

HackMac 1.0

HackMac is a skeletal platform on which to prototype Cocoa add-ons and hacks.

The skeleton comes with a small amount of code to implement an About dialog, an optional, skeletal Preferences dialog, and to allow optional insertion of a menu in the application menu of apps that load it.

HackMac is a Cocoa InputManager bundle. This means that if it is installed in one of the allowable locations (`~/Library/InputManagers`, `/Network/Library/InputManagers`, or `/Local/Library/InputManagers`), all Cocoa applications will load it when they launch.

Warning: InputManagers can provide a very powerful ability to modify the behavior of Cocoa apps, but it can also be dangerous. Your bug can render many of the apps on your system, including some really important ones, useless if you are not careful.

For a more concrete example of what can be done with this mechanism you might want to check out TextExtras which is available at:

<http://www.lorax.com/FreeStuff/FreeStuff.html>. TextExtras is an InputManager bundle that adds lots of cool text-related features to all Cocoa apps.

The purpose of the HackMac project is to get you quickly started in this form of hacking. Starting with this skeleton you can merely add the code to actually implement whatever add-on or hack you are prototyping and not worry about the rest of the setup.

Warning: Namespace pollution is an important consideration for loadable bundles. Bundles, especially InputManager bundles, should take care to be extremely considerate residents in their host applications. It is very important that if you ever want to distribute an InputManager like this that it not collide with other such bundles. If you want to distribute a bundle, make sure that all the classes it uses have a unique prefix (NOT "HQ") and that any preferences you write also have a prefix. Finally, if you add category methods to existing classes that you did not write, prefix the method names as well. All this helps avoid potential collisions between your bundle and the apps that load it or other bundles.

To aid the development cycle, the project has been set up so that the InputManager "Info" file is copied into the build results folder of the project when you build. What this means is that during development you can create a symbolic link called "HackMac" in the folder "~/Library/InputManagers" which points at the build results folder for the project so that apps that are launched will load your development build of the bundle. Obviously, doing this is even more dangerous than the normal dangers of this sort of thing. You probably don't want to leave such a link in place permanently (especially, you should probably remove it before you log out).