.2**

- Fixed a critical bug which prevented toolbars from triggering their ItemClicked events on Intel Macs.

1.1.1

- Fixed a bug caused by a change in REALbasic 2007r2

- Added a bundle identifier for example project to support opening in Rosetta on Intel Macs.

1.1

- HIToolbar is now Universal, and includes the universal PNG Utilities 1.8.1.

- Support easy Unified Titlebar/Toolbar style windows. Simply call window.HasUnifiedLook = true.

- Default toolbar items now appear properly on first launch.

- Updated code allows support for both 10.3 and 10.4.

1.0

- Initial Release

License

These classes are open source and free for use in any application without special mention. You are not allowed to distribute modified copies of the archive, without prior permission. Please link to the original page whenever possible.

Contact

Made by Thom McGrath, <thom@thezaz.com>