

What Price Multitasking? Optimizing OS/2® with an Eye Toward SAS® Software

John M. Martz, University of North Carolina at Chapel Hill

ABSTRACT

Power demands its price. As the most powerful computer operating system for the Intel platform, OS/2 exacts its due through complex system setup and optimization. While the novice user might feel overtaxed, often major performance gains can be achieved with minimal investment of time and effort. The current work provides insight into getting the most out of OS/2 by furnishing suggestions on installation, configuration, and maintenance, including — but not limited to — a brief discussion of formatting and layout of the hard drive, editing the CONFIG.SYS file, setting up work areas, and maintaining INI files.

INTRODUCTION

This paper provides the resources to optimize OS/2 to create a more productive SAS working environment through the use of OS/2's object-oriented interface and efficient multitasking. The following topics are addressed: installation, configuration, and maintenance of OS/2. Each topic includes suggestions you may follow to improve the performance of OS/2 in general and the SAS System for OS/2 in particular.

Since performance often varies widely among similar computer systems, you should consider information provided here as suggestions rather than hard guidelines. It is important that you monitor your system after implementing each change in order to determine its impact on performance (cf. Moskowitz & Kerr, 1995). Finally, you should always back up your data before making changes to your system files.

Unless otherwise noted, suggestions throughout this paper apply to OS/2 Warp 3.0; however, most also apply to earlier versions of OS/2. Other conventions include the following. Since OS/2 permits the redefinition of the mouse buttons, this document will conform to IBM's convention and refer to mouse button 1 (usually left) and mouse button 2 (usually right). Whenever an OS/2 menu selection, notebook tab, or other selection contains a "hot key" character — such as the Open selection on pop-up menus — the hot key will be underlined.

INSTALLATION

RTFM! People tend to rush into things. When installing OS/2, there are a lot of variables to consider. Don't rush into it. Take some time to familiarize yourself with the installation instructions *before* installing it. Take note of all your options and carefully consider your choices. While it will take longer than simply sticking the installation disk into A: and rebooting, planning your setup will help you avoid future pitfalls.

OS/2 "for Windows" or "Fullpack"? Actually, this is a trick question — OS/2 Warp comes in more than 2 flavors. As of this writing, OS/2 is available in the following additional flavors: Connect (includes networking features), SMP (supports multiple CPUs), and Power OS/2 (for the Power PC platform). Assuming that you aren't using a Power PC or an Intel-based PC with multiple CPUs, the most useful distinction remains between "for Windows" and Fullpack" versions even if you require the networking support from OS/2 Connect, because it also comes in both versions.

The only distinction between OS/2 "for Windows" and "Fullpack" (besides the lower price of the former) is how OS/2 handles Windows support. The former does not ship with Windows code; it uses the

copy of Windows (3.1 or later) currently installed on your machine to provide support for Windows applications. OS/2 "Fullpack" comes complete with Win-OS/2® code (IBM's recompiled and optimized version of Windows). If you currently have Windows installed or do not need Windows support, get the "for Windows" version; if you need Windows support and don't have a copy of Windows already, get the "Fullpack" version. Which OS/2 you choose makes no difference to the SAS System for OS/2, since it doesn't rely on Windows.

Minimal System. Forget the minimal system specifications on the box. While OS/2 can run on a 386 SX based PC with as little as 4Mb of RAM, you don't want to do so — especially when running the SAS System. Some recommend 6Mb (e.g., Moskowitz, 1995a), but after using the SAS System for OS/2 on 8Mb, 16Mb, and 20Mb systems, this author recommends you consider installing OS/2 only if you have at least 8Mb of RAM.

OS/2 is a virtual memory operating system — it can utilize more memory than is installed (Moskowitz, 1995a). When OS/2 requires memory, it "swaps" less frequently used code to the hard disk (into the file SWAPPER.DAT); when that code is required, OS/2 reloads it into memory. Since no disk drive approaches the speed of RAM, excessive swapping takes a toll on overall system performance. So, while 8Mb will provide a useful system, you should definitely buy as much RAM as you can afford (of course, this advice is not limited to OS/2). And then, if you expect to use the SAS System extensively, don't consider less than 12 or 16Mb.

Of course, you will need to have the space on your hard disk to accommodate both OS/2 (35-55Mb) and the SAS System for OS/2 (varies widely depending on what modules you install). In addition, Warp's BonusPak includes 30Mb when completely installed. Fortunately, you do not need to install the complete SAS System on your hard disk. Consider locating less used modules on a network drive or running them from the CD-ROM.

Dual Boot or Boot Manager? Assuming you wish to retain DOS support (i.e., you don't want to have OS/2 as your sole operating system), you have two choices in configuring your system. While most DOS and Windows programs can run in an OS/2 VDM (virtual DOS machine), there are a few that can not (e.g., programs that directly access the physical disk to perform low-level system calls). In addition, you might want to use your computer to run other operating systems, such as one of the many flavors of Unix. Both dual boot and boot manager setups provide the ability to boot into another operating system to run these programs.

Dual boot, which is the default installation of OS/2 Warp, places OS/2 and DOS on the same hard drive partition. With dual boot, you can reboot your computer into stand-alone DOS by simply clicking an icon. In this setup, OS/2 and DOS must reside in the same partition. This partition must be formatted with the FAT file system (see below), it must be a primary partition (i.e., it can't be a logical partition) on the first hard disk of the computer, and it must be C:.

Boot manager, on the other hand, requires a bit more work to set up, but it is worth the effort. It provides much more flexibility, such as the ability to boot multiple operating systems (e.g., UNIX, different versions of DOS, Windows NT, etc.) from the same machine, as well as the ability to isolate different operating systems from each another. Using Boot Manager is also the only way you can place OS/2 on a partition formatted with HPFS.

What File System? OS/2 ships with two file systems: FAT and HPFS. FAT (file allocation table) is the file system used by DOS; HPFS (high-performance file system) is native to OS/2.² Compared to HPFS, FAT uses less overhead. But it does not perform as well on larger drives (drives greater than 50Mb or so) and retains the archaic 8.3 DOS naming convention. While HPFS-formatted partitions are "invisible" to stand-alone DOS, most DOS/Windows programs can run from within OS/2, so this limitation has relatively no impact unless you need to spend a great deal of time outside OS/2.

Besides offering performance gains, HPFS does not waste as much disk space as FAT, it minimizes file fragmentation, provides for enhanced data recovery, and it supports long file names up to 254 characters, including blanks (SoftTouch Systems, Inc., 1994; Tyson, 1994). Furthermore, HPFS directly supports extended attributes. Extended attributes are additional information about a file such as file type, comments, and icon. While OS/2 does provide a means of retaining extended attributes on FAT volumes, it must store them in a separate file (EA.DAT.SF) — sacrificing performance and causing the potential for serious mishap (should the EA file be deleted or corrupted). Thus, unless you have a strong need for using FAT as your sole file system, you should consider making HPFS your primary file system, relegating FAT to the smallest portion necessary (Parsons, 1995).

Figure 1: Sample Hard Disk Partitioning.

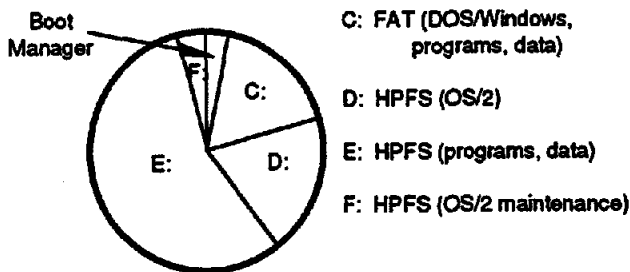


Figure 1 presents just one way you might partition your hard disk into smaller volumes and mix both file systems. Here, you can use either the "for Windows" or "Fullpack" versions of OS/2. But for the sake of illustration, it will be assumed that you have installed the "for Windows" version of Warp, not wanting to waste disk space on what would be essentially two separate copies of Windows. This approach is probably most reasonable, since you can place your DOS/Windows setup on its own FAT partition in the event that you need to boot into stand-alone DOS. In addition, your Win-OS/2 and stand-alone Windows sessions will be "synchronized" because any changes made to the Program Manager during either session affect the same copy of Windows (cf. The Cobb Group, 1994). Note, however, that when you do boot into stand-alone DOS, you will not be able to access D:, E:, and F:.

Installing OS/2 in its own partition makes the installation of new versions or the reinstallation of the current version easier (should your system become hopelessly corrupted — a rare occurrence, but one worth considering). For example, it is often desirable to reformat the partition as a means of removing outdated system files that might not be removed by the installation program. Of course, reformatting a partition destroys *all* the data on that partition, so make sure to back up any files you wish to retain (e.g., custom fonts, bitmaps, etc.).

You should install the SAS System for OS/2, and all other native OS/2 programs, on E:. You should even install DOS and Windows programs on E: unless you will need to access them from stand-alone DOS or they do not run under OS/2 (in which case, you would need them installed on a FAT partition). Remember, when you run a DOS/Windows program from an OS/2 VDM, the program is able to read HPFS formatted drives (it just won't be able to read files with

long file names, such as "SESUG95_paper"). It is just when you boot into stand-alone DOS that programs can't "see" HPFS drives.

A useful precaution is to construct a 3 to 5Mb maintenance partition (F: in Figure 1). OS/2 Warp allows you to easily create three utility disks. These disks provide the bare minimum OS/2 configuration and are used to run utilities, such as CHKDSK.COM, on the OS/2 boot partition that require the volume to be locked. Since OS/2 maintains open files as long as it's running, your only option is to boot from another drive to perform these actions on the boot drive, and it is much quicker and easier to boot and perform maintenance from the hard drive than from a floppy drive. Make this partition bootable and add it to the Boot Manager menu (just as you do with C: and D:). Copy the entire contents of the utility disks to this partition. You should also copy third-party maintenance and recovery utilities (e.g., GammaTech's HPFSOpt and GTDisk) to the maintenance partition.

CONFIGURATION

CONFIG.SYS. You will find this file in the root directory of your boot drive. It contains configuration information that OS/2 requires to boot your system. While OS/2's defaults are often adequate, there are some commands that you should consider modifying. Several will be discussed briefly.³ For more detailed information, you are encouraged to obtain CNFGINFO.EXE (Meigs, 1995). This free-ware program loads your CONFIG.SYS file into one window, provides exceptionally complete information (including suggestions for optimization) on nearly every CONFIG.SYS command with the click of the mouse, and allows easy editing of the file.

REMEMBER TO BACK UP YOUR CONFIG.SYS FILE PRIOR TO EDITING IT! The data you save may be your own.

BUFFERS=70

The BUFFERS line indicates how many 512-byte blocks to reserve as buffers. Increasing this number can improve performance in both DOS and OS/2 sessions. However, since a cache is more effective than buffers, you should divert more resources to caches. If you are running an all HPFS system, try setting this parameter to 3 (any less, and OS/2 will default to 30).

DISKCACHE=256,32,LW,AC:C

DISKCACHE establishes a cache for FAT-based disks. If you are not using FAT, save the memory and either set the cache to 64 kilobytes or comment out this line by placing REM in front of it. While you can FAT create a larger cache, do not increase it beyond 512 kilobytes unless you are running a FAT-only system, as any performance gains will be outweighed by the loss of available memory. A cache of 256 kilobytes might provide optimal performance. The second switch sets the threshold size for caching; the maximum record size to be cached. LW enables lazy write for FAT and is similar to the lazy write option for HPFS drives — don't turn it off (see below). The final switch turns on autochecking of the listed drives should OS/2 be improperly shut down. When the system is rebooted, OS/2 performs CHKDSK /F on each FAT drive listed.

IFS=D:\OS2\HPFS.IFS /CACHE:1024 /CRECL:64 /AUTOCHECK:DEF

This line installs the HPFS file system. If you are not using HPFS, save the memory and comment out this line by placing REM in front of it. The first parameter (/CACHE) specifies the size of the cache in kilobytes to be used as a temporary holding area for data being read from or written to the hard disk. If you have enough memory to run the SAS System, you should probably set the cache to 1024 or higher (the maximum is 2048). The second switch (/CRECL) sets the maximum record size to be cached. The final parameter (/AUTOCHECK) indicates which HPFS drives are to be automatically checked in the event that OS/2 is improperly shut down. When the system is rebooted, OS/2 performs CHKDSK /F:2 on each HPFS drive listed.

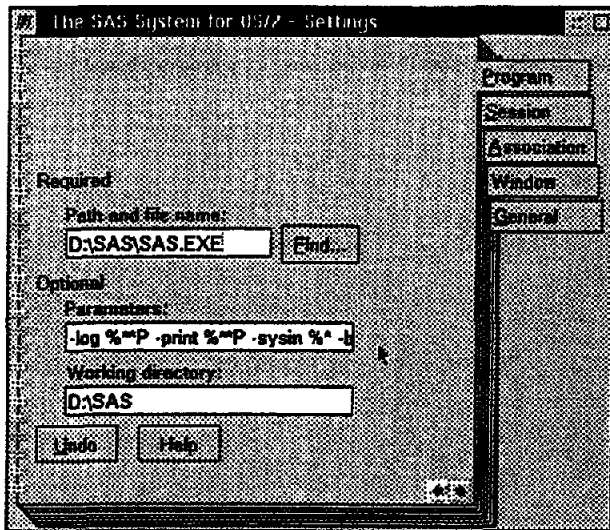
LASTDRIVE=F

While not normally included in CONFIG.SYS during installation, use LASTDRIVE to indicate the maximum number of drives on your system. It saves a small amount of memory (Dvorak, Whittle, & McElroy, 1993).

MAXWAIT=2

MAXWAIT defines the maximum amount of time a thread has to wait before OS/2 increases its priority. The default is 3. You can improve system performance by increasing this number to 2 or even 1. But be aware that increasing this parameter costs system resources.

Figure 2: Specifying Program Information in the Settings Notebook.



```
SET PATH=D:\OS2;D:\OS2\DLL;D:\OS2\SYSTEM;
D:\OS2\MDOS;...
```

The PATH list indicates directories that OS/2 should search to find programs. Related commands are LIBPATH, SET DPATH, SET BOOKSHELF, and SET GLOSSARY (which won't be explained here). In general, you should list the most frequently accessed directories first. Keep this line as short as possible; the longer the list, the more directories OS/2 must search. You shouldn't add a directory to this list unless you need to access its contents from the command line (and even then, you might not want to add it). When setting up program objects in the Workplace Shell®, use the settings notebook to set up the program's working directory (Figure 2). First, click mouse button 2 on the program object and select **S**ettings from the pop-up menu. On the first page of the notebook (Program), move to the Working directory field. Enter the program's working directory.

PAUSEONERROR=YES

Normally, this line isn't included in the CONFIG.SYS. Setting this parameter to yes causes the boot process to pause and wait for user input if the system encounters an error while processing the CONFIG.SYS file. It is a good idea to include this line when tinkering with your CONFIG.SYS file or making changes to your system in general.

PRIORITY_DISK_IO=YES

This parameter toggles whether OS/2 gives disk input/output associated with the foreground task greater priority over background tasks (YES or NO). Unless you have a reason for equating background and foreground tasks (e.g., if you are running a BBS in the background), you should leave this parameter set to yes.

PROTECTONLY=NO

Normally, OS/2 reserves memory for DOS sessions. If you are not

using DOS or Windows programs, set this parameter to yes. It saves memory by disabling DOS and Win-OS/2 sessions.

REM SET DELDIR=D:\DELETE,512;C:\DELETE,512;...

This parameter is OS/2's way of protecting against accidental erasure of files. When enabled (by removing the REM from the above line), deleted files are copied to the specified directory and maintained there until the size of the files exceeds the number of kilobytes specified. When this number is exceeded, files are removed on a first-in first-out basis. Unfortunately, having to copy deleted files to the delete directory slows down system performance. Furthermore, given that the SAS System writes temporary datasets to disk and deletes them at the end of a session, the delete directory inevitably becomes filled with these files with even modest usage of the SAS System. For protection against accidental file deletion, you are probably better off considering a third-party utility, such as GammaTech's Undelete utility. Of course, a frequent, regular backup schedule is also an important part of safe data management.

```
RUN=D:\OS2\CACHE.EXE /LAZY:ON /MAXAGE:40000
/DISKIDLE:30000 /BUFFERIDLE:20000
```

If you are using HPFS, do not turn lazy write off! If this parameter does not appear in your CONFIG.SYS, that is okay — lazy write is on by default. If you have lazy write off, turn it back on. Lazy write improves system performance (Parsons, 1995). Data written to the cache is not written to the hard disk immediately. Rather, it remains in the cache until an idle period of disk activity occurs within the limits specified by /MAXAGE, /BUFFERIDLE, and /DISKIDLE. /MAXAGE defines the maximum amount of time frequently written data can reside in the cache before being written to disk. /BUFFERIDLE sets the amount of time the cache buffer can be idle before its data is written to disk. And /DISKIDLE defines the amount of time the disk must be idle before accepting data from cache memory.

SET KEYS=OFF

This line turns off the command line history buffer for OS/2 sessions. If you use the command line rarely, set this parameter off.

SWAPPATH=D:\ 5120 10240

Move the swap file (SWAPPER.DAT) from its default location, D:\OS2\SYSTEM (where D: is your boot drive) to the root directory of the *most used* partition on the *least used* drive on your system (or, if you only have one hard drive, move it to the root directory of the most used partition) and edit the SWAPPATH to indicate the change. This location minimizes the distance the disk head has to move when data is swapped to disk. After you reboot, make sure to delete the old SWAPPER.DAT file and reclaim the disk space.

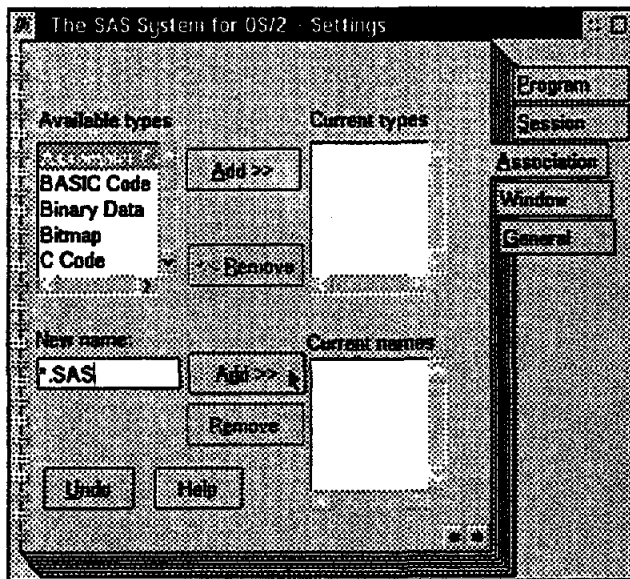
The second switch of the SWAPPATH command indicates when OS/2 should begin warning you that you are running out of space on the swap partition (the partition where SWAPPER.DAT is located). The number represents the number of kilobytes remaining on the disk. Increase it to provide early warning and give you time to close some applications or erase some files before you run out of space. If you run out of disk space, programs and OS/2 may begin to fail (Parsons, 1995). The last parameter is the initial size of the swapper file in kilobytes. You should monitor the size of your swapper file over the course of an average work day and note its maximum size — the IBM Employee Written Software program DINFO is a simple program designed to accomplish this task (Christensen, 1992). OS/2 must divert resources to resize (either grow or shrink) SWAPPER.DAT, so for best performance, you should allocate a swapper file that exceeds the maximum daily usage. Reducing the number of times OS/2 must resize the swapper file not only spares system resources, but it prevents SWAPPER.DAT from fragmenting (especially problematic on FAT volumes).

Associations. With OS/2, you have the ability to link data files to programs. Once an association is established, the name of the associated application appears on the **O**pen submenu of all matching data objects and the data object's icon will change to that of the

application's icon. If it is the only association (or the first) established with that data object, it will become the default Open action. For example, assume you want to set up an association with all your SAS programs, so that the default Open behavior is to execute the program as a SAS batch job. Click mouse button 2 on the SAS System for OS/2 object and select Settings from the pop-up menu. Select the Association tab of the notebook (Figure 3). Type the following in the New name field:

*.SAS

Figure 3: Creating an Association.



Press Add (note the d, as there are two Add buttons on the Association page). The text you typed should appear under the Current names list. Close the notebook. The association is now established, all files ending in *.SAS will bear the SAS System icon, and double-clicking mouse button 1 on any of them will submit that program for execution as a SAS batch job.⁴

Work Areas. One of OS/2's most useful features is the ability to create work areas. Work areas are folders with two unique properties. First, when you close a work-area folder, all open objects within that folder close. When you reopen the work-area folder, all objects that had been open reopen using their default behavior.⁵ Besides reducing the amount of work required to open and close a set of objects, this feature allows you to easily halt work and pick up where you left off. Second, when you minimize the work-area folder all open objects in that folder minimize; when you "show" the folder again, all minimized objects are redisplayed. This feature provides you the ability to clear your screen quickly, should you need quick access to the desktop.

For example, assume you decide to create a work area for each research project, paper, and presentation you work on, and you want to start by creating a work area for your SESUG'95 paper and talk. First, you create a folder as usual — either by dragging a Folder template from the Templates directory or using the MD command from the command line — and you name it SESUG95. To make this folder a work area, you click mouse button 2 on the SESUG95 object and select Settings. Select the File tab and click on Work area to place a check mark in the appropriate box (see Figure 4). Close the settings notebook. The folder is now a work area. You can place all material germane to the project into the work folder: data files; folders containing SAS datasets, programs, and correspondence; shadows of frequently accessed programs

(e.g., the SAS System for OS/2), drives, and folders — basically, you can include an object pointing to every relevant piece of information accessible by your computer.

SAS Options. Several options can increase your productivity while working within the SAS System for OS/2. A few are discussed below.

-BUFNO 5

Increase the number of buffers to use for SAS datasets. The default is 1. You should try increasing the number of buffers to speed up DATA steps. However, remember that increasing the number of buffers increases the demands on your system. Too many buffers will hurt overall system performance.

-BUFSIZE 0

You can specify a permanent buffer size for output datasets by changing this parameter; however, the default of 0 is probably the best choice for average usage. Specifying 0 forces the output engine to select the buffer size based on the number of observations in the dataset.

-CBUFNO 3

Similar to -BUFNO, this option specifies the number of buffers to allocate for processing each open SAS catalog. The default is 0. Increasing the number of catalog buffers can reduce I/O actions and improve CPU performance.

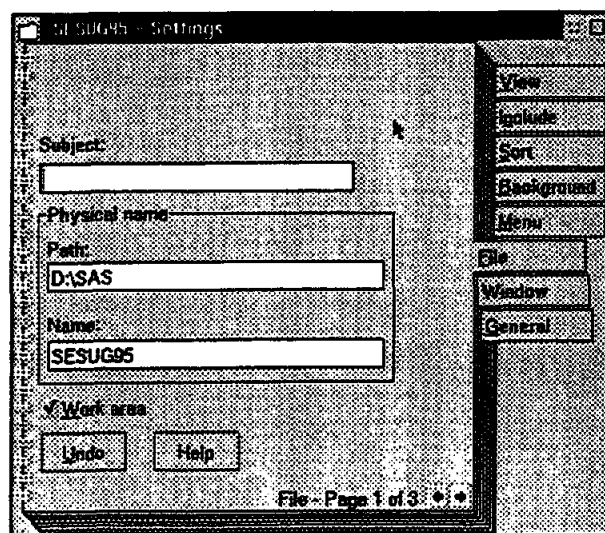
-LOG and -PRINT

Probably the first thing you will notice when submitting a batch job via the Workplace Shell is that the LOG and LST files are saved to SAS's home directory. While you can move these files to the appropriate directory each time after you are finished, there is a much simpler method. Use the -log and -print options to redirect the output. For example, assume you have a central directory called J:\REVIEW where all programmers are to copy the LOGs and LSTs of their projects. Assuming you already have an association established between your SAS program and the SAS System for OS/2, do the following. First, click mouse button 2 on the SAS System for OS/2 object and select Settings from the pop-up menu. On the first page of the notebook (Program), move to the Parameters field and enter the following:

```
-LOG J:\REVIEW -PRINT J:\REVIEW -SYSIN %*
```

Above, the %* parameter indicates that the file's name and extension (if any) should be passed to the executing program.

Figure 4: Creating a Work Area.



It might be more useful, however, to write the LOGs and LSTs to the same directory as the source file, especially when you are running multiple batch jobs simultaneously. While you could specify a different directory each time you launch a batch job, OS/2 permits a much more elegant solution. Repeat the above procedure, but type the following in the Parameters field as in Figure 2:

```
-LOG %**P -PRINT %**P -SYSIN %*
```

The %**P parameter passes the program's path to the SAS System. Now, whenever you launch a SAS batch job, the LOG and LST will be redirected to the same directory as the SAS program. You can also write copies of the LOG and LST files with the -altlog and -altprint parameters. For example, you can combine the two previous parameter strings:

```
-LOG %**P -PRINT %**P -ALTLOG %**D\REVIEW -ALTPRINT  
%**D\REVIEW -SYSIN %*
```

Not only will the output from each program be redirected to the directory containing the SAS program, but copies will be written to the \REVIEW directory of the drive on which the program resides. Here, %**D inserts the drive letter and colon (e.g., E:) into the parameter string passed to the SAS System.⁶

-NOLOGO

The SAS System, Version 6.10 displays an attractive logo at the start of each session, including batch jobs. After a dozen or so jobs, you are probably ready to do away with it and divert those computing resources to other tasks. This option gives you the power.

-REGISTER "--CALCULATOR" "PMDCALC"

You can place operating system commands and/or external programs into the File pull-down menu. The first parameter provides the menu title as it will appear; the optional tilde indicates which letter to use as a hotkey for the new menu selection. The second parameter is the command or complete path of the program (no path need be specified when the program is located in a directory specified in the PATH statement of the CONFIG.SYS file).

-SYSLEAVE 48K

This option specifies how much memory to allocate for normal exit. The speed with which an interactive SAS session exits can be altered by increasing this parameter from the default 16 kilobytes. Experiment with increasing this number (the maximum value is 64K).

-WORK J:\SASWORK\

The SASWORK directory is the SAS System's scratch pad. Moving this directory to a different physical hard disk can improve performance. The logic is similar to the justification for moving SWAPPER.DAT discussed above. If you place SASWORK on a separate drive from the SAS System, you will reduce the amount of repositioning of the primary drive's read/write head. Locating the work directory on the least accessed drive will reduce the repositioning required to read from and write to SASWORK, and it will diminish the likelihood of fragmenting the files in SASWORK. You might consider placing SASWORK on a virtual disk (using the VDISK command); however, doing so is not recommended unless you have a lot of memory, so it will not be discussed further.

Multiple SAS Program Objects. Of course, you can place some of the preceding options in your CONFIG.SAS file; however, it may be more useful to place them — and others — in the settings notebook. First, when viewing and modifying options, it is more convenient to do so from the desktop (assuming you have the SAS program object on your desktop) using the settings notebook. But more importantly, you can set up multiple SAS program objects, each with different options as specified in the parameters field. For example, you might create two SAS program objects with differing line sizes, one for formatting output to print in portrait orientation and the other landscape. Or you might customize an object for batch jobs and another for interactive sessions — note, make sure

that the former has an association with the *.SAS extension and the latter does not.

MAINTENANCE

INI Files. OS/2 stores system information (e.g., printer definitions) in the OS2SYS.INI file and user preferences (e.g., desktop background, window positions) in the OS2.INI file, which is often called the User INI file. After extended usage, these files — especially the OS2.INI file — tend to accumulate a lot of outdated information, such as references to deleted programs, files, and directories. Given the frequency with which the SAS System can generate and destroy temporary datasets and folders, the amount of clutter can accumulate quickly. This unnecessary junk degrades system performance. Since these INI files are binary, you can't load them into a text editor and edit out unnecessary lines, and OS/2 does not ship with a utility to address this problem.

Fortunately, there are several products available to address the problem. Probably the most well-known is INI File Maintenance (Carry Associates, 1994). Actually, one of three closely related programs that vary in the extensiveness of their capabilities, this program provides an exceptional wealth of information about the contents and operation of INI files, and it is simple to use.

Disk Optimization. Unlike FAT partitions, HPFS volumes are extremely resistant to file defragmentation (i.e., when a file is written in non-adjointing sectors). Indeed, since the HPFS driver attempts to allocate contiguous sectors to files, some optimization can be performed by simply moving the files off the volume and then back onto it. However, files can still become fragmented on HPFS volumes.

In general, you probably want to avoid using DOS utility applications on any drive "touched" by OS/2 — especially on FAT partitions under stand-alone DOS. Rather, you should consider investing in a native OS/2 package. The author is only familiar with GammaTech Utilities; however, there are other packages available both commercially (e.g., The Graham Utilities) and as freeware (e.g., Kimes, 1995). Most of these packages also include additional useful utilities (e.g., sector analysis, file attribute manipulation, undelete, etc.).

CONCLUSION

OS/2 provides an optimal setting for working with the SAS System. The suggestions made here are not intended as a complete list of all possible improvements. Rather, they provide basic guidance on the installation, configuration, and maintenance of OS/2 and the SAS System for OS/2. While these suggestions might prove sufficient for your purposes, you will be doing yourself a disservice if you do not consider other sources and applications of OS/2 — for example this article has not addressed REXX (an easy-to-use, powerful programming language included in OS/2). In addition to the references already cited, utilize the significant resources of the on-line community. Here is just a sample of useful sources of information on OS/2 and/or the SAS System on the internet:

ftp sites:
hobbes.nmsu.edu
ftp-os2.cdrom.com

web pages:
<http://www.teamos2.org>
<http://www.cyberblue.com/>
<http://www.sas.com/>
<http://www.marshall.edu/~bordner1/os2.html>
<http://www.austin.ibm.com/pspinto/os2.html>
http://www.yahoo.com/Computers/Operating_Systems/OS_2/

on-line magazines:
Inside OS/2 at <http://www.cobb.ziff.com/~cobb/os2/9506/index.html>

OS/2 Warp Monthly Newsletter at
<http://www.austin.ibm.com/pspinfo/mnl/volume1.html>

Usenet news groups:
comp.os.os2.apps
comp.os.os2.misc
comp.comp.os.os2.setup
comp.soft-sys.sas

In addition, there are OS/2-centric forums and libraries on the major commercial on-line systems (e.g., CompuServe).

ACKNOWLEDGMENTS

The author gratefully acknowledges the following individuals for useful suggestions in response to a post in comp.soft-sys.sas: Thomas G. Abemathy, Richard Denby, Phil Mason, and James Ssemakula. Also, gratitude is extended to those who's suggestions were not included due to space and time limitations: Jugdish K. Mistry, Kevin Thompson, and Steven C. Velasco.

ENDNOTES

¹"Read the F—ing Manual!" — a common "flame" on the internet. Avoid being the target by consulting the documentation first.

²Actually, if you use OS/2 to format a FAT volume, you are using "Super-FAT", which is faster than the ordinary FAT that comes with DOS. However, HPFS is still faster (Tyson, 1994).

³Most of the suggestions on tuning CONFIG.SYS are fairly common in the OS/2 literature. Since the author based the bulk of his suggestions on Meigs (1995), Moskowitz (1995b), and Tyson (1994) — only additional sources will be noted in the text.

⁴The icons in any open folders might not change immediately — you might have to close and reopen these folders first. However, if you've successfully established the association, the SAS program file's default open behavior (i.e., as a SAS batch job) will take effect immediately.

⁵Assuming you've established an association between *.SAS and the SAS System for OS/2, note that if you were editing a SAS program when you closed the work-area, it will be launched as a SAS batch job when you reopen the folder.

⁶For a complete list of the % parameters, search OS/2's on-line help for the string "% parameters" (without the quotes).

REFERENCES

Carry Associates (1994). INI File Maintenance (Version 3.3j) [computer software]. Available FTP: hobbes.nmsu.edu Directory: /os2/wpsutil File: inimt33j.zip

Christensen, N. M. (1992). DINFO.EXE (Version 1.30) [computer software]. Available FTP: ftp-os2.cdrom.com Directory: /4/os2/ibm/news File: dinfo.zip

The Cobb Group (1994). Synchronizing your Windows and WIN-OS/2 program groups. *Inside OS/2*, 3(7), 1-4.

Dvorak, J. C., Whittle, D. B., & McElroy, M. (1993). *Dvorak's Guide to OS/2 Version 2.1: Learn to Navigate the Operating System of the Future*. New York: Random House.

Kimes, M. (1995). FM/2 Utilities [computer software]. Available FTP: hobbes.nmsu.edu Directory: /os2/diskutil File: fm2utils.zip

Meigs, R. (1995). CNFGINFO.EXE (Version 4.0a) [computer software]. Available FTP: hobbes.nmsu.edu Directory: /os2/editors File: cfinfo4.zip

Moskowitz, D. (1995). Installation issues. In Moskowitz, D., & Kerr, D. (Eds.). *OS/2 Warp Unleashed, Deluxe Edition* (pp. 1-47). Indianapolis, IN: SAMS Publishing.

Moskowitz, D. (1995b). System configuration, setup, and tuning. In Moskowitz, D., & Kerr, D. (Eds.). *OS/2 Warp Unleashed, Deluxe Edition* (pp. 49-90). Indianapolis, IN: SAMS Publishing.

Moskowitz, D., & Kerr, D. (Eds.). (1995). *OS/2 Warp Unleashed, Deluxe Edition*. Indianapolis, IN: SAMS Publishing.

Parsons, C. (1995). File systems. In Moskowitz, D., & Kerr, D. (Eds.). *OS/2 Warp Unleashed, Deluxe Edition* (pp. 49-90). Indianapolis, IN: SAMS Publishing.

SoftTouch Systems, Inc. (1994). *GammaTech Utilities for OS/2* (Version 3.0). Oklahoma City: SoftTouch Systems, Inc.

Tyson, H. (1994). *Your OS/2 Warp Version 3 Consultant* (2nd ed.). Indianapolis, IN: SAMS Publishing.

CONTACTING THE AUTHOR

I welcome your comments at:

John M. Martz
Psychology Dept., UNC Chapel Hill
CB# 3270, Davie Hall
Chapel Hill, NC 27599
919-962-3030 or 919-962-7636
JOHN_MARTZ@UNC.EDU

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. IBM, OS/2, Workplace Shell, and Win-OS/2 are registered trademarks or trademarks of International Business Machines Corporation. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.