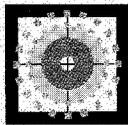


Demystifying CONFIG.SYS in OS/2 2.0 • part 2



In last month's issue, we looked at three of the six categories of command you can use in your OS/2 CONFIG.SYS. These are

Operating System Commands, Optional Devices and Memory Management. Now we'll consider commands that affect performance, application configurations and user preferences: Performance, Application Configuration and User Preferences.

Performance

These parameters affect the speed of OS/2 operations either directly, or indirectly, through the way the operating system interacts with hardware subsystems like the hard disk or printer. BUFFERS, CACHE, DISKCACHE, FILES and PRINTMON-BUFSIZE also have an effect on system memory management: the higher the value you set for each parameter, the more memory it uses. Conversely, lower values free up RAM for your applications to use. In discussing these parameters, we'll suggest the values that offer the most performance for the least memory expenditure.

BUFFERS=

This command sets the number of 512-byte disk buffers that the system can use for direct disk writes. Unlike the other statements that control disk caching, BUFFERS primarily affects direct I/O writes to FAT disk partitions. These direct writes are controlled by the IOPL command, which we covered in Part 1.

At installation, OS/2 sets this parameter to 30. You can save some memory by setting BUFFERS=10 without adversely affecting performance, according to the authors of the *OS/2 2.0 Installation and Performance Considerations* mentioned in Part 1. If you have an all-HPFS system, you can reduce this setting to 3. Don't omit the command or try to set it lower than 3, however, or OS/2 will default to the initial setting of 30.

CACHE.EXE

This OS/2 utility starts and controls the HPFS disk cache. The size of the cache is established on the IFS= command line that loads the HPFS drivers. Although not precisely a CONFIG.SYS command, CACHE.EXE can be loaded by CONFIG.SYS using the RUN= command. (RUN will be covered fully in the section on user preferences below.) As an alterna-

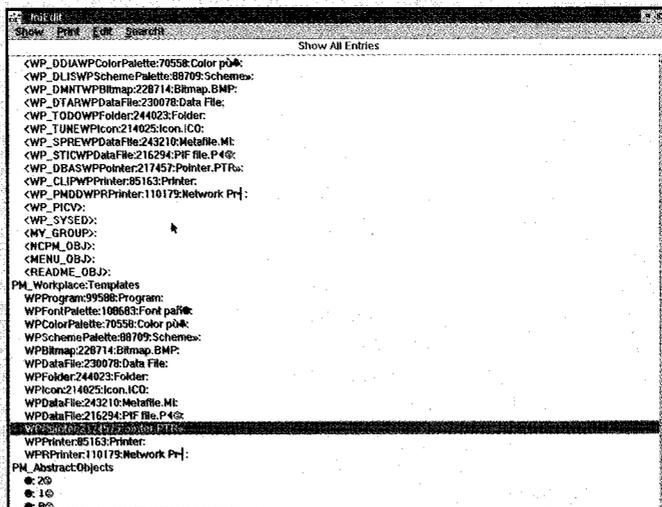


Figure 1: The remainder of the system's configuration details come from the INI files, OS2.INI and OS2SYS.INI. You can use shareware utilities, such as INIEDIT, to modify them—with care.

tive, you can also load CACHE.EXE from the STARTUP.CMD. The installation process puts it in the CONFIG.SYS file, however, and it should be left there. The line in CONFIG.SYS looks something like this:

```
RUN=C:\OS2\CACHE.EXE
```

The HPFS disk cache supports what are called lazy writes. The name derives from the fact that the operating system delays writing changed data to disk rather than writing it immediately. Lazy writes increase performance, but at the risk of losing data if the system crashes before all the changes are permanently saved. CACHE defaults to having the lazy writes option turned on. If you prefer to turn the option off, you must create a STARTUP.CMD file and put the following line in it:

```
C:\OS2\CACHE.EXE /LAZY:OFF
```

You can enter the above line at an OS/2 command prompt at any time to turn lazy writes off. If you want to suspend lazy writes only temporarily, you can turn them back on again by entering the following at a command prompt:

```
DETACH C:\OS2\CACHE.EXE  
/LAZY:ON
```

(DETACH is an OS/2 command that tells the operating system to run the application in the background. It can be used only with applications that don't require user input while running.)

It's important to note that if you have lazy writes active you must run the Shutdown option from the WPS menu before turning off your system or rebooting it. This ensures that any data left in the cache will be written to disk. Failure to shut down

the system properly can cause data loss.

Three CACHE parameters control the lazy writing process (you can put one or more of these parameters on the command line when you load CACHE.EXE using RUN, and you can change them at any time by running CACHE at a command prompt):

- **/BUFFERIDLE:** This parameter sets the number of milliseconds that the cache buffer must be idle before the cache writes update data to the disk. The default is 500ms (0.5s).

Changing this default can have an adverse effect on system performance. If you decrease the setting, the cache might start writing to the disk before the application finishes, making any necessary changes to its cached data. This would then force the cache to make a second disk write to incorporate the updated data. Indeed, a decreased BUFFERIDLE setting generally produces more frequent disk writes, which slow down the system. On the other hand, increasing the idle time could result in data loss, particularly if the operating system crashes before the disk has been updated. My recommendation is to leave BUFFERIDLE at its default setting.

- **/DISKIDLE:** This parameter specifies how long the disk itself must be idle before the cache writes out changed data. The default is 1,000ms (1 second), which IBM recommends as the best balance between system performance and the risk of data loss in a system crash.

- **/MAXAGE:** This parameter sets the amount of time (in milliseconds) that the cache waits before reorganising the cached data or writing it to disk. For read-only data, MAXAGE controls the time CACHE waits before moving data that hasn't been

accessed recently to the back of the cache and moving more frequently used data to the front.

If the data in the cache changes, **MAXAGE**: controls the maximum amount of time the system can wait (that is, laze about) before writing the data to disk. Naturally, this applies only if you have lazy writes active. If you've set **LAZY=OFF**, the **MAXAGE**: parameter has no effect on how long the cache waits to write updates to the disk.

The default **MAXAGE**: setting is 5,000ms (5 seconds), which is really too long to wait. We recommend you decrease this setting to reduce the chances of data loss in the event of a system crash, especially if you're storing critical data or running a database that's constantly being updated. Like the **BUFFERIDLE**: parameter discussed above, however, a setting of less than 500ms can adversely affect system performance. Depending on the number of updates you make to data on the disk, the best setting for preserving data integrity is between 500 and 2,000.

The best way to find the right setting for your system is to start with **MAXAGE**: set to 500 and increase the setting by 250 at a time until you find the proper number for your situation. If one of your primary applications is a disk-intensive database, for example, you'll probably end up with a setting of 1,000 or lower. You'll know you've hit the proper balance between guarding against data loss and maintaining system performance when there are no noticeable pauses in the foreground application caused by frequent writing to the disk. You can inspect the current cache settings by running **CACHE** without any parameters at any command prompt.

DISKCACHE=

This **CONFIG.SYS** command sets the size and parameters for the FAT disk cache; it has no effect on HPFS partitions. The first parameter on the **DISKCACHE** command line must be the size of the cache, in kilobytes. Following the cache size, you can add three other parameters (separated by commas) on the command line:

- **LW** Adding this parameter turns on lazy writes for the FAT drives. The default is **OFF**, although lazy writes for the HPFS cache can be turned off on the fly. To turn off lazy writes for the FAT cache you must delete the **LW** parameter from **CONFIG.SYS** and reboot the system. All the cautions about performing a proper system shutdown discussed in connection with **CACHE** also apply here.

- **T** The **T** parameter is actually a second number. It sets a threshold for the number of 512-byte sectors the cache will read

from the disk at one time. The default is 4 if this parameter is omitted. IBM recommends setting it to 1/4 the size of the disk cache. For caches of 256K and higher, this would mean a setting of 128 sectors (that is, 64K), which is the maximum allowable. The **T** setting doesn't increase the amount of memory used by the cache, but it can have a significant impact on the level of hard disk performance.

- **AC:ddd** This parameter is equivalent to the **AUTOCHECK** parameter for the HPFS device driver: it tells the system to automatically run **CHKDSK /F** on the drive(s) specified when the system boots. This is particularly handy for checking the drive on which you've installed the main OS/2 files. OS/2 keeps some of the system files open the entire time it's operating, which prevents you from running **CHKDSK /F** on that drive. Putting this command in **CONFIG.SYS** ensures that the disk is checked for errors or problems each time you start OS/2.

The following example shows a **DISKCACHE** line for a system with 8Mb of RAM and a mixture of FAT and HPFS drives. It sets the cache to 256K, turns on lazy writes, sets the threshold to 128 sectors (64K), and runs **CHKDSK /F** on the C: drive when the system boots:

```
DISKCACHE=256, LW, 128, AC:C
```

FILES=

This command is identical to its DOS counterpart. It sets the maximum number of open files for an individual DOS or Windows session and has no effect on OS/2 sessions. The minimum is 20 (which is also the default), and the maximum is 255. Increasing **FILES** to 50 provides more than enough file handles for the majority of DOS or Windows applications. Note that many DOS applications are self-limited to using only 20 open files; no amount of fiddling with OS/2 parameters can change that.

MAXWAIT=

OS/2 is a pre-emptive operating system, which means that the system has complete control over the amount of time different applications can run before giving up the CPU to another application.

MAXWAIT= sets the maximum number of seconds that a low-priority background task can stay idle. If this maximum time elapses, OS/2 increases the task's priority level to ensure that it gets a chance to run. The default is 3 seconds, which establishes a reasonable balance between smooth performance of the foreground application and continued processing of background applications. IBM says there's no reason to change it.

PRINTMONBUFSIZE=a,b,c

This command sets the size, in bytes, of OS/2's LPT1, LPT2 and LPT3 print buffers, respectively. If this command is omitted from **CONFIG.SYS**, the default (also the minimum) is 134 bytes per printer port. The maximum is 2,048 bytes. You must specify values for all three ports even if you're only changing one of them. So, if you want to increase the size of the print buffer on LPT2 to 512 bytes, you should add this line to **CONFIG.SYS**:

```
PRINTMONBUFSIZE=134, 512, 134
```

According to IBM's recommendations, you should change the buffer size to either 256 or 512, depending on the speed of your printer; generally, slower printers should have larger buffers.

PRIORITY=

This is another parameter that controls OS/2's pre-emptiveness. The default is **DYNAMIC**, which lets OS/2 change the priority of different tasks to ensure that they all share the CPU's time. If you're running an application that depends on critical timing, such as one that gathers realtime data, you should change this setting to **ABSOLUTE**, which will prevent the operating system from changing the priority level of a task to ensure it gets a chance to run. If one application has a much higher priority level than all other tasks, the **ABSOLUTE** setting gives that application, not the operating system, complete control over when to release the CPU. For all but critical timing situations, leave the **PRIORITY** setting at **DYNAMIC**.

PRIORITY_DISK_IO=

This parameter controls which applications have priority access to the hard disk. The default setting is **YES**, which gives the foreground application primary access to the disk, at the expense of background applications. If you usually run disk-intensive applications in the background (for example, database sorts or large file downloads), set this to **NO**, which gives all applications equal access to the disk.

TIMESLICE=

The final pre-emptive control parameter is **TIMESLICE=**. It controls the minimum and maximum amounts of time an execution thread can be active before it gives up the CPU to another thread. **TIMESLICE=** can take either one or two values. If two values (separated by a comma) are present, the first sets the minimum number of milliseconds a thread gets, and the second sets the maximum amount of time. The bare minimum is 32ms, and the maximum is

65,536ms. The only rule is that the maximum setting must be equal to or higher than the minimum. If only a single value is present, OS/2 uses that value for both the minimum and the maximum.

The `TIMESLICE=` parameter is usually omitted from `CONFIG.SYS` when OS/2 is first installed, so the operating system uses its default value of 32ms for both values. At the default `TIMESLICE=` settings, if you usually run more applications than you have RAM for, your system will waste a lot of processor time paging data to the disk. Studies by IBM and others have shown that the best values under such circumstances are 64 and 128. These increase the time a task runs before the operating system pages the memory contents to disk, and can be implemented by adding the following line to your `CONFIG.SYS` file:

```
TIMESLICE=64,128
```

Application configuration

Like DOS, OS/2 can use environment variables to configure OS/2 applications. These variables also use the

```
SET varname=command
```

format to put the value of the variable into the environment space. Note that `SET` commands in the OS/2 `CONFIG.SYS` apply only to the operating system and OS/2 applications. The `PATH` command for DOS and for Windows applications running in OS/2 is established in the OS/2 version of `AUTOEXEC.BAT`. The five environment variables OS/2 uses to control the overall operation of system utilities and OS/2 applications are `BOOKSHELF`, `DPATH`, `GLOSSARY`, `HELP` and `PATH`.

- **SET BOOKSHELF=** The `BOOKSHELF` variable tells OS/2 where to find the INF files that contain the on-line reference manuals. OS/2 ships with two manuals: a command reference and a reference for the REXX extended batch programming language. The default setting puts the `\OS2\BOOK` subdirectory on the system boot drive. You can free up disk space by moving these files to another partition and changing `SET BOOKSHELF=` to point to the new location. If you have an HPFS partition, for example, you should move the reference manual files there to speed up searches.

- **SET DPATH=** `DPATH` stands for data path; OS/2 applications can use this environment variable to find data and configuration files that aren't in the current directory.

When you first install OS/2, `DPATH` is set to the directories in which the operating system data files are stored. End-user applications, when installed, will usually

add their own data paths to the end of the `DPATH`. If not, the application manual may tell you to edit this command and add the data path yourself. Either way, make sure that the new information goes after the default data paths that OS/2 sets. That will ensure that the operating system can find its own data files quickly.

- **SET GLOSSARY=** The `GLOSSARY` variable tells OS/2 where to find the data files for the glossary portion of the on-line documentation. The default setting is the `\OS2\HELP\GLOSS` subdirectory on the system boot drive but, as before, you can free up disk space by moving these files to another partition. You can also move the glossary files to an HPFS partition to speed up searches.

- **SET HELP=** This variable tells OS/2 where to find its on-line help files. OS/2 applications can also use the directory to which this variable points to store their own help files. The default location is the `\OS2\HELP` subdirectory on the system boot drive; again, you can move these files to another drive or to an HPFS partition to speed up searches.

- **SET PATH=** This command is identical to the `PATH` command in DOS: it tells the operating system where to find executable programs and batch files that aren't in the current directory. Note, however, that the `CONFIG.SYS PATH` command applies only to OS/2 sessions. You must set the `PATH` for DOS and Windows applications in the OS/2 version of `AUTOEXEC.BAT`.

User preferences

The remaining `CONFIG.SYS` commands and parameters are used to configure the operating system to your liking. Some are established when you install OS/2 and usually don't have to be changed. You can add or change the rest to customise the system to your working style.

AUTOFAIL=

This command tells OS/2 what action to take when an error occurs; for example, when a disk is not ready or a file isn't found. The default action parameter `NO` pops up a dialog box with an error message and presents you with a choice of actions to take. Changing `AUTOFAIL` to `YES` causes OS/2 to simply display an error number and return the error code to the application that caused the error, precluding user intervention. You should leave this set to `NO` so you get complete information on system problems.

BREAK=

This command affects only DOS and Windows sessions. It determines how often an application checks for the Ctrl-C key com-

bination. The default value is `OFF`, which tells the applications to check for Ctrl-C only during standard input or output, printer output or auxiliary output. This setting provides the best performance. Setting `BREAK` to `ON` lets you break the operation of the program at any time. This forces the program to check the keyboard buffer for the Ctrl-C keystrokes constantly. Since this will have a negative effect on the performance of DOS and Windows applications, you should leave the setting at `OFF`.

CODEPAGE=

Code pages are IBM's name for the customised character sets that establish the keyboard and display characters for various native languages. Code pages are identified by number. In the UK, `CODEPAGE` defaults to the 437 and 850 code pages; 437 is English, and 850 is a default multilingual character set. The `CONFIG.SYS` line looks like this:

```
CODEPAGE=437,850
```

The first number is the initial code page used. You can switch between code pages by using the `CHCP page#` command at any system prompt. Other code pages shipped with the operating system support Japanese, Canadian French, Hebrew, Chinese and other languages. Although you could change the `CODEPAGE=` value manually, since other settings depend on it, it's better to use the Country and Selective Install programs which are found in the System Setup folder.

COUNTRY=

This command sets the default values for country-specific parameters like the common formats for date, time and currency. It's somewhat like the International Dialling Access code used to make long-distance calls. The command takes two parameters. The first is a three-digit number that tells OS/2 which country's defaults to use. For the UK, the first parameter is 011; for the US, it's 001. The second parameter is the location of the file that contains the various settings. This defaults to `\OS2\SYSTEM\COUNTRY.SYS` on the system boot drive. Unless you have a custom `COUNTRY` file, there's no need to change this setting. In the UK, the line is:

```
COUNTRY=011,C:\OS2\SYSTEM\COUNTRY.SYS
```

Again, it's best to make any changes to this line with the Country setup program in the System folder.

DEVINFO=

`DEVINFO` stands for device information. The command works with `CODEPAGE` to

set up the character sets for different system devices: the screen, the keyboard and the printers. Each device has its own DEV-INFO= line; the first parameter on the command line determines which device will be affected. Screen controls are set by SCR, the keyboard by KBD, and the printers by LPT#, where # is the printer port number 1, 2 or 3.

DEVINFO= statements for the screen and keyboard must be present; printer controls are optional and are usually installed by printer drivers that require them. The second parameter establishes the default setting for a device, and the third is the name of the file that contains the different settings. In the UK, these two lines are placed in CONFIG.SYS when OS/2 is installed on a VGA system:

```
DEVINFO=SCR, VGA, C:\OS2\VIO
TBL.DCP
DEVINFO=KBD, US, C:\OS2\KEY
BOARD.DCP
```

You can use the Selective Install utility in the System Setup folder to change these values as necessary.

SET varname=

A number of environment variables are used to customise the system for user preferences. SET VIDEO_DEVICES= and SET VIO_VGA= tell OS/2 what type of screen display you have. These are set during the installation process. If you change video cards, you can change these settings with the Selective Install utility in the System Setup folder. Otherwise, leave them at their initial values.

There are three SET commands that customise other system functions and that you might want to add to your CONFIG.SYS:

- **SET DELDIR=** This turns on OS/2's undelete protection so you can recover any accidentally deleted files. The command works hand in hand with the included UNDELETE.COM utility. The two parameters for this command are the directory used to store the deleted files, followed by the maximum amount, in kilobytes, to be stored. The undelete directory and the files in it have the 'hidden' attribute set, so you usually won't see them when you do a DIR.

As an example, if you want to turn on deletion protection for your C: and D: drives, and to save a maximum of 512K of deleted files on each, you would add the following line to your CONFIG.SYS:

```
SET DELDIR=C:\DELETE, 512;
D:\DELETE, 512
```

- **SET KEYS=** If you enter ON as its parameter, this command turns on the command history buffer in OS/2 character-

Reference G325-5019

IBM Technical Publication Centre
(0256) 343000

CONFIG.SYS

Complete Listing

```
IFS=C:\OS2\HPFS.IFS /CACHE:256 /CRECL:4 /AUTOCHECK:F
PROTSHELL=C:\OS2\PMSHELL.EXE
SET USER_INI=C:\OS2\OS2.INI
SET SYSTEM_INI=C:\OS2\OS2SYS.INI
SET OS2_SHELL=C:\OS2\CMD.EXE
SET AUTOSTART=PROGRAMS, TASKLIST, FOLDERS
SET RUNWORKPLACE=C:\OS2\PMSHELL.EXE
SET COMSPEC=C:\OS2\CMD.EXE
LIBPATH=.;C:\OS2\DLL;C:\OS2\MDOS;C:\;C:\OS2\APPS\DLL;

SETPATH=C:\OS2;C:\OS2\SYSTEM;C:\OS2\MDOS\WINOS2;C:\OS2\
INSTALL;C:\;C:\OS2\MDOS;C:\OS2\APPS;F:\OS2\UTIL;
SET
DPATH=C:\OS2;C:\OS2\SYSTEM;C:\OS2\MDOS\WINOS2;C:\OS2\INSTALL
;C:\;C:\OS2\BITMAP;C:\OS2\MDOS;C:\OS2\APPS;
SET PROMPT=$I[$P]
SET HELP=F:\OS2\HELP;F:\OS2\HELP\TUTORIAL;
SET GLOSSARY=F:\OS2\HELP\GLOSS;
PRIORITY DISK IO=YES
PRIORITY=DYNAMIC
FILES=50
DEVICE=C:\OS2\DOS.SYS
DEVICE=C:\OS2\PMDD.SYS
BUFFERS=10
DISKCACHE=256, LW, 128, AC:C
RUN=C:\OS2\CACHE.EXE /MAXAGE:2000
MAXWAIT=3
TIMESLICE=64, 128
MEMMAN=SWAP, MOVE, PROTECT
SWAPPATH=F:\OS2\TEMP 2048 4096
BREAK=OFF
THREADS=256
PRINTMONBUFSIZE=256, 134, 134
COUNTRY=001, C:\OS2\SYSTEM\COUNTRY.SYS
SET KEYS=ON
SET
DELDIR=C:\DELETE, 512;D:\DELETE, 512;E:\DELETE, 512;F:\DELETE
, 512; BASEDEV=PRINT01.SYS
BASEDEV=IBM1FLPY.ADD
BASEDEV=IBM1S506.ADD
BASEDEV=OS2DASD.DMD
SET BOOKSHELF=F:\OS2\BOOK;F:\OS2\HPFSUTIL;
SET EPATH=C:\OS2\APPS
PROTECTONLY=NO
SHELL=C:\OS2\MDOS\COMMAND.COM C:\OS2\MDOS /P /E:512
FCBS=16, 8
RMSIZE=640
DEVICE=C:\OS2\MDOS\VEMM.SYS
DEVICE=C:\OS2\MDOS\VMOUSE.SYS
DOS=HIGH, UMB
DEVICE=C:\OS2\MDOS\VDPX.SYS
DEVICE=C:\OS2\MDOS\VXMS.SYS /UMB
DEVICE=C:\OS2\MDOS\VDPMI.SYS
```

(continued)

OS/2 2.0 installation and performance considerations

```

DEVICE=C:\OS2\MDOS\VWIN.SYS
DEVICE=C:\OS2\MDOS\VCDROM.SYS

DEVINFO=SCR,VGA,C:\OS2\VIOTBL.DCP
SET VIDEO DEVICES=VIO_VGA
SET VIO_VGA=DEVICE(BVHVGA)
DEVICE=C:\OS2\MDOS\VVGA.SYS
DEVICE=C:\OS2\POINTDD.SYS
DEVICE=C:\OS2\MOUSE.SYS SERIAL=COM2
DEVICE=C:\OS2\COM.SYS
DEVICE=C:\OS2\MDOS\VCOM.SYS
CODEPAGE=437,850
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
IOPL=YES

```

Available as: INSPER.ZIP in Productivity Library (#3), in PCMAGUK forum. To access GO ZNT.PCMAGUK



mode sessions, permitting you to scroll back and forth between previously entered commands. SET KEYS=OFF, the default, turns the command history buffer off.

• **SET RESTARTOBJECTS=** This is an undocumented environment variable that controls the startup action OS/2 takes with respect to applications that were active when the system was last shut down. Given its usual ON default, OS/2 will automatically restart any applications that were open when the system stopped, returning you to where you left off. The command:

```
SET RESTARTOBJECTS=OFF
```

prevents OS/2 from starting anything but the WorkPlace Shell.

You can also use:

```
SET RESTARTOBJECTS=
STARTUPFOLDERONLY
```

which tells OS/2 to start only the applications placed in the Startup folder on the desktop. Note that because this command is undocumented, and may not work in future versions of OS/2.

RUN=

The RUN= command lets you automatically load and run character-mode OS/2 applications from the CONFIG.SYS file. You can have more than one RUN= command in CONFIG.SYS. These commands are the last lines processed during system startup.

The RUN= command is most commonly used to start the HPFS CACHE. Some LAN drivers can also be started using RUN= instead of being loaded in STARTUP.CMD; the manuals will tell you if this is the case. RUN= is also used to start up special services, such as LAN file servers or database servers. The text after RUN= should comprise the full pathname

of the application and any parameters needed; see the discussion of CACHE above for an example.

VERIFY=

This command tells OS/2 to verify that a file was written to disk correctly by immediately comparing it with the original. This is equivalent to using the /V option on the COPY and XCOPY command lines. The default is OFF. If you store a lot of critical data, or if your system is being used as a concurrent LAN or database server, set VERIFY=ON.

Bringing it all together

Listing 1 consists of a sample OS/2 2.0 CONFIG.SYS file that has been tuned for best performance on an 8Mb 33MHz 486 system. The system has a VGA monitor and a mixture of FAT and HPFS partitions on a 420Mb hard disk. Drives C:, D: and E: are FAT, and F: is HPFS. The system boots from C:. As you examine the listing, note the following points as a guide to tuning your own OS/2 system:

- The first step is to move the location of the swap file from the boot drive to one that has more room. As you can see in the example, SWAPPATH is set to a subdirectory on the HPFS drive F:.
- UMBs are activated for DOS and Windows sessions by the addition of the /UMB parameter on the

```
DEVICE=C:\OS2\MDOS\VXMS.SYS
```

line. DOS=HIGH,UMB lets the built-in DOS interpreter load itself into the UMBs, and permits loading device drivers and TSRs into the UMB with DEVICEHIGH and LOADHIGH commands.

- The HPFS and FAT disk caches are sized according to the recommenda-

tions in Part 1. The HPFS cache is further tuned by reducing the amount of time it waits to write data to the disk by using the /MAXAGE:2000 parameter on the line:

```
RUN=C:\OS2\CACHE.EXE
```

BUFFERS have been reduced to 10 in accordance with IBM's recommendation. Lazy writes are turned on for the FAT drives through the addition of LW after the DISKCACHE command. This speeds up the file system performance.

- The PRINTMONBUFSIZE= command enhances printer performance on LPT1 by increasing the buffer to 256 bytes. This setting is fine for a laser printer; for a dot-matrix printer, 512 would be better.
- All on-line help files and reference manuals have been moved to the HPFS drive both to free disk space on the boot drive and to speed up searches.
- Finally, OS/2 is given full control over memory management and multitasking by using the MEMMAN=, the PRIORITY_DISK_IO= and the PRIORITY= commands. The TIMESLICE= values are raised to 64 and 128 (which appears as 64,128 on the CONFIG.SYS line), to provide smoother operation of foreground and background applications when the system has to swap committed memory to disk.

Get the balance right

The correct settings for your particular needs might vary from those shown in Listing 1. You must be willing to experiment a little, to try different values for the various settings until you strike your own balance between performance and memory usage.

The best way to check memory usage is by monitoring the size of your SWAPPER.DAT file. If SWAPPER.DAT consistently gets much larger than its initial value after you've started up only one or two applications, you should lower some of the values that affect the amount of memory used by the system.

The guidelines we present can serve as the starting point for your own exploration. The goal is to make the system's operations as smooth as possible, so it spends more time processing applications and less time swapping data to the disk. ■

OS/2 2.0 Installation and Performance Considerations

Reference G325-5019

IBM Technical Publications Centre

(0256) 343000