FileOpen 1.0
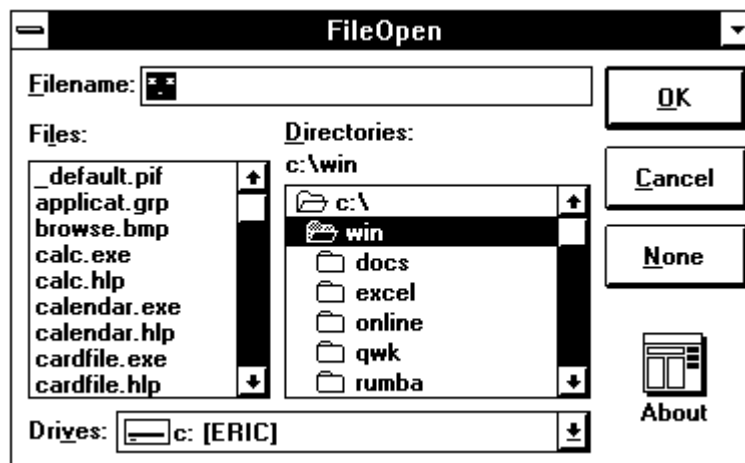## A GenEric Application

Introduction

Basically, FileOpen is a utility that allows you to load a document file along with a specified application, rather than loading the application, then loading a document.

The impetus for the application came from the amount of time it took to load WinWord and Excel, close the default, empty documents, and load the files I wanted in the first place. Many of you may spend most of your time creating new documents, spreadsheets, etcetera, but I find myself editing and reviewing more than creating. Communication programs like Crosstalk for Windows are probably better examples, since you probably call the same BBSes more than you create new entries. Whatever.

As you can tell from the image below, FileOpen is nothing more than a dialog box. That's it; nothing more. It consists of a filename edit box that supports drive and path specifications, as well as wildcards. The file and directory listings are just what they appear, as is the drive dropdown box. The OK button will load the filename specified, change directories, or change the file specification. The Cancel button does what it should, which is kick you out of the app. The None button is used for loading the desired application without any files, which means FileOpen won't get in the way too much when you do want to create a new document. The about icon will give a little message when clicked on.
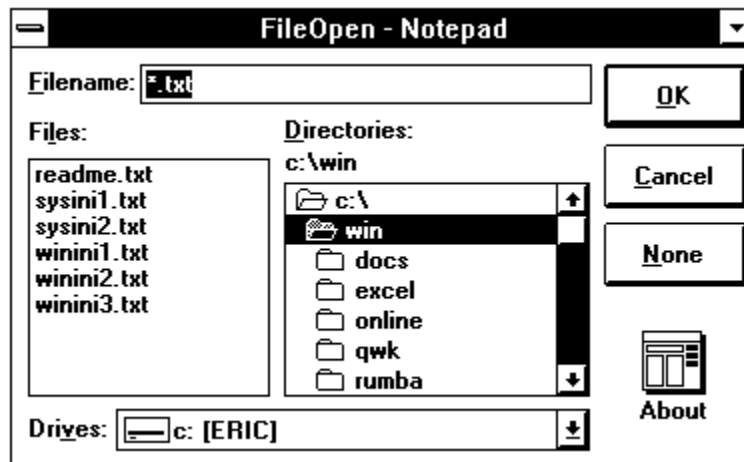
Loading FileOpen

There are three components to the FileOpen command line. Each of the three parameters are separated by a comma. If you exclude one of the parameters, FileOpen will default to it's internal settings, with the exception of the third parameter.

    1)   The title of the FileOpen window. I.E., "Word for Windows" or "Excel 3.0". Whatever you want in the title of the dialog box... If you leave this blank, the app defaults to "FileOpen".

    2)   The file specification and/or wildcards you want FileOpen to display on startup. I.E., "*.DOC" or "*.XL?". Whatever files you want displayed... If you leave this blank, the app defaults to "*.*".

    3)   The application you want loaded along with the selected file. I.E., "C:\WORD\ WORD.EXE", or "C:\EXCEL\EXCEL.EXE". Whatever app you want fired up when you're done with FileOpen. THIS PARAMETER IS REQUIRED! THERE IS NO INTERNAL DEFAULT!

Separate the parameters with commas. I.E., "FILEOPEN Notepad, *.txt, NOTEPAD" will display the following dialog, and load Windows' notepad utility with whatever file you specify.
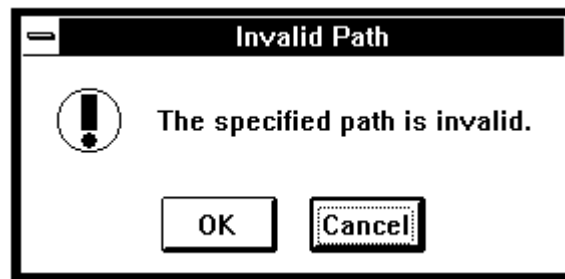

If you do exclude a parameter, you still need to put in a comma so FileOpen will know what to do. If you do something wrong, FileOpen will display an error message and you can figure it out from there.

General Usage

As stated in the introduction, FileOpen was written to be used with Program Manager, but I guess any menuing system would be able to utilize it. However, Microsoft seems to be the only one with a menu that doesn't have a similar utility built in, so it would seem redundant to use it with any other menus.

Within Program Manager, there are several minor annoyances that may or may not be corrected in future releases. One of 'em is the ability to select the document you want to edit with the app being called, but that's already been covered. Another is the way that ProgMan handles startup directories. Since FileOpen uses the current path as the default for it's directory list box, the optimum method of use would be to start FileOpen where your documents are.

To do this with Windows 3.0, you have two choices. One is to put a copy of FileOpen in each of your storage areas, which would seem suspiciously ineffective. The other is to lie to ProgMan when you create the entries for your applications. For example, if all your WinWord documents are in "C:\WIN\DOCS", but FileOpen is in "C:\WIN\UTIL", you can tell ProgMan to load "C:\WIN\DOCS\FILEOPEN parm1, parm2, parm3". At this point, Windows will say:

Choose the OK button anyway, and Windows will do the right thing. Trust me! Windows will change to the "C:\WIN\DOCS" directory, and then call FILEOPEN. As long as "C:\WIN\UTIL" or wherever is in your path, FileOpen will load, and show the files matching the specification in the current directory.

However, that's not the end of your woes with ProgMan... After putting a description and the command line into ProgMan's Program Item Properites dialog box, ignoring the error above, and selecting OK again, you'll be stuck with the plain jane icon. At this point, you'll need to change the item's icon, which you'll probably want to be the desired application for FileOpen to load, which is WinWord in the example directly above. To do this you simply need to point to "C:\WIN\DOCS\WINWORD.EXE" or wherever it is, and choose the OK button. Selecting the OK button in the Program Item Properties dialog will result in the above error message as before, but just ignore it again and you'll be set.

I know this seems like a lot of work, but hey, I didn't write Program Manager, so don't complain to me. Second, if this takes more time than FileOpen will save you, don't bother installing it in the first place, since you probably don't need it.

General Information and Legalities

In case you were wondering, FileOpen is being distributed as shareware. Actually, I hereby proclaim FileOpen as good for the free world, and am therefore placing it squarely in the public domain, along with all the source code and the icon. If you wanna send me some money to make yourself feel good, go ahead. That's what you worked so hard for, right?

Since I've given up all ownership, I can only make a couple of requests, and not bind you to anything legally. So if it's not too much trouble, here they are:

   1)   If you plan on uploading the package to any BBSes or other arenas, please make sure that all the original files are included. I want everybody to have the same opportunity you did, whether you utilized the code in an application or not.

   2)   If you use the code for another application, I'd like to maybe get a look at it. By the way, this utility was written in Microsoft's Visual Basic for Windows. The DLL included with FileOpen belongs to Microsoft and is copyrighted material, while FileOpen itself is not. Please pay full attention and respect to Microsoft's ownership of the dynamic link library.

If anybody wants to reach me, I can be found on the WINDOWS and OS2 FidoNet echoes. I don't hang out on CompuServe.

Regards,

Eric Hall
806 18th Avenue South, Apt 103
Nashville, TN 37203