

Copyright © 1996-1997 Jeremy Wyld. All rights reserved.

More Routing

This auto part adds items to all routing menus. It overrides a method in root that may cease to exist someday, thus possibly killing this hack. Don't blame me! Playing with undocumented "features" isn't for the light-hearted. Extra items can be added, or those I provide may be yanked. Read on to find out how to do it yourself.

Installing

Use NPI or NBU to install this extension onto your Newton device. This extension will only work on Newton devices with Newton OS 2.0 (or greater). The package will take around 1K of user store space.

Using More Routing

Tap on any routing/action button and a slightly modified popup will appear. This popup will look similar to figure 1.



Figure 1.

Notice the added quick jumps to "IOBox" and "Styles." Others may be added with close to no effort.

Rolling Your Own

Included Files

- | | |
|-------------------|--|
| Graphics.f | Loads graphics used in the routing frames. |
| MiscFunctions.f | Contains the definition for general functions used in the rest of the files. |
| RouteFrames.f | Defines the route frames to be added to the routing menu. |
| GetRouteScripts.f | Definition for the "new" :GetRouteScripts(). |

InstallScript.f InstallScript() definition for this auto part.
RemoveScript.f RemoveScript() definition for this auto part.

Adding More

In order to add more items to the routing menu, I've supplied a few compile time functions. These include `AddRouteFrame()`, `AddRouteItem()`, `AddOpenItem()`, and `AddSeparator()`. Place the calls to these functions in the file `RouteFrames.f` (this keeps them all together in one place).

`AddRouteFrame(routeFrame)`

This function takes one argument, the actual frame to be added to the routing menu. For documentation on routing frames, see the “Routing Interface” chapter in NPG.

`AddRouteItem(title, icon, routeScript)`

This function then makes a routing frame from these arguments and calls `AddRouteFrame()` with this new frame. The routing frame it creates looks as such:

```
{
  title      title,          // Name to show up in the menu.
  icon:      icon,          // As that returned by
                                   // GetPICTAsBits()
  routeScript: routeScript // Function to be called.
}
```

`AddOpenItem(title, icon, sym)`

This function takes three arguments, the title, the icon and the symbol of the item to open in root (e.g. 'iobox). It builds a frame similar to the one above but with a method that opens the root item.

`AddSeparator()`

This function takes no arguments and adds a pick separator (dotted line) to the menu.

Version History

v1.5 First public release. Added all the documentation. Got rid of the NTK compiler warnings.

v1.0 Personal version.

Any Ideas Or Requests?

Send email to me, Jeremy Wyld, for any requests or ideas.
jeremy-wyld@postmaster.net