

Running DOS applications in Windows XP

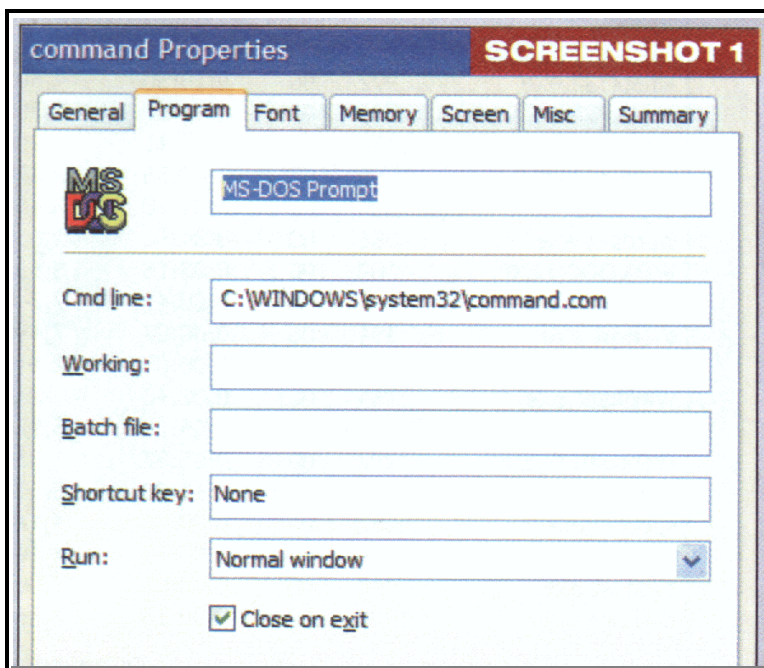
When Microsoft released Windows XP in October 2001 it was its first consumer operating system that wasn't based in any way on legacy Dos code. Despite the absence of Dos, XP is still capable of running most old DOS applications perfectly well, but it isn't 100 per cent. If you are having problems running DOS applications (such as Turbo C++ V1), then the information below may help.

Check your properties

Before attempting the steps below, try running the program by simply clicking on it then try various settings on the compatibility tab, perhaps to trick it in to thinking it is running under Windows 95.

If a DOS Program doesn't run properly, the odds are its program window will open and close so quickly you can't even see its contents. Since you can't troubleshoot a program without seeing the error it's giving you, you need to access the MS DOS shortcut for it so you can modify its run parameters.

To do so, right click the program's executable choose Properties and disable the Close on exit checkbox found on the Programs tab (see screenshot 1). Once you've set this option try opening your DOS program's executable again. This time the DOS window should remain open after the program bombs out, allowing you to see the error. Sometimes this error will be something that Windows XP can't get around, but other problems are fixable. If it's a memory problem, usually having to do with not enough XMS or EMS memory, there's a simple fix.



Uncheck the Close on exit tick box to read error message

Since many DOS programs use or required what was then a lot of memory, and DOS could only offer 640Kb of Conventional memory, you had to use commands such as himen.sys or emm386.exe to give your programs use of Extended and Expanded memory. If a program is going to have memory problems, you'll typically get some form of no Expanded / Extended memory error when you try to run it.

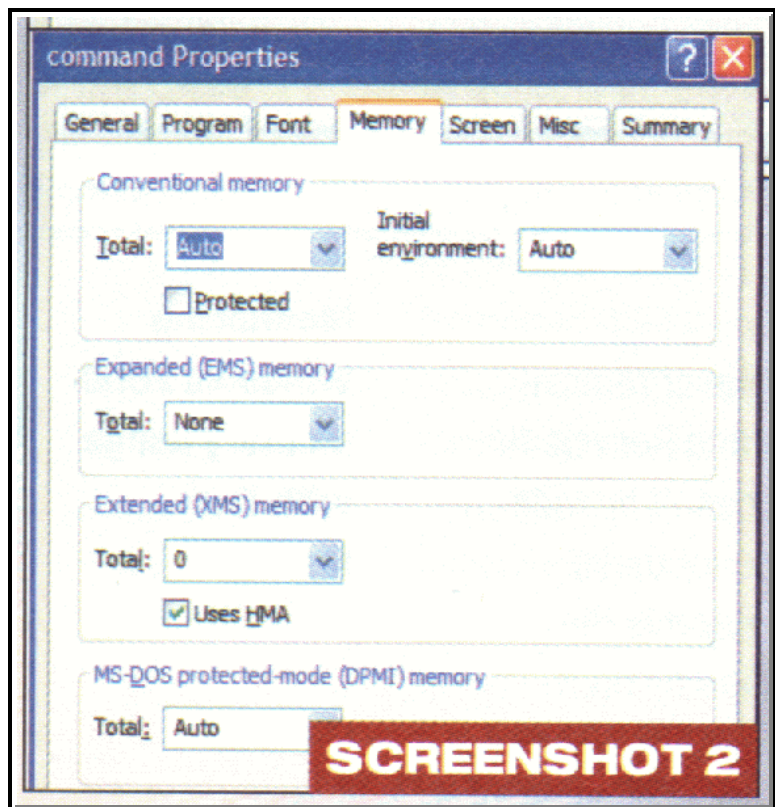
To try to fix this problem, right click on the program and select Properties. This time, choose the memory tab. On this tab are sections for configuring Conventional, Expanded, Extended and Protected memory. By default,

you should find that the dropdown list in the Total field of the Conventional memory area is set to 'Auto'. You'll also find that the dropdown lists for Expanded and Extended memory are set to None and 0, respectively (see screenshot 2). Choose 'Auto' for both of these options and click OK to close out of the Properties window.

The initial environment should also be set to its highest setting of 4096. Under the Total dropdown box is a box labelled 'Protected'. This protects the system memory from modifications made by the selected program and should be checked.

Configuration files

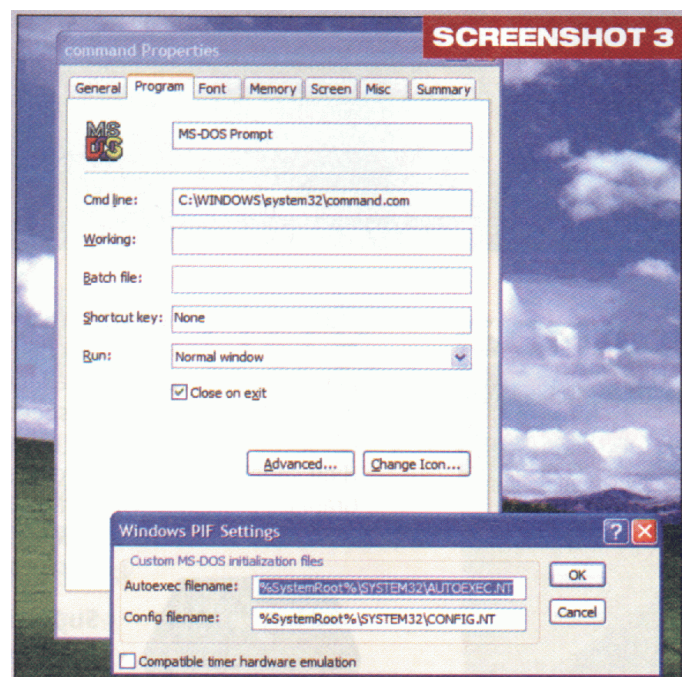
If this doesn't solve your memory problems, there's still one trick up XP's sleeve. Those who remember the autoexec.bat and config.sys files will remember the constant tweaking of command lines, attribute setting and so on in order to squeeze the last ounce of performance from our machines. Well, these old files haven't completely gone away, they've just assumed new identities.



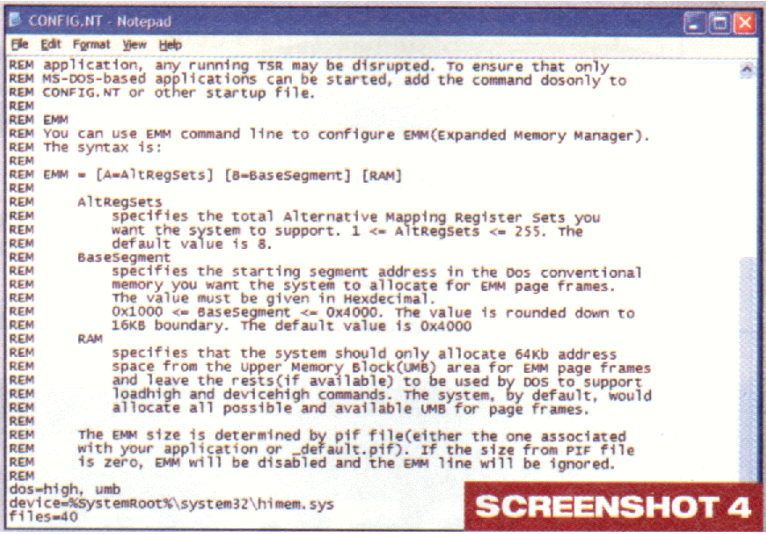
By default, support for Expanded and Extended memory is switched off.

If you open My Computer and go to your Windows/System32 folder, you'll find them disguised as config.nt and autoexec.nt and, guess what, you can tweak them just like you could tweak their predecessors and each program can also have its own private copies of these files. As a precaution, copy these files into your DOS program directory before making changes to them. It's important to remember that the autoexec.bat and config.sys files in your root directory have nothing to do with DOS programs.

To make a DOS program uses a new set of config and autoexec files, right-click its executable file and choose Properties. This time, return to the Program tab and click the Advanced button. This opens a dialogue box labelled Windows PIF Settings, which contains two fields: one for an autoexec filename and one for config. By default these two fields point to your %SystemRoot%\System32 directory but if you put copies of your config and autoexec files in a DOS program's directory called for example "TC\BIN" then you'd change them to:
C:\TC\BIN\AUTOEXEC.NT and
C:\TC\BIN\CONFIG.NT in these two fields (see screenshot 3). Enabling the Compatible Timer Hardware Emulation checkbox that appears beneath these two fields will have the effect of slowing down the processing, this might be used with DOS Games that run too fast.



To edit the contents of config.nt or autoexec.nt use something like Notepad - both files are well commented (see screenshot 4). Unfortunately, the changes you can make here are restricted, confined mainly to things such as the EMM=RAM command. Some games require this option to be set before they'll recognise your Expanded Memory.



```
CONFIG.NT - Notepad
File Edit Format View Help
REM application, any running TSR may be disrupted. To ensure that only
REM MS-DOS-based applications can be started, add the command dosonly to
REM CONFIG.NT or other startup file.
REM
REM EMM
REM You can use EMM command line to configure EMM(Expanded Memory Manager).
REM The syntax is:
REM
REM EMM = [A=AltRegSets] [B=BaseSegment] [RAM]
REM
REM AltRegSets
REM specifies the total Alternative Mapping Register Sets you
REM want the system to support. 1 <= AltRegSets <= 255. The
REM default value is 8.
REM BaseSegment
REM specifies the starting segment address in the Dos conventional
REM memory you want the system to allocate for EMM page frames.
REM The value must be given in Hexdecimal.
REM 0x1000 <= BaseSegment <= 0x4000. The value is rounded down to
REM 16KB boundary. The default value is 0x4000
REM RAM
REM specifies that the system should only allocate 64kb address
REM space from the Upper Memory Block(UMB) area for EMM page frames
REM and leave the rests(if available) to be used by DOS to support
REM loadhigh and devicehigh commands. The system, by default, would
REM allocate all possible and available UMB for page frames.
REM
REM The EMM size is determined by pif file(either the one associated
REM with your application or _default.pif). If the size from PIF file
REM is zero, EMM will be disabled and the EMM line will be ignored.
REM
REM dos=high, umb
REM device=%SystemRoot%\system32\himem.sys
REM files=40
```

SCREENSHOT 4

[The suggestions made in this document are adapted from an article Personal Computer Weekly (PCW) magazine by Roger Gann]