

# Using ResEdit

## *Modifying Menus*

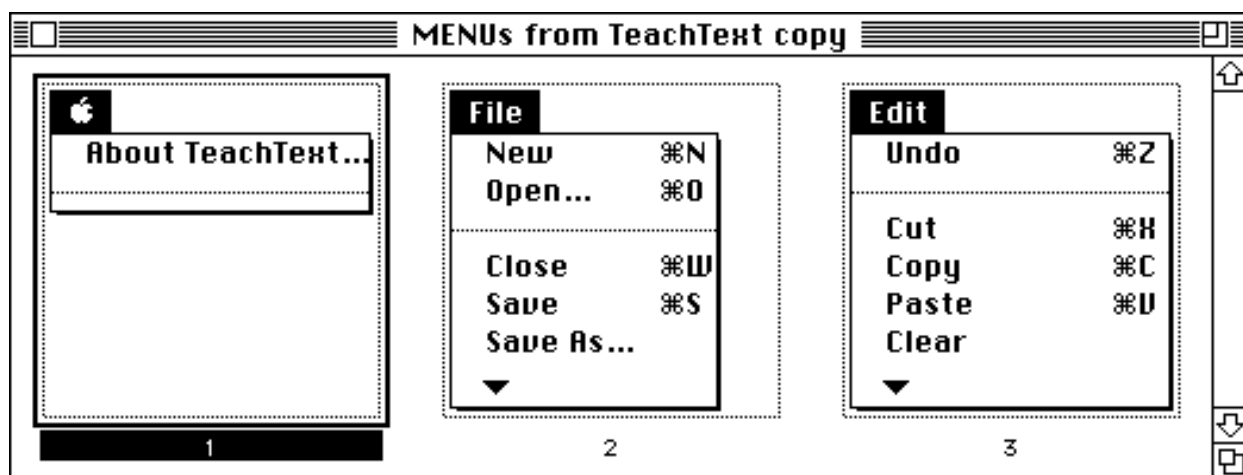
by Chris Reed

After you've played around with icons for a while, it's time to learn how to edit something that can actually improve your efficiency and make using the Mac easier—menus. These resources are what really make a Macintosh special, because you can change just about anything in them and they'll still work. Be careful, however—always work on backups. Most programs will tolerate the maximum amount of twiddling, but there are some that simply will not work correctly—they might even bomb—if you change anything. That's why you need to keep backups.

Often a program will have command keys which you dislike or think are unfitting. If the program uses normal 'MENU' resources, the command keys can be changed very easily. Other changes you can make to menus are: renaming menu items and titles, changing the style(s) of a menu item, and coloring menus and items.

### Opening a menu

To open a menu, follow the normal procedure for opening a resource picker. The 'MENU' picker looks like the window in figure 1. It draws as much of each menu as it can in the space for it. This feature is new to ResEdit 2.1, and makes looking for a particular 'MENU' much easier. When you have found the menu you want, double-click on it or select it and choose Open Resource Editor from the Resource menu. After opening a 'MENU' resource, the 'MENU' editor window will show up and a MENU menu will be added to the menu bar. Also, an example menu (it's not really in the menu bar, and it doesn't do anything) will be added to the right of the MENU menu. This menu represents the resource you are editing, so you can see what it will look like when in use.



Figure

1. The 'MENU' resource picker with the basic menus.

### The menu editor window

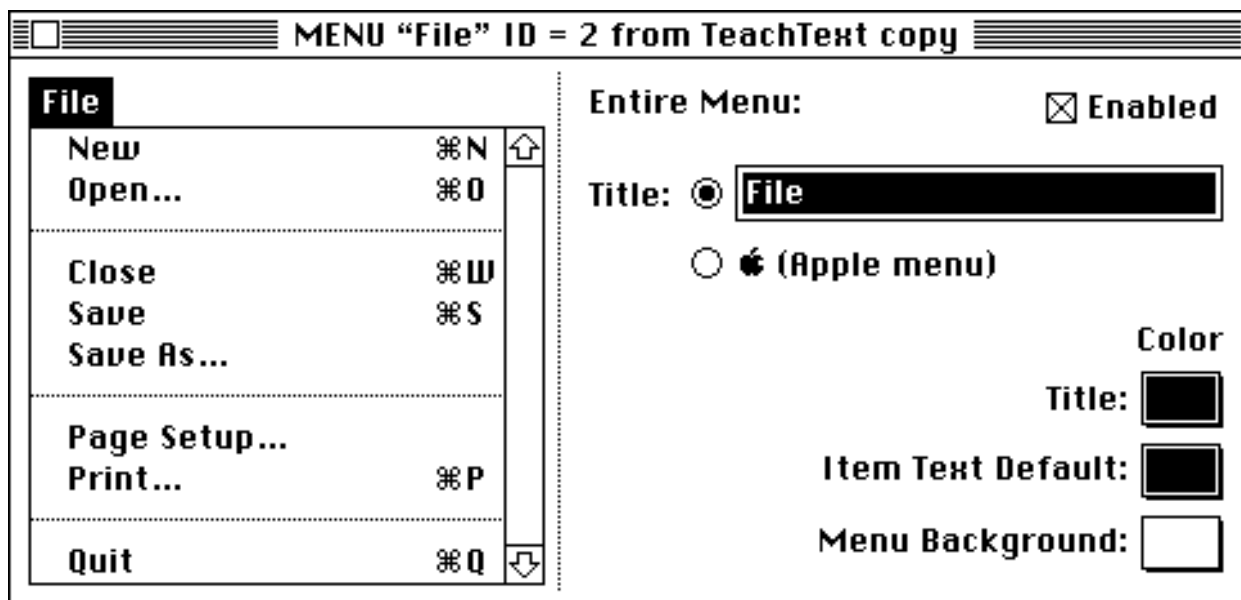
A picture of the menu editor window is shown in figure 2. On the right side of the gray divider bar is a

list of items in the menu and a box for the title. It shows a representation of each item, in the appropriate colors. However, it won't show an icon attached to an item. In figure 2, the title is selected. The options on the right side of the divider will change depending on what you have

selected. For the title, they will look like they do in figure 2. When an item is selected, they change to look like figure 3. If you select an item in the fake menu it will be selected in the item list.

With the title selected, the Enabled checkbox enables or disables the whole menu. If there are any disabled items (they can be individually disabled) then this will not affect them. You can change the title of the menu with the edit text field labeled Title. Clicking on the ☐ (Apple menu) radio button will make the title an ☐, which is easier than typing it. As you change the menu's title the changes will be reflected both in the item list and the example menu.

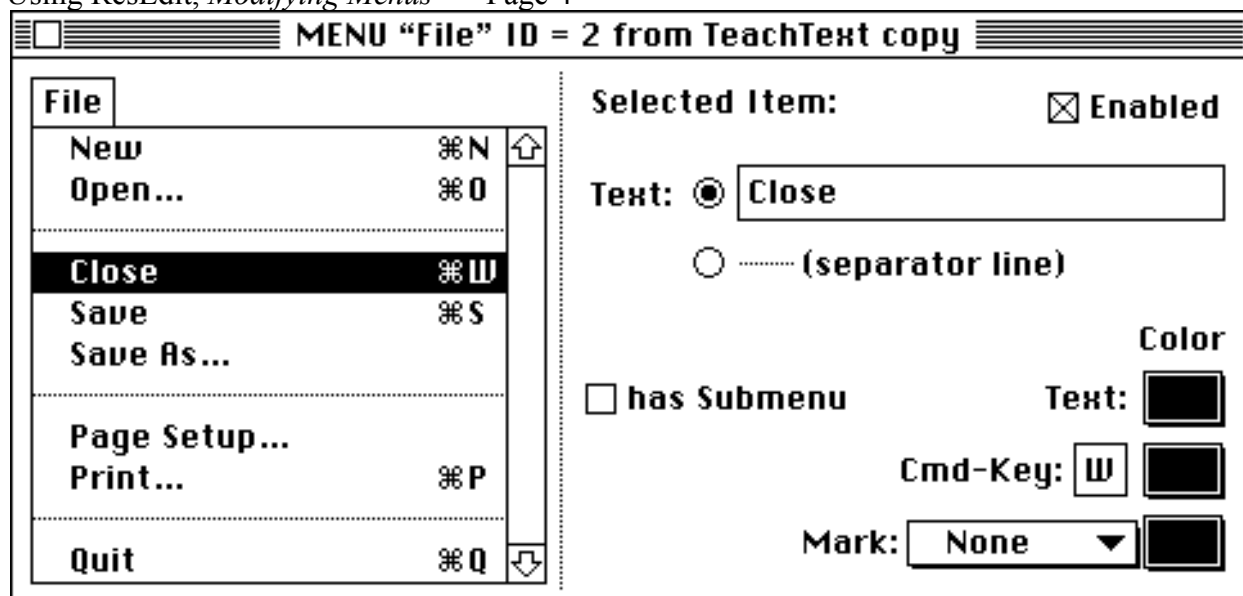
You can change the colors for the title, the default text color, and the menu background by clicking on the color swatches. A popup menu will appear that lets you select a color from the standard palette for whatever depth your screen is set to.



Figure

2. The 'MENU' editor window with the title selected.

Once an item is selected, the right side of the divider bar will change to look like the window in figure 3. The Enabled checkbox lets you turn the selected item on or off. You can change the item's text the same way you change the title's. To change the item's command key equivalent, click on the Cmd-Key edit text. The entire text will be selected if there's already a command key. Typing a character will change the command key and reselect the text. If you would like to mark an item (e.g. to mark all commands that delete data with a †) click on the Mark popup menu. If the character you wish to mark the item with is not in the menu, select Other. A dialog will appear for you to type the character you wish to use as a mark. The Remember in Preferences option will save your new mark character in ResEdit's preferences file. The colors of the text, command-key, and the mark can all be different.



Figure

3. The 'MENU' editor window with a menu item selected.

If an item is hierarchical—that is, it has a submenu that pops out from it when it is selected—the Has Submenu option will be checked. For a hierarchical item, the window changes to look like figure 4. You can change the ID of the submenu with the ID edit item. The ID must be within the range of 0 to 255. You can change the color of the little arrow in the normal manner.

When a selected item has a submenu, you can either double-click on it in the item list or select it and choose Open Submenu to open the menu editor for the item's submenu. If the submenu doesn't exist it will be created and opened.

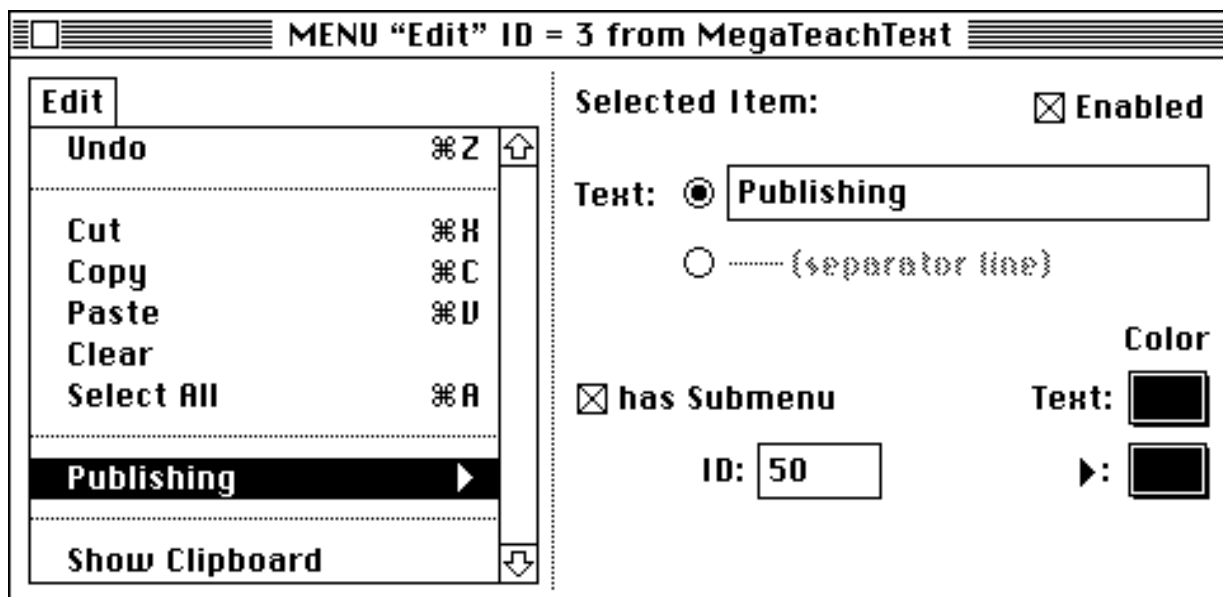


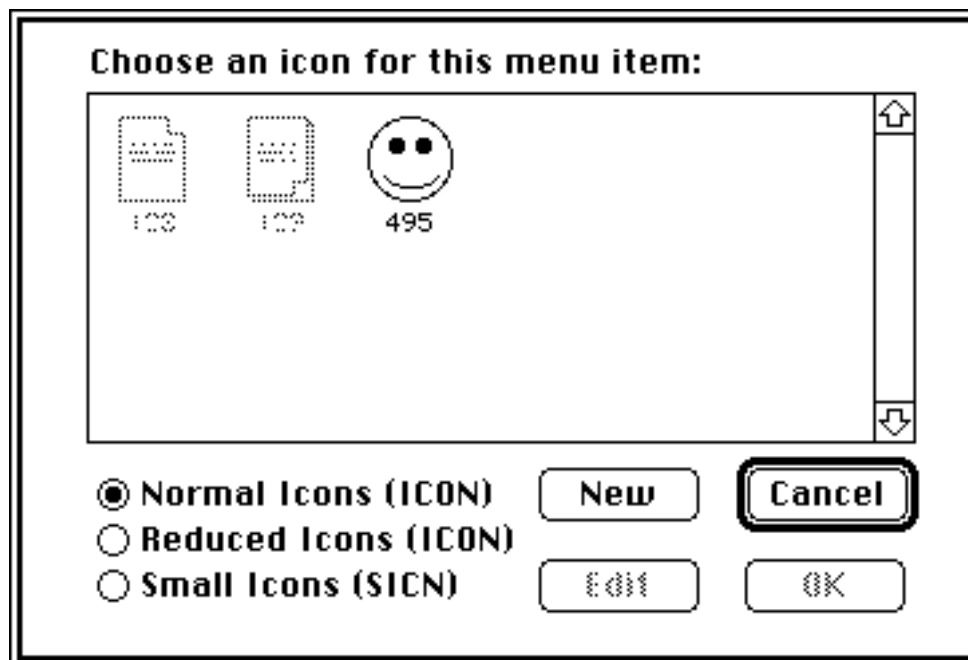
Figure 4. A menu item with a submenu.

## Adding icons to your menus

If you want to spruce up your menus with icons, you can easily do it with the Choose Icon item in the MENU menu. Selecting Choose Icon will bring up the dialog in figure 5. Initially, Normal Icons will be selected and the list will contain all of the 'ICON' resources in the same file as the menu. Any gray icons in the list cannot be selected, because they are not in the range of 257 to 511. Any icons within this range can be selected. The icon will not show up in the item list, but will appear in the example menu.

If you want to use a smaller icon, you have two options. You can shrink a large icon to 16-by-16, which usually results in ugly icons, by clicking the Reduced Icons radio button. This displays the same icons as the Normal Icons radio button does, but they're half size. If you wish to use a 'SICN' (Small ICoN) resource, click the Small Icons radio button. The list will be updated to reflect your selection.

If you want to remove an icon from the selected item, use the Remove Icon command in the MENU menu.

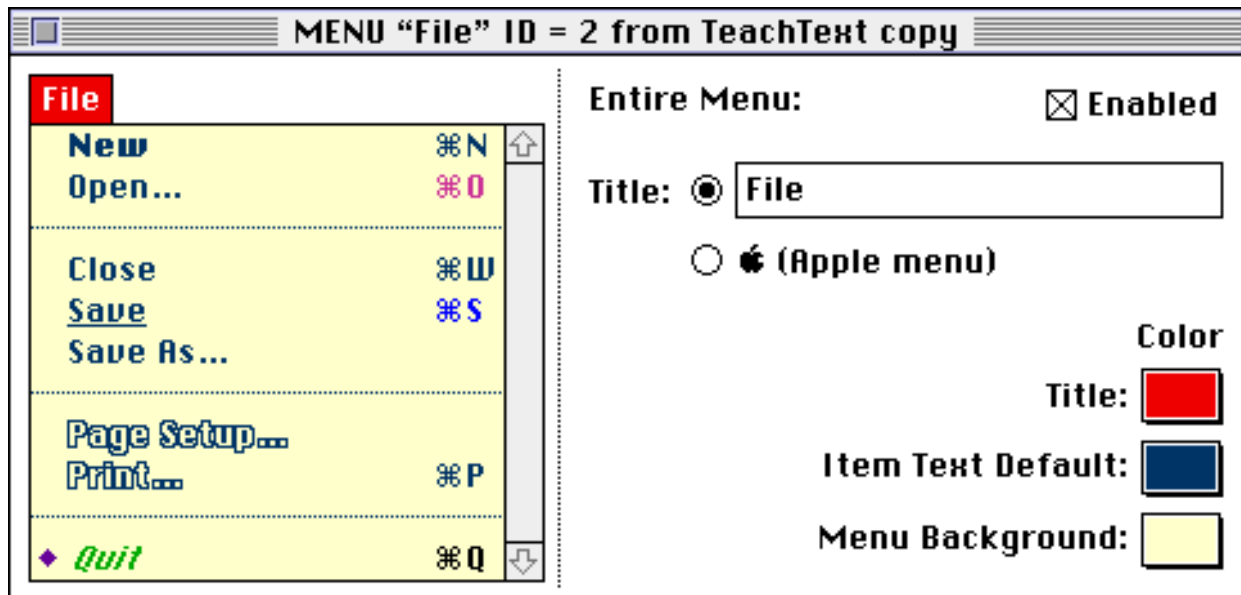


**Figure 5.** Choosing an icon.

## Using color and styles in menus

If you are using a color Mac, you can colorize your menus. The best use of these features would be to categorize commands in a program. For example, you could color all commands that deleted data red, all commands that changed data irreversibly yellow, and all commands that could be undone green. Similarly, styles menu items can also have different text styles. The title can't change styles. The Style menu lets you change the style of the selected item. Figure 6 shows a menu that was customized a bit too much.

The color information for a menu is saved in another resource type, a 'mctb' (menu color table) resource of the same ID. The first time you add color to a menu, you will be told that adding color to a menu will add an 'mctb' resource in which to store the color information. The Remove Colors command deletes all color from the menu by deleting the menu's corresponding 'mctb' resource.



Figure

6. An extreme example of customization.

### The menu ID and MDEF ID

If you are developing a program, you need to take a look at the Edit Menu & MDEF ID command, otherwise it will be of no use to you. It displays the dialog in figure 7 which lets you edit the menu ID (*not* the resource ID) and the ID of the menu's definition procedure.

The menu ID is what the program refers to it's menu by. Usually the resource ID and the menu ID will be the same, but this is not always the case. If you change the menu's resource ID when the menu editor is open, you will be asked if you want to change the menu ID to the new resource ID. Faulty menu IDs often cause lots of problems, because in ResEdit they are not as easy to spot.

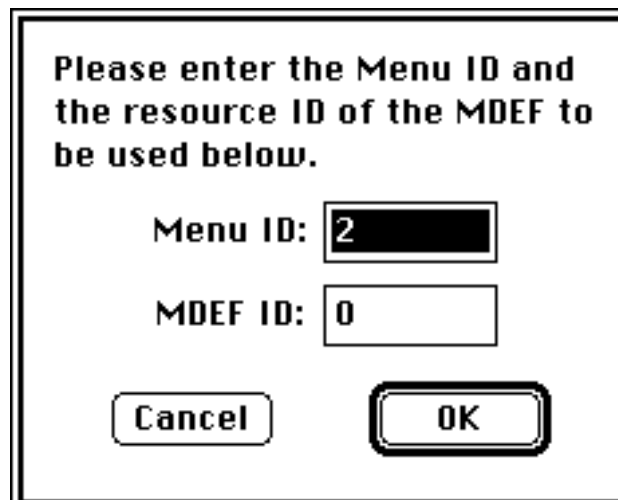


Figure 7. The Menu and MDEF ID dialog.

## **What is safe to customize.**

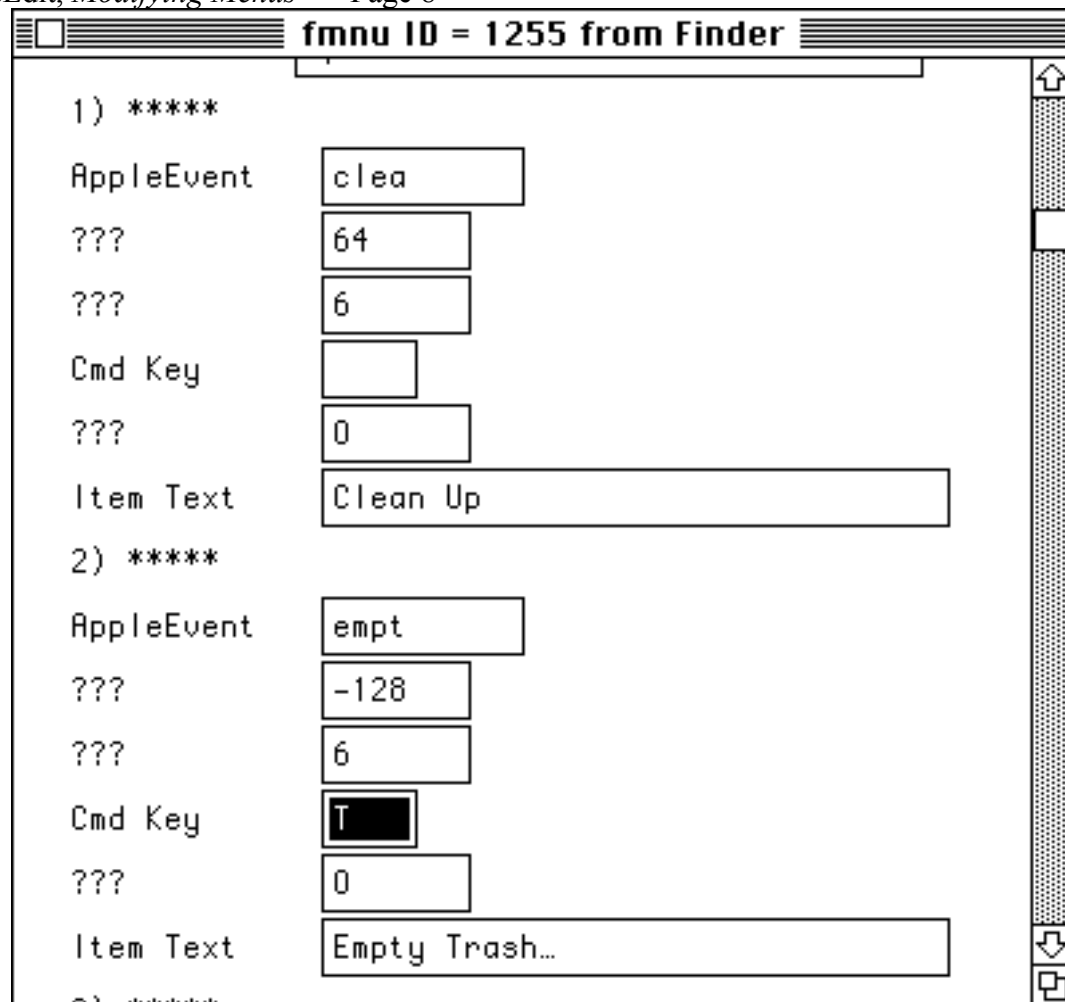
The safest thing to do is to add and change colors. There is nothing that can go wrong with this unless you are modifying a really weird program. The next safest thing, and probably the most useful, is adding and changing command keys. Editing menu titles and names will most likely work fine, but the program might change items on the fly—“Show Clipboard” and “Hide Clipboard” for example—so there is a possibility your custom items won’t show up. (To remedy this, try looking through all the 'STR ' and 'STR#' resources for the replacement menu items.) Also, there are menus that there is no reason to edit, like Font, Size, and Style menus.

While editing, remember that programs refer to items by numbers and not by text. For example, the standard Undo command in the Edit menu would be item 1 of menu 3. Programs don’t care what the item text of a menu item is, only what the item number is. Because of this, you cannot move commands around in a menu. For example, you can’t swap Cut and Copy in the Edit menu because the program will still treat item 3 as Cut even though you changed the text to Copy. For that matter, you can’t add commands to a program by adding new menu items.

## **Changing the Finder**

One of the most popular programs to change is the Finder itself. For example, you can add command keys to Empty Trash, Restart, and Shut Down. Because the Finder is always running, you can’t directly edit it under MultiFinder. There is an easy solution, however. (This works in System 7, too.) First, duplicate the Finder in the System Folder. Now drag the original out of the System Folder. Finally, rename the duplicate to “Finder.” You can now edit the duplicate Finder in ResEdit and still run the original. When you reboot, it will be with the modified Finder.

**Finder 7.** If you take a look at Finder 7, you’ll notice it doesn’t have any 'MENU' resources. That’s because it stores its menus in 'fmnu' resources. There is no way to edit 'fmnu' resources unless you use the template in the ResEdit 2.1 TMPLs for 7.0 file I have included. Figure 8 shows what the template window looks like.



**Figure 8.** The 'fmnu' template window.

To install the template (and several other System 7 templates), open the ResEdit 2.1 TMPLs for 7.0 file and copy everything but the 'fmnu' resource into the ResEdit Preferences file in either the System Folder or the Preferences folder (depending on which system you're running). If you are asked if you want to replace resources with the same ID, use unique IDs.

The only field that you should touch in the 'fmnu' resources is the Cmd Key field. *Do not* change anything else. Notice the three ??? fields. That shows how much of the 'fmnu' resource is actually documented.

### 'Til next time...

...have fun with your menus! I'll be talking about dialogs, windows, and extending ResEdit.