

SYSEDIT2

'SYSEDIT2.EXE' is a (slightly) modified version of the 'MULTIPAD' demo program included with the Windows 3.0 Software Developer's kit. This is a rather handy little editor, and it can be clearly seen that it was this program which was used to develop the 'SYSEDIT.EXE' program that the Windows INSTALL program placed in the \WINDOWS\SYSTEM directory when you installed Windows 3.0 . This program was written primarily to correct a bug in the 'SYSEDIT' program, which will be discussed below. A later version may include a HELP file, but I do not plan on releasing the source files as it would probably be a violation of copyright. As it is, the program itself owes all of its design (minus a few modifications) to the Microsoft Engineers who developed it. I merely made the necessary changes to allow it to duplicate the functionality of the existing SYSEDIT.EXE that came with Windows .

Because this is not an 'original' work, but was taken from sample source files, I cannot claim any rights to this program (nor can anyone else, except Microsoft), so I am offering this free of charge in the public domain. As usual, the official disclaimer is 'any problems or loss of information which can be attributed to this software (i.e. SYSEDIT2.EXE) is not the responsibility of the author (myself and Microsoft), but of the user.' That is, use it as-is at your own risk guys and gals. However, I have had no problems with MY system because of it, and I ran the virus scanner yesterday and haven't downloaded anything since, so I can give you a very good impression about its likeliness to function without a hitch. Just make sure you to frequent backups, 'cause that is the best assurance against any data loss.

The 'Big Bug' in SYSEDIT.EXE

In case you were wondering what the BUG was (referred to in the 1st paragraph) it is easily demonstrated as follows:

- 1) Because I have alot of developer's software loaded, and it ALL needs a path to it, I have generated several logical drives ('I' 'J' and 'K', to be exact, for Microsoft 'C' 6.0, Windows Developer Kit, and Microsoft Windows). I did the 'SUBST K: E:\WINDOWS' command (my 'Windows' is on the 'E' drive 'cause it's a SCSI and I wanted speed for swapping) to get 'K:\' to be my Windows path.
- 2) Windows loads and runs just fine on the 'K:\' drive, and probably runs quicker that way (though I haven't bench-marked it yet). However, when I load 'SYSEDIT.EXE' it **does not read the '.INI' files in the 'Windows' path** (K:\) because it **tries to open them as 'K:\SYSTEM.INI', 'K:\\WIN.INI'** (etc.)!!!
- 3) Apparently Windows is returning it's path via the GetWindowsDirectory() function as 'K:\', which already has the '\' in it, and 'SYSEDIT.EXE' doesn't expect it to end in a '\\!!!
- 4) What I did to eliminate this problem is check for a '\' in the windows path (retrieved by GetWindowsDirectory()) and add one if it isn't there (*win_path[lstrlen(win_path) - 1]!='\\').

NOTE: It is recommended that you developers out there be careful of this pitfall!! Microsoft wrote the package and they STILL blew it!

To make a long story short, I fixed this little bug using MULTIPAD, and rename'd it as 'SYSEDIT2.EXE'. Not only that, but I kept the 'File Open' capability, and added a command line enhancement! If you specify a list of file names separated by spaces, as command line arguments, 'SYSEDIT2' will read each file and put them on the MDI desktop in the same way it already does with the 5 system files. FIVE files? Yep! I added 'PROGMAN.INI' to the list of files (SYSTEM.INI, WIN.INI, CONFIG.SYS, and AUTOEXEC.BAT, plus PROGMAN.INI).

I couldn't figure out a simple way to determine which drive the 'boot' drive was, so I made an assumption: it's the 'C' drive, no matter what. I know that it's POSSIBLE to set up any drive to be the boot drive, but I had to make some sort of an assumption. So, if you have your 'CONFIG.SYS' and 'AUTOEXEC.BAT' files somewhere besides on the 'C' drive root directory, you can specify a command line for SYSEDIT2.EXE as follows:

(from the WINDOWS directory:)

```
SYSEDIT SYSTEM.INI WIN.INI PROGMAN.INI X:\CONFIG.SYS X:\AUTOEXEC.BAT
```

(where 'X:\' represents YOUR boot drive root directory)

If you have any comments/questions/suggestions/etc. I occasionally visit the San Diego Computer Society BBS in San Diego. You can also leave EMAIL on the PCMAGNET (although I rarely get on, so I may not answer quickly).

Robert (BOB) Frazier
[pc magnet 70172,177]
7414 Mesa College Dr #32
San Diego, CA 92111
May, 1991