

I've been using Frontier off and on, and I do manage to make some use of it, but it's interface for dealing with files, disks and other stuff left a lot to be desired. Standard file sort of bites after a while, particularly when you want to do something to a lot of files.

Frontier supports a cool technology called Menu Sharing, which allows savvy applications to import a menu from Frontier. If the user edits the menu in Frontier, the changes are automatically communicated to the App.

Well, it seemed obvious that the Finder was the obvious target for this type of technology - being able to select a bunch of files and say "Make aliases of these files on the server".

FinderMenu 1.0 was the result - I've been hacking on it for a about a month. I wanted to wait for this conference to unveil it, but I made the mistake to showing it to the Frontier guys, who just about cried when they saw it. They wanted people to have it, so they arranged for testing and even did a press release. (Sort of out of the spirit of a hack, right?) (The guys from UserLand aren't that bad, though - they just wanted people to get this utility as soon as possible)

Well, I added a few features while I've been at this conference (just so that it doesn't seem like I'm completely unfair about this) but there is a more important thing about FinderMenus than just Frontier: more than one Application can add menus to the Finder.

So, with the source code provided (and unfortunately, I didn't have time to perfect the sample app that provides menusharing for Frontier) any application (or background app, which is what the Sample program is) can install menus into the Finder.

Look at "FinderMenuApp.c" in the FinderMenuApp f for an example of how to use this technology.

I'll run down basically what all is involved:

1) Initialize your connection to FinderMenus with:

*short firstMenuID = FMInit();*

*the return value, firstMenuID is the first menu ID that you should use for your menus (hierarchical and otherwise.) Don't make too many menus: no attempt is made to prohibit you from doing so, but it is possible to write over other peoples menus.*

2) set up apple event handlers for handing a cmd-., and for getting a menuhit Apple event (I sometimes use Userlands IAC toolkit - available on Compuserve, or from me if you catch up to me at this conference.)

*IACInstallhandler(cGetFinderMenuProc, cFinderCancelHitEvent, (ProcPtr) &HandleCancel)*

*This event is called when the user presses cmd-. in the Finder, between the time the App is sent it's MenuHit event, and when the App informs the INIT that the App is done with the menu selection*

*IACInstallhandler(cGetFinderMenuProc, cFinderMenuHitEvent, (ProcPtr) &HandleMenuHit)*

*The parameter to the FinderMenuHitEvent is a DescList containing FSSpecs for each item selected in the Finder.*

3) Install your menus:

*FMInstallMenu(MenuHandle hMenu, short beforeID*

*Note that FinderMenu clones the handle. If you want to change it, then you:  
FMRemoveAllMenus()  
to remove all your menus from the Finders menubar.*

4) when you get hit, you can periodically ask for the beachball to be rotated while the finder is frontmost:

*FMRollBeachball()  
will roll the beachball at the next SystemTask in the Finder*

5) when you're done with processing, you need to tell FinderMenu so he can reenable your menus:

*FMFinishedProcessing()*

6) when you're all done, remove yourself cleanly:

*FMRemove()*

**I realize that these instructions are something less than clear - I had intended on writing a small sample App that would demonstrate these things, but time did not allow it.**

**Due to a series of Bizarre Circumstances™, this code is Copyright by Berkeley Systems. Any derivitave work needs to make mentioin of this in it's documentation.**

**While it may appear that this is a nasty hack, please remember that it's gotten a lot of testing, and I've been getting a lot of feedback on how to improve it. And I almost certainly will be improving it. The UserLand forum on Compuserve is curretnly reaping the most benefits from FinderMenus, but I've gotten more technical tidbits in the past two days than in a month of CIS. If you've got comments, please let me know:**

**Steve Zellers**

**Berkeley Systems**

**2095 Rose Street**

**Berkeley, CA 94709**

**CIS: 75460,1375 <- email preferred!**

**510-540-5535**

**p.s.: a note to former FinderMenu users. The version in this release, 1.1a1 has been stable for me - the important difference is that it is now a faceless background app, and the INIT is bundled within it. Nuke your old "FinderMenu App" and "FinderMenu INIT" files, and drop this one into the Extension Folder. No other changes are needed.**