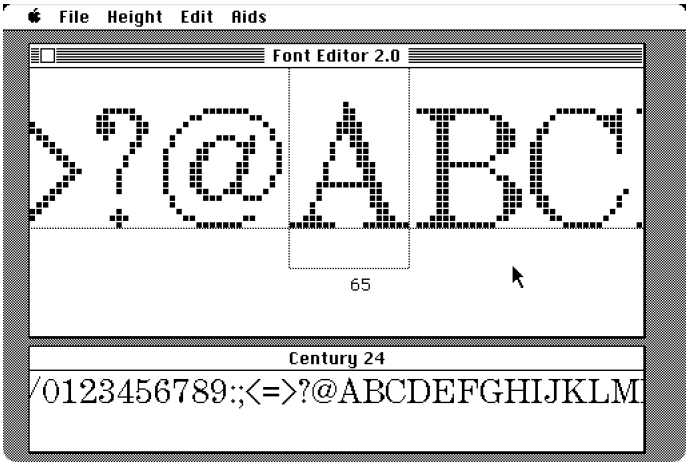
Version 2, May 31, 2024

Mark's Guide to Using Apple's Ancient "Font Editor 2.0" App

In 1984, as owner of an original 128K Macintosh, I was very interested in how to make fonts for it. It came with a set fonts named after cities, like Geneva, New York, Monaco, and Athens. But there seemed to be no way for an ordinary user to make their own fonts. As a graphic designer and aspiring type designer, this was something I really wanted to do.

Not even a custom icon.

A solution arrived in the July 1984 issue of *St. Mac*, an early magazine devoted to the Macintosh. There was a short article on the last page saying that you could send $100 (plus tax) to Apple and get a set of disks containing "supplemental software" (i.e., developer tools) including an app called "Font Editor." This was exactly what I was looking for. So I sent the check.



One of my early bitmap fonts, based on Century Expanded.

Font Editor 2.0, as far as I can remember, had no manual or documentation. By trial and error (and many system crashes), I managed to figure out how it worked. I was able to design a handful of bitmap fonts in preliminary stages. A few months later, Altsys released Fontastic, a commercial bitmap editor for the Mac that was better than Font Editor in almost every way—and that's where I did most of my bitmap font work.

My vision was to sell sets of these fonts like clip art. But after Apple introduced the LaserWriter in 1985, and Altsys released Fontographer (an app for making PostScript fonts to use with it), I lost interest in making bitmap fonts.

Fontastic™ Plus

The better tool.

Bitmap fonts I made from 1984-86. I started out in Font Editor, but soon switched to Fontastic and Fontastic Plus.

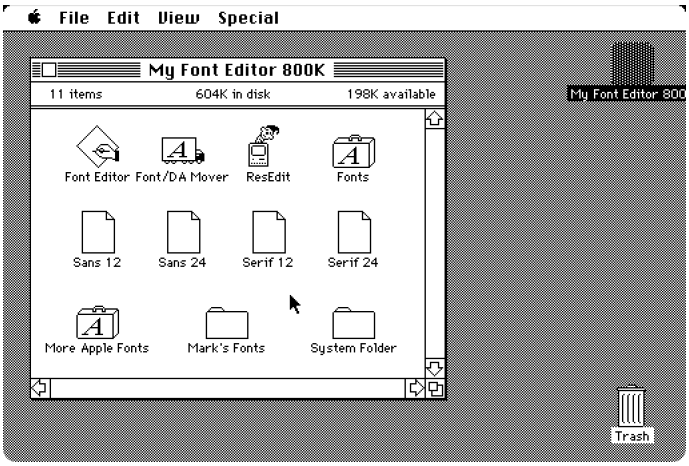
In Font Editor is very primitive and tricky to work with if you don't know what you're doing. It appears to *only* work on a 128K or 512K Macintosh (not the 512Ke or later). The only emulator I've gotten it to work with is a custom 128K Macintosh variant of Mini vMac (See https://www.gryphel.com/c/minivmac/). The original disk included Mac system software 1.0 (1984). It will also run on system software 2.0 (1985), but nothing later. This means that it can only be run from floppy disks (or floppy disk images in the case of vMac), since hard drives were not supported by these early systems.

This also means that it will only work with MFS disks (or disk images), not HFS disks. MFS disks have a flat file structure, meaning that all files on the disk are at the same level—no folder hierarchy. You *can* create folders in the Finder, but they are purely cosmetic and won't appear in open or save dialog boxes. And you can't have two files with the same name, even if they are in different "folders" because folders don't really exist.

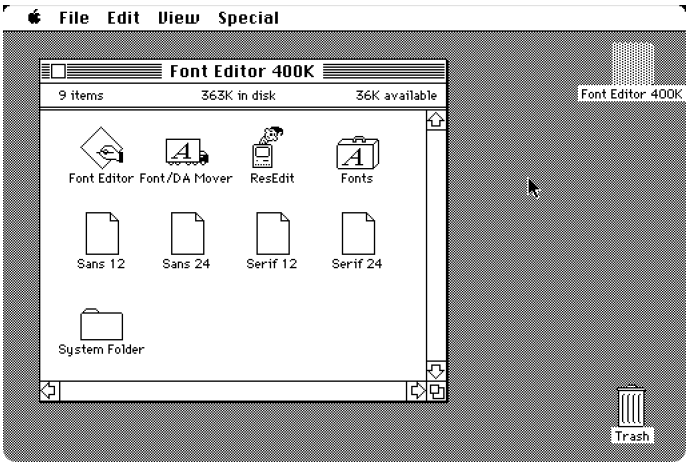
System software 2.0 is preferable. System software 1.0 is missing a lot of handy features, like “shut down” and “new folder.” In system 1.0, there had to be an “Empty Folder” on every disk. If you wanted a new folder, you were supposed to make a copy of “Empty Folder.” It’s hard to believe this was ever a thing, but it was for us early users in the first year of the Mac.

If you can meet all these requirements, you’re ready to use (or try to use) Font Editor.

First Steps



800K Font Editor disk image setup.



400K Font Editor disk image setup.

For this user guide, I’m going to be working in a 128K variant of Mini vMac on a modern MacBook Pro. I’ve set up an 800K MFS disk image containing the following files and apps:

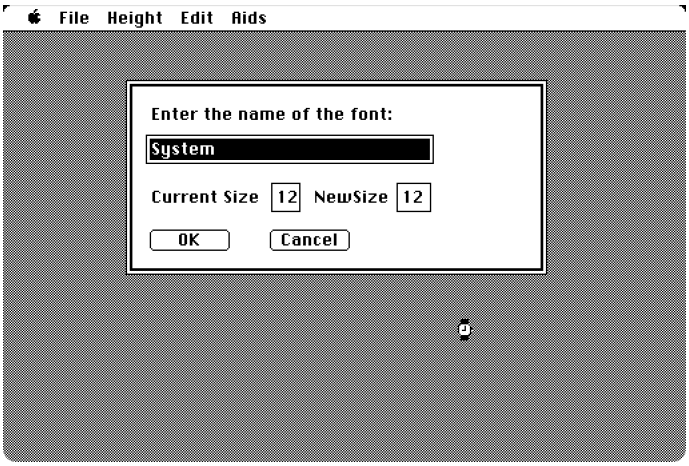
- **System Software 2.0** (from early 1985) with Finder 4.1
- **Font Editor 2.0** (from the Apple developer disk “5/85 MacStuff 2”)
- **Font/DA Mover 1.5** (from the disk “5/85 MacStuff 3”)
- **ResEdit 0.5** (from the disk “5/85 MacStuff 1”)
- **Fonts** (font suitcase containing a single font, Venice 14)
- **Starter Font Documents** (12 and 24 point copies of Geneva and New York, saved as font documents for use in Font Editor)
- **Mark’s Fonts** (a folder containing some of the fonts I made c. 1984-85 using Font Editor)

I’ve also prepared a 400K disk image which you’ll need to use if you have a real 128K Mac, since it can only read 400K disks. If you have a compatible external 800K drive or a drive emulator, such as the Floppy Emu from BMOW, you can use the 800K image, which includes most of Apple’s early fonts plus some of my own since there was enough space.

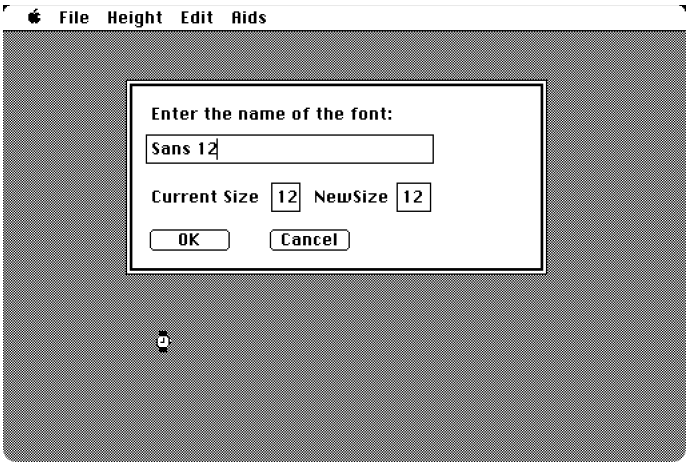
To make more space on the 400K disk, you may want to delete the 24 point starter font documents, which take up 8K each.

You don’t have to set it up exactly like this. The main requirements are System Software 2.0 with at least a few fonts like New York and Geneva installed, Font Editor, ResEdit 0.5, Font/DA Mover 1.5, and a Fonts suitcase containing at least one font, (preferably not Los Angeles, for reasons which will be covered below). And it must be an MFS floppy disk (for a real Macintosh) or MFS disk image (for Mini vMac).

Starting Up Font Editor



Don't click anything yet, and *don't* use the name "System".



Enter the name and size of an existing font document.

When you first start up **Font Editor,** you will be greeted by a dialog that says, "Enter the name of the font:" followed by a text field pre-populated with the name "System", fields for "Current Size" and "NewSize", and buttons for "OK" and "Cancel".

Do not click "OK" or "Cancel" or you will get a system error (the dreaded "bomb") and you will have to restart.

It's *not* possible to create a new font from scratch. You can *only* open and edit existing fonts. Apparently, when this app was created, Apple developers had some other way to make fonts prior to Font Editor 2.0.

When you launch the app, it's expecting is the **name of a font document,** along with the **current size** and **new size**, which allows you to *scale* an existing font to a new size. If you want to keep the size the same, then the current size and new size should match.

Note: In spite of the fact that it's the default name provided, for whatever reason, the name "System" is *not* allowed and will cause a crash if you try to use it.

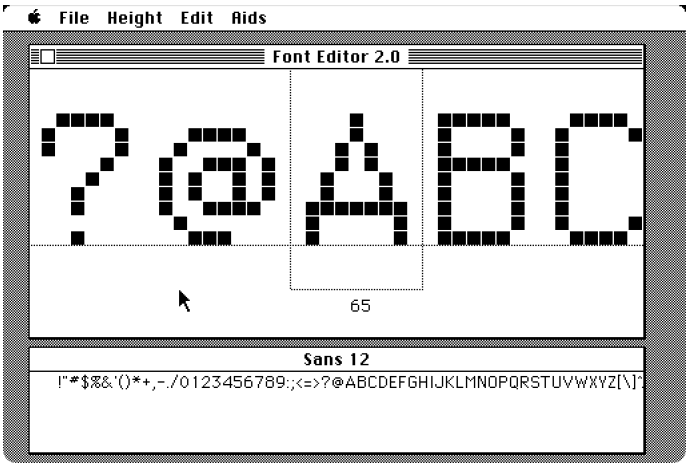
The font document, for which you provide a name, must already exist on the disk you're currently using. Font Editor does not follow normal conventions, like using an "open" dialog box to choose a file from the start up disk, or even switch to a different disk. You must know the name of the font document ahead of time.

I've provided a few starter font documents you can use: Sans 12, Sans 24, Serif 12, and Serif 24. These are just copies of Geneva and New York. If you want to save your work, I encourage you to rename them, either by duplicating and renaming them in the Finder or by using the **Save Font In...** command and providing a new name.

Be sure to type the name *exactly,* including spaces and capitalization. You will also need to type the size (24) in both size fields. If you type a larger or smaller size in the "NewSize" field, it indicates that you want to scale the font up or down (yes, this is possible—more on this later). Any mistakes here will cause a system error.

Click "OK" and the font will load.

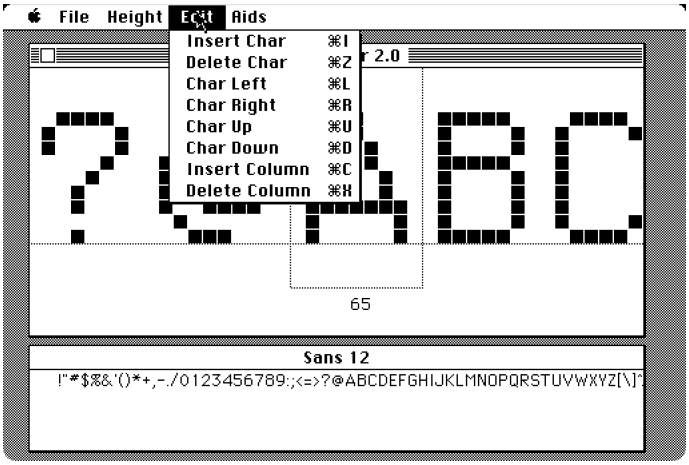
Note: If you make a mistake (assuming it doesn't crash), it will beep twice and display a blank document. You won't be able to do anything here, but that's okay. You can start over by choosing **Read Font...** from the File menu and enter the correct name and sizes.



Font Editor's main windows.

The Active Area

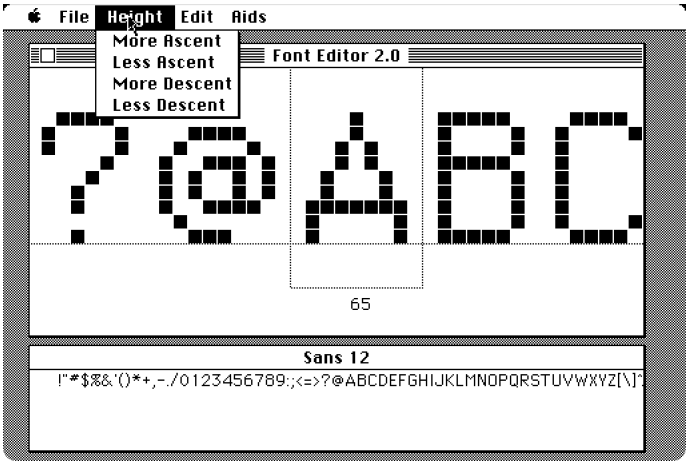
- The active editing area is indicated by the dotted rectangle in the middle of the large window. If you click inside this area, pixels will turn on and off. It's very similar to the Fat Bits mode in MacPaint.
- The dotted line along the bottom of the letters is the font baseline.
- The number below the active editing area is the character's ASCII number.
- If you click on any of the other visible characters, either in the large window or the small window, that character will become the active character. In this way, you can navigate through the entire font. (I guess it was too much trouble to implement scroll bars.)
- The close box (upper left corner) doesn't do anything.



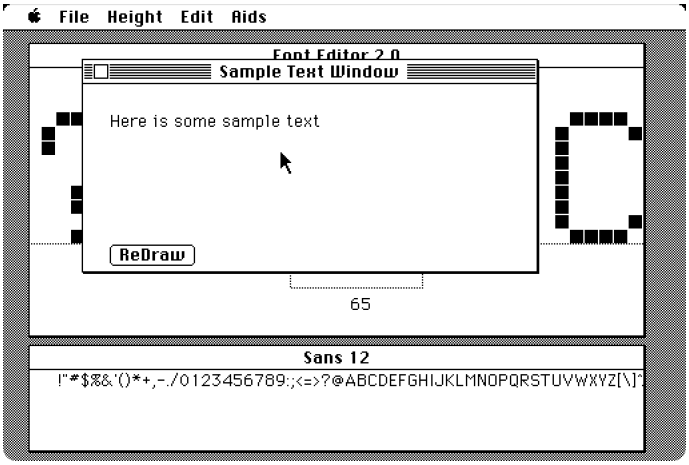
The Edit menu.

The Edit Menu

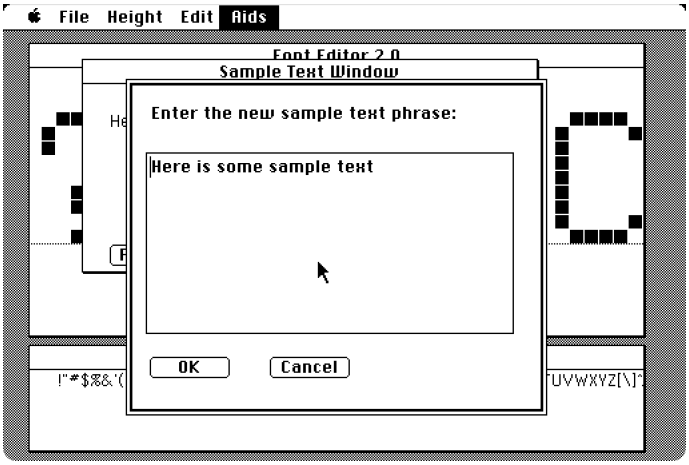
- **Insert Char.** You probably don't need this if you're editing an existing font like New York, which already has a complete character set. This is presumably for when you're starting from scratch, but see the next item for an example of when you might need it.
- **Delete Char.** This can be handy if you want to start over with a character. It deletes the active character. If you then do Insert Char (using the same ASCII code as the character you deleted), you need to provide an existing character's ASCII number from which to copy, to be used as the basis for the new character. Be careful: If you leave the "character to copy" field blank you will get a system error and have to reboot.
- **Char Left.** This shifts the character to the left by one pixel. Note: Any black pixels that cross the active rectangle boundary will be cut off.
- **Char Right.** This shifts the character to the right by one pixel. Note: Any black pixels that cross the active rectangle boundary will be cut off.
- **Char Up.** This shifts the character up by one pixel. Note: Any black pixels that cross the active rectangle boundary will be cut off.
- **Char Down.** This shifts the character down by one pixel. Note: Any black pixels that cross the active rectangle boundary will be cut off.
- **Insert Column.** This adds a column of pixels along the



The Height menu.



The Sample Text Window.



The Set Sample Text window.

left edge of the character box, shifting the rest of the character to the right and makes the character one pixel wider.

- **Delete Column.** This deletes a column of pixels from the left edge of the character box, shifting the character to the left and makes the character one pixel narrower. Note. Any black pixels in the deleted column will be lost.

The Height Menu

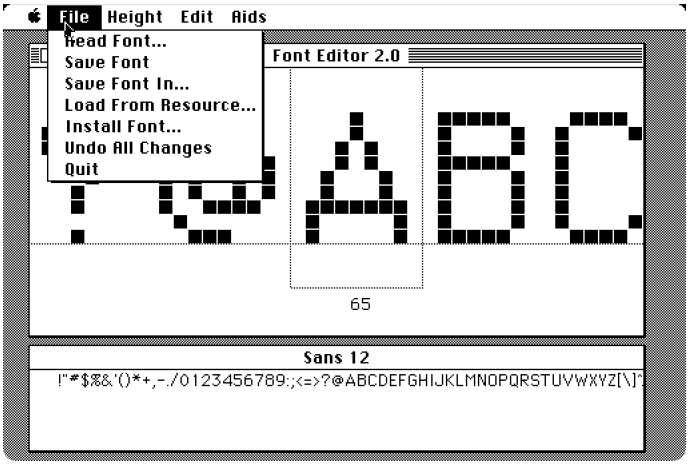
- **More Ascent.** Adds a row of blank pixels to the top of the entire font.
- **Less Ascent.** Removes a row of pixels from the top of the entire font.
- **More Descent.** Adds a row of blank pixels to the bottom of the entire font.
- **Less Descent.** Removes a row of pixels from the bottom of the entire font.

The Aids Menu

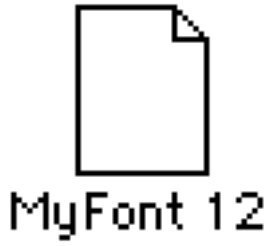
- **Show Sample Window.** Shows a sample of the font in its nominal point size in a smaller window. You can leave this window open while working in the other two windows, but it can't be resized. You may need to click the "ReDraw" button if it doesn't update to reflect the changes you've made. You can close this window with its close box (upper left corner) or by choosing **Hide Sample Window** from the Aids menu.
- **Set Sample Text.** This allows you to change the text in the Sample Window.

The File Menu

- **Read Font...** This allows you to open a **font document** that was previously saved using the **Save Font In...** command. Note: If there are unsaved changes in the currently open font, they will be lost with no warning. See also **Save Font In...** below for more info about this file format. If you specify a different size in the "NewSize" field, Font Editor will scale the font to that font size. This is a good way to start making a new size of a font. Note: Always use the actual size of the existing font in the "Current Size" field. Invalid values may lead to unpredictable results.



The File menu.



A *font document* has a generic icon and contains only one font in one size.

A *font suitcase* can contain multiple fonts for installation and distribution.

- **Save Font.** Writes the current font to disk, saving any changes you've made. This applies both to font documents opened using the Read Font... command *and* to fonts loaded from the System file using Load From Resource... .

- **Save Font In...** Writes the current font into a **font document** file on the disk with the name you provide. I would recommend using a name that specifies the font name and size, such as MyFont 12. If the file already exists, it will be overwritten with no warning. If you want the file to be in a folder, you will have to move it there in the Finder. These documents can only contain a single font in a single size and have a generic icon. They are only readable or writable using Font Editor, and cannot be read by Font/DA Mover.

- **Load From Resource...** Loads a font from the System file. The exact font name and size must be used. If not, you will get a system error and have to reboot. As with **Read Font...,** if you specify a different size in the "NewSize" field, Font Editor will scale the font to that font size. This is a good way to start making a new size of a font. Note: Always use the actual size of the existing font in the "Current Size" field. Invalid values may lead to unpredictable results.

- **Install Font...** adds the font to the **Fonts** suitcase on the disk, but be careful. It can overwrite existing fonts in the Fonts suitcase since it does *not* check for FONT ID conflicts. Font Editor assigns font number 12 to all new fonts. Do *not* use this command if you have used **Install Font...** with a different font you've made in Font Editor and have not yet chosen and assigned a font number other than 12 to it. Doing so will overwrite some or all of your earlier work. See FONT ID section below for more info about FONT IDs and font numbers.

- **Undo All Changes.** This does not seem to work. At best the app will hang, doing nothing but beeping when you click the mouse. At worst, it will cause a system error. In both cases, you have to reboot. **Do not use this command.**

- **Quit.** Quits the Font Editor app. If there are unsaved changes, they will be lost with no warning.

FONT IDs and Font Numbers

Font Editor, by default, assigns **font number 12** to any fonts created in the app, including new fonts or fonts derived from existing fonts. Font number 12 also happens to be the number for Apple's Los Angeles font. So, if you install your new font into the Font suitcase file and it containins Los Angeles, it will it will be a problem.

For example, you have a font suitcase containing Los Angeles 12 and 24. You make a new font, MyFont 24. Using Font/DA Mover to put MyFont 24 into that font suitcase will wipe out Los Angeles 24, and Los Angeles 12 will get its name changed to MyFont, even though it looks the same as before.

The same is true with fonts you have made previously, since they were *also* assigned font number 12. The key to avoiding this issue is to change the font number of your font from 12 to something else as soon as you install it into the Fonts file. In order to this, you will need to use **ResEdit.**

First, you need to pick a font number. The allowable range in the original font manager was 0-255. Numbers 1-127 were reserved by Apple, who actually only used a few. (See the list at left.) Don't use these.

Fonts are stored in **FONT** resources. If you open the System file in ResEdit, you will see that the **FONT IDs** do not match the **font numbers** at left. This is because the FONT ID is a combination of the font number and the font size. This is calculated by multiplying the font number by 128 and adding the size.

Each size of a font has its own FONT resource and FONT ID. The font name also has a resource with a FONT ID equivalent to what it would be at size 0.

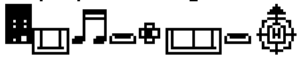
For example, London, size 18 has a FONT ID of 786. 786 – 18 = 768. 768 ÷ 128 = 6, the font number of London, as listed above left.

In short, FONT ID = (font number × 128) + size.

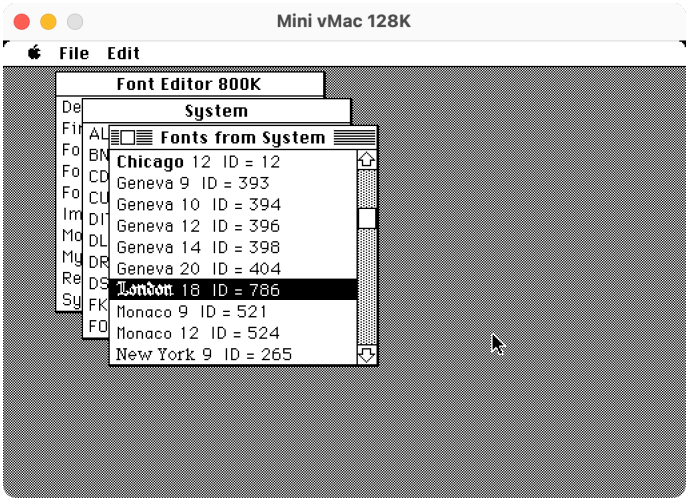
Let's say you have a font called "MyFont" and want to assign the font number of 44 to your font, and that you have 12 and 24 pt sizes. The FONT IDs are calculated as follows:

FONT ID for the name "MyFont" = (44 x 128) + 0 = 5632
FONT ID for size 12 = (44 × 128) + 12 = 5644
FONT ID for size 24 = (44 × 128) + 24 = 5656

That's all there is to it.

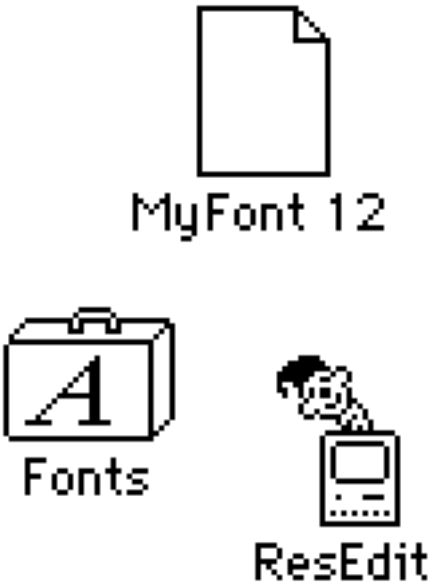
0 **Chicago**
1 application font
2 New York
3 Geneva
4 Monaco
5 **Venice**
6 **London**
7 **Athens**
8 **San Francisco**
9 Toronto
11  (Cairo)
12 *Los Angeles*
20 Times
21 Helvetica
22 Courier
23 Σψμβολ (Symbol)
24  (Taliesin)

Font numbers to avoid.



FONT IDs are not the same as FONT numbers, but they *are* derived from them.

Fixing the Font Number and FONT ID with ResEdit



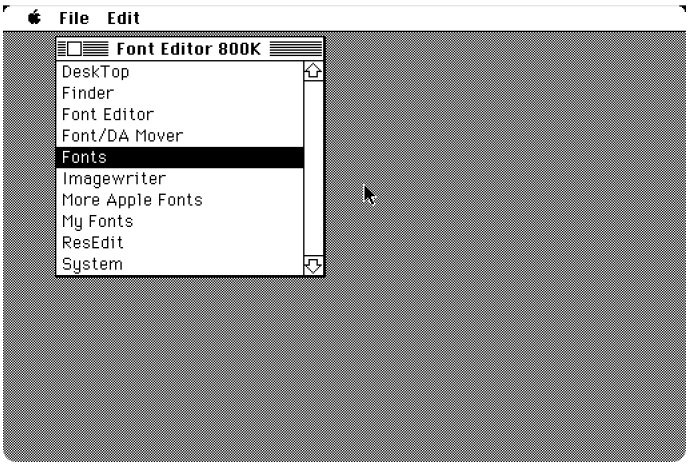
ResEdit to the rescue.

When you're working on your font, you usually want to save it using the **Save Font In...** command, which is similar to the Save As... command on modern computers. This creates a font document (not a font suitcase) that you can open with the **Read Font...** command.

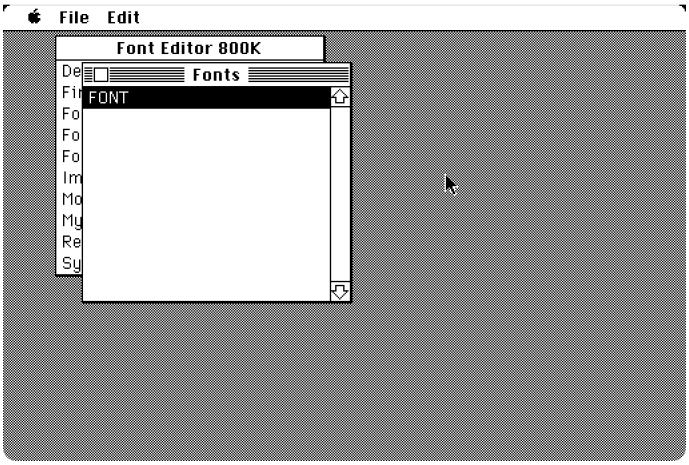
While a font document is open in Font Editor, the **Save Font** command will save changes to this file. However, to use the font on your Mac and move it around with Font/DA Mover, you have to install it into a font suitcase, namely, the **Fonts** suitcase file on your disk, which is the only place that Font Editor *can* install it.

You do this opening the font and then using **Install Font...** to install it into the Fonts suitcase file. By default, Font Editor assigns font number 12 to your font, so don't do this if your Fonts file already contains an earlier font for which you have not assigned a new font number or you may overwrite it. In this case, you need to fix the font number in *that* font first (using the following instructions, of course).

When you've finished adding your font to the Fonts file, quit Font Editor and open ResEdit. In ResEdit, you will see a list of all the files on your disk. Double-click on "Fonts".



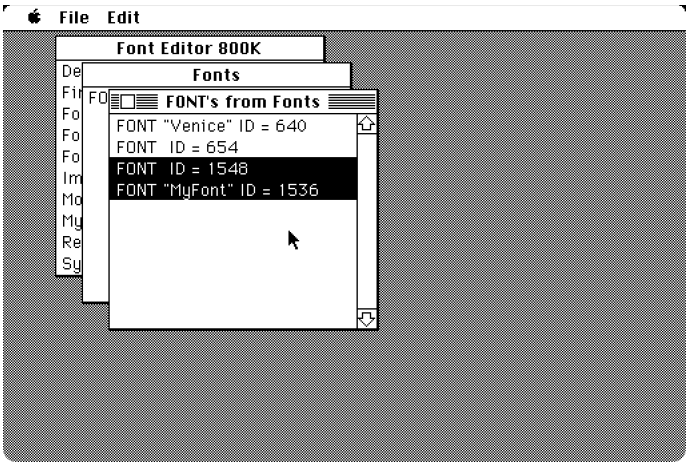
Open the "Fonts" file.



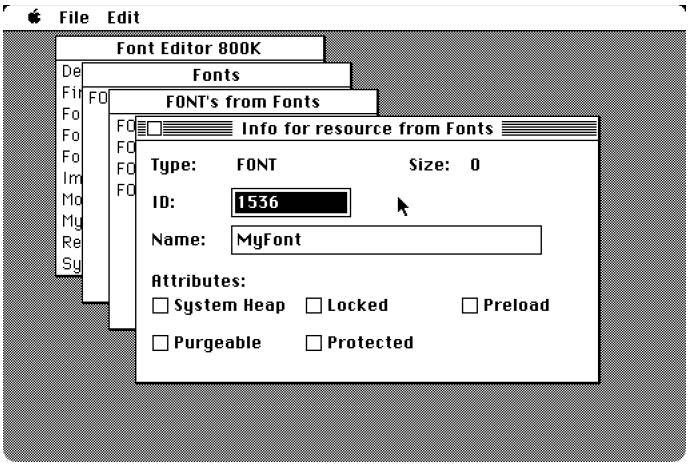
Option-double-click "FONT".

You should see a single resource item called "FONT". This contains all the font resources in the Fonts suitcase file.

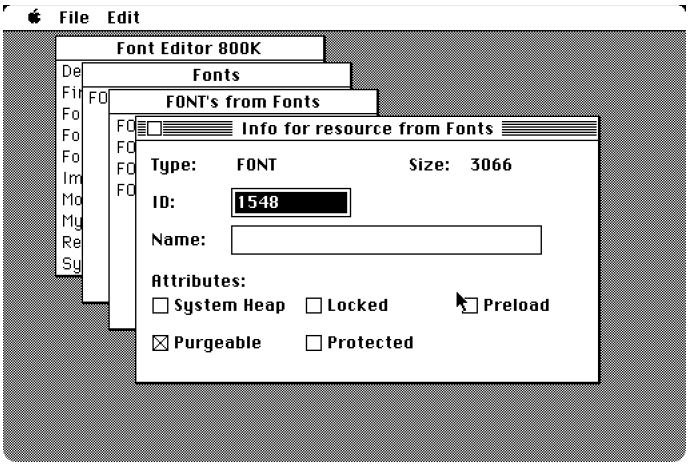
Instead of opening this by double-clicking, hold down the Option key first, *then* double-click. This gives you a more **detailed resource list** than you get without the Option key, and that's what you need for this operation.



The detailed resource list.



The name resource FONT ID for FONT number 12.



The bitmap data FONT ID for size 12 (1536 + 12 = 1548).

You will now see two or more entries which pertain to your font. In the screenshot example they are FONT ID = 1548 and FONT “MyFont” ID = 1536. The first one is the FONT ID for the *bitmap data* for font number 12, size 12 (1548), the second entry is the FONT ID for the *name* resource (1536) which links the name “MyFont” to FONT ID 1548. These are the values we need to change.

Let’s start with FONT ID 1536, the resource for the font name. Select it and choose Get Info (command-I) from the File menu.

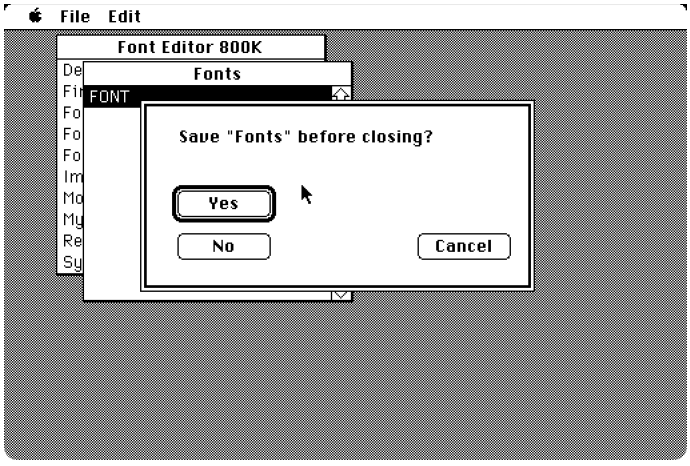
Let’s say you want to change the font number to 44. That means the FONT ID for the name resource will be 44 × 128 + 0, or **5632**. Enter this in the ID field and close the window. (Note: You can also change the name of the font here if you want.)

Next, select FONT ID = 1548 and do Get Info. The bitmap data resource FONT ID for this will be (44 × 128) + 12, or **5644**. Enter this in the ID field and leave the Name field blank.

If you have more than one size for your font, you’ll need to do this with each size of the font using the same formula.

Close all the windows and click “Yes” when asked to Save “Fonts” before closing.

That’s it—you’re done. You can quit ResEdit and install and use your font anywhere you like. (As long as it’s a classic Mac.)



Click the “Yes” button and you’re all done.

Closing Thoughts

Font Editor is *not* the tool you would use today if you want to make screen fonts for vintage Macs. I would recommend getting Fontastic Plus for that. It's better in practically every way and works on more machines and versions of the Mac operating system—even as late as MacOS 8—and supports many more font IDs and font families through the use of the FOND resource. And you don't have to deal with that font number 12 business.

My purpose in creating this how-to guide is to preserve a small but significant piece of Macintosh history. The Mac was the first affordable computer that went beyond the 40 or 80 column by 25 line text display, typical of personal computers in the early years. The Mac could display multiple proportional or fixed-width fonts in different sizes and styles *anywhere* on its screen. This was a radical new thing at the time.

Almost certainly, those early fonts were made using Font Editor, by people like Susan Kare and Bill Atkinson. With this guide, anyone can try it themselves and see what it was like.

There are still some unanswered questions, mainly...
Who wrote it? I poked around in the app using ResEdit and Fedit, and looked at stories of the original Mac development team at folklore.org, but have been unable to determine who wrote Font Editor. My guess is Bill Atkinson or Andy Hertzfeld, but it's only speculation.

In the first version of this how-to guide, I thought there must be a way to make a font from scratch—and I also couldn't figure out what it wanted the user to do at launch. I've come to the conclusion that Font Editor can *only* edit and modify existing fonts, and that it wants you to specify a font document to open when you first launch it. Obviously, the folks on the Macintosh development team must have had some other tool or method to make the first fonts. According to an article by Andy Hertzfeld at folklore.org, QuickDraw fonts first existed as plain data files, until he had the idea of storing them as system resouces (FONT). My guess is that the "font document" format that Font Editor reads *is* this earlier data file font format, and that Font Editor was used in part to convert them into FONT resources.

I hope people find this guide illuminating or even useful. If you have questions, corrections, or suggestions, feel free to contact me.

Mark Simonson, 5/31/24
marksimonson@mac.com