

## DOS's INTERNAL AND EXTERNAL COMMANDS

Windows, Windows, Windows. That's all we hear today. However, there are still plenty of us that still use good old DOS for a significant part of our day-to-day computing activity. For example, even though I am writing this using MS-Word in Windows 95, when I finish and back up the article to a floppy, I will exit Windows and use DOS to do it. For me, it is simpler, faster and safer to do it that way. And to repeat the statement I have made in previous articles, when you exit Windows 95 you are in DOS 7. Version 7 is the upgrade from DOS 6.22 on which Windows 95 depends. It is plain, old, vanilla DOS, though it has been improved over the previous version.

I thought it might be useful, for those who use DOS occasionally or more often, to review internal and external commands. Knowing the difference can give you insight into why, for example, you get the message "Bad command or filename" when you type a command that you thought was correct. This article will tell you why, and give you the background.

The file COMMAND.COM is a DOS program supervises all interaction with you, the human user. It watches the keyboard to see if you typed anything, and if you do, it is the part of the operating system that tries to interpret and carry out your commands. But that is not all it does.

In the early days of computing, a very wise programmer who was designing COMMAND.COM decided that it would be a good thing to put some of the most important little programs like DIR and DEL and COPY directly inside COMMAND.COM, and that is where they are today. If you do a search, you won't find any little program lying around called DIR.COM or DEL.COM or COPY.COM, because the program code for these functions is contained right inside COMMAND.COM. Actually, there are quite a number of these little programs inside it. Here is a complete list with a very abbreviated description of what each does:

BREAK	Tells DOS how often to watch for Ctrl-C.
CALL	Calls one batch file from another.
CD	Change directory.
CHCP	Change or report on code page, for foreign characters.
CLS	Clear the screen.
COPY	Copy a file.
DATE	Change or display the date.
DEL	Delete a file.
DIR	Show a file directory.
ECHO	Controls the information DOS shows on the screen.
ERASE	Same as DEL.
EXIT	Close the COMMAND.COM session.
FOR	Looping command for batch files.
GOTO	Controls moves within batch files.
IF	Decision making step in batch files.
LOADHIGH	Load a program into upper memory.
MD	Make a subdirectory.
PATH	Change or display COMMAND.COM's search path.
PAUSE	Display the message: "Press any key to continue".
PROMPT	Change the command prompt.
RD	Remove a subdirectory.
REM	Labels a line in a batch command as a remark.
REN	Rename a file or group of files.
SET	Display or create environmental variables.
SHIFT	Moves the position of parameters in a user command.
TIME	Change or display the system time.

TRUENAME	Display a complete filespec for a file.
TYPE	Copy a file to the screen.
VER	Display the DOS version number.
VOL	Display the disk volume ID.

That's a lot of stuff! And that is why COMMAND.COM is a relatively big program, as DOS programs go (54,645 bytes for MS-DOS 6.22).

That means that if you had a computer with only the operating system on the hard drive (just the two hidden files IO.SYS, MSDOS.SYS and COMMAND.COM), if you typed DIR at the command prompt, COMMAND.COM would display a directory on your screen. That's because the programming code that displays the directory is located right in COMMAND.COM itself. DIR is called an **internal command** for that reason.

All other DOS commands use separate programs, such as the commands CHKDSK, DISKCOPY, FORMAT, MEM and SYS, and these are called **external commands** since they are not inside COMMAND.COM. If you look in the DOS directory on your hard drive, you will find the programs CHKDSK.EXE, DISKCOPY.COM, FORMAT.COM, MEM.EXE and SYS.COM, among the others found there. In the hypothetical computer mentioned previously with only the three operating system files on it, if you typed MEM, COMMAND.COM would look inside itself first, and then on the hard drive, for a file named MEM with the extension EXE, COM or BAT. Having found no such program, COMMAND.COM would send you the old familiar message "Bad command or filename".

That is why the PATH command is so important. The PATH command tells COMMAND.COM where to look for files, after it looks inside its own "belly" and does not find the command there. Without the PATH statement in your AUTOEXEC.BAT to tell COMMAND.COM where to search, COMMAND.COM will not look in the appropriate subdirectories to find the program you want to run. PATH is a road map for COMMAND.COM.

Had the programmers who devised DOS been more creative and sensitive to the way that humans think, the "Bad command or filename" message might have looked like this: "I can't find the program you want to run; perhaps you typed its name incorrectly". That would have made a lot more sense to most of us. Happy computing!