NEW!

Console yourself

A GUI is great for lots of jobs, but Dale Strickland-Clark suggests that a console window might make you more productive.

W elcome to the first Hands On Windows NT column. Over the coming months I'm going to tease and provoke this operating system into giving up some of its secrets. Armed with a copy of the monstrously-proportioned Windows NT Resource Kit, a stack of MSDN CDs, a web browser and a snorkel, I'm going to wade into the murky depths of NT and pull out anything that vaguely resembles buried treasure.

As a keen advocate of getting the most done with the least effort, I abandoned Program Manager long ago as it was too much hard work, switching to a console

window and a bunch of DOSKEY macros to launch most applications. A console window, sized to 13 lines by 50 columns, is enough to enter most commands, but I keep a second, larger window minimised for longer-running commands and anything that needs more room.

These, along with a short macro called "/" to launch commands in a new window (see the DOSKEY Macros panel) have served me pretty well up to NT4, and I haven't seen much need to change since. In fact, the extended command syntax introduced in NT4 has led me to try and put more and more complex commands on

DOSKEY macros.

Firstly, let's just clear up one common misunderstanding from which I suffered for too long. The console window is not running a DOS environment. Yes, I know Microsoft chose an icon for the command prompt that screams MSDOS at you. However, that was just to make people feel comfortable that good old, understandable MSDOS was just a double-click away. Frankly, I would have thought that this was more likely to induce panic, but what do I know?

It's only when you attempt to run a DOS program that a virtual machine is created to simulate the old DOS environment. It's at this point that the system runs the AUTOEXEC.NT batch file found lurking in the \SYSTEM32 sub-directory, off the system root. You can add commands to this file to further prepare the virtual machine for your DOS applications, but unless one of these commands issues messages, you probably won't even know that it's running.

Curiously, the MEM command is one of the few real DOS commands shipped with NT. If you type "MEM /C" at the command prompt, NT will create a virtual machine, run AUTOEXEC.NT which loads a few device drivers, and then run the MEM command. This lists the device drivers just loaded and not much else, and then terminates. The virtual machine is then removed, along with all the device drivers just loaded. What's the point of that?
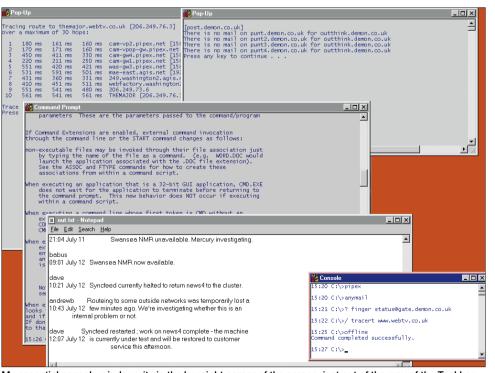
We'll have a look at the DOS

My essential console window sits in the low right corner of the screen, just out of the way of the Taskbar, and never moves. The other windows pop up and vanish on demand. If I'm expecting a lot of output from a command, I just stick a "/" or a "?" on the front of it and the output appears in another window

DOSKEY Macros

Here's a selection from my much larger list of DOSKEY macros. Most of these work on either NT 3.51 or 4.0 but there are a few that use NT 4.0 features. Some of these macros simply launch applications, such as "word". They use the "START" command to prevent the console window from waiting until the application finishes before returning to the prompt. With NT4 this isn't necessary, because the "START" command is implied when launching applications.

```
/=start "Pop-Up" cmd /c cmdinit.bat ^& $* ^& pause
?=start /min cmd /c cmdinit.bat ^& $* ^>c:\temp\out.txt ^& start
      notepad c:\temp\out.txt\=start $*
+=start "Pop-Up" cmd /c cmdinit.bat ^& $* ^| more /e ^& pause
acc2=start C:\KITS\MSOFF42\ACCESS\MSACCESS.EXE $*
acc7=start C:\kits\MSOffice\Access\MSACCESS.EXE $*
agent=start /ds:\internet\agent i:\comms\agent\agent.exe $1.ini
ameol=start /max /ds:\ameol I:\COMMS\AMEOL32\AMEOL32.EXE $*
anymail=start "Pop-Up" cmd /c finger outthink@post.demon.co.uk ^&
      pause
BulkRename=for %i in ($1) do rename %i $2%i
cixip=start "Msgs" cmd /c rasdial "cix ip" ^&^& start rasmon
convertmacs=for /f "tokens=1,* delims==" %i in ('doskey /macros ^|
      findstr "[&|]"') do @echo >batch\conv\%i.bat @%j
cw=start "Console" cmd /k cmdinit.bat
editmacs=Start notepad c:\batch\macros.txt
ex=explorer $*
exr=explorer /e,/root,$*
ext=explorer /e,$*
history=doskey >c:\temp\history.txt /history && start notepad
      c:\temp\history.txt
job=explorer "s:\jobs\$*" && if exist "s:\jobs\$*\tools" explorer
      "s:\jobs\$*\tools" & if exist "s:\jobs\$*\startup" for %i in
      ("s:\jobs\$*\startup\*.*") do %i
loadmacs=doskey /macrofile=c:\batch\macros.txt
offline=rasdial /disconnect && sleep 1 && kill >nul rasmon
qmac=doskey /macros | find "$*"
qmacs=doskey /macros
resume=dir s:\ameol\data\resume\$1.rsm | find /i "$*" & start notepad
      s:\ameol\data\resume\$1.rsm
savemacs=doskey /macros | sort >c:\batch\macros.txt
space=for /f "tokens=1-3" %i in ('dir /d c:\ d:\ \\server1\drive-c
\\server1\drive-d \\server1\drive-e ^| find "bytes free"') do @echo %i
      %j %j
word=start /max C:\kits\MSOffice\Winword\WINWORD.EXE $*
```

environment in a future issue, but meanwhile, back to the console window. The need to keep the mini console window free for entering more commands means that many of the macros use a START command to open a new window. This introduces its own little complication, because the new window vanishes without so much as a puff of smoke as soon as the command has finished. This is thanks to a remarkable bit of inconsistency where DOS windows in Windows 3.1 and Windows 95 have the option of remaining open once the command has terminated. But Windows NT

offers no such sophistication. To overcome this shortcoming, I've got into the habit of adding a PAUSE to the end of most commands that run in a separate window.

Although NT's console language is virtually indistinguishable from that of DOS, it does have a number of useful additions and the list has grown rapidly with NT4. Ignoring the new NT4 stuff for now, the principal extensions are "&" (an ampersand) to separate two commands on the same line, "&&" to separate two commands and execute the second only if the first was successful, and "||" (two vertical bars) to

Windows NT Resource Kit (3.51)

I've been fortunate enough to get my hands on one of these magnificent works which comprise five stout volumes and a CD-ROM brimming with goodies. The books cover networking, optimisation, messages, the Registry and a whole lot more, and occupy over six inches of shelf-space between them.

■ Volume 1. Deals with the NT architecture and is essential reading for anyone who wants to achieve a sound understanding of what makes NT tick. It also includes a description of each file shipped with NT and an explanation of many of the values stored in the Registry. The knowledge junkie will be able to open this book on just about any page and be instantly absorbed.

■ Volume 2. Crammed with networking information. Identifies the purpose of the networking options that most mortals dare not touch. Its detailed coverage of the RAS SWITCH.INF file to automate logon processing should make the essential connection to your ISP a quicker operation.
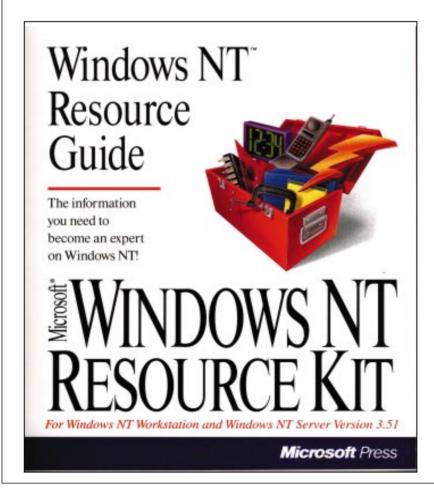
■ Volume 3. A book full of system messages is of dubious value when the accompanying explanation offers little more advice than the original message, but Volume 3 is full of them anyway. There is some help here, yet this volume will still look new long after the other four are ragged from use.

■ Volume 4. If you want to strangle every ounce of throughput from your gasping system, you'll want to be propped up in front of this into the wee small hours. It covers optimisation, and shows how to use the System Monitor to measure performance and identify bottlenecks.

■ Volume 5. A supplement containing additions to the first four volumes.

One or two items from the CD-ROM can be found on the internet but it is, nevertheless, a valuable collection of utilities and general goodies that no serious NT shop should be without. I will be looking in detail at some of this stuff in future months.

See PCW Contacts *for details of Resource Kit version 3.51.*

separate two commands with the second, only executing if the first failed.

This is clear enough until you begin stringing long lists of commands together. Consider the following:

```
cd nosuchdir && type missing.txt ||
echo Can't find directory
```

If the CD command is successful, the file is typed, otherwise the ECHO prints a message. This works as required but perhaps not as expected. Given that the ECHO follows the TYPE command, shouldn't the message be "Can't find file"? In fact, the ECHO executes if either the CD or the TYPE command fails.

It turns out that the real behaviour of "&&" and "||" are linked to the current errorlevel (usually set to non-zero by failing commands), causing the next command to be executed or skipped depending on its value. However, commands that are skipped leave the errorlevel unchanged.

With a bit of imagination, it's possible to take advantage of this in some interesting ways. Here's an example where a string of commands employ a common error handler:

```
cmd1 && cmd2 && cmd3 && cmd4 ||
echo oh goodness me
```

If any command fails, the remaining are skipped and the ECHO is executed.

More specific error-handling demands the use of a further addition to the original DOS syntax; namely, the use of parentheses to group commands. This example provides specific error messages:

```
cd nosuchdir && (type missing.txt
|| echo Can't file file) || echo
Can't find directory
```

It appears that the errorlevel is saved on entering a command group and restored again on exit.

With this groundwork in place, it is possible to string some quite complex command sequences together on a single line or in a single DOSKEY macro. The list of macros shown ranges from simple shorthand to some pretty complex loops that depend on features introduced in NT4, with which I will deal next month.

●PCW Contacts

Dale Strickland-Clark is a consultant on Windows/NT and the internet. Email **dale@outthink.demon.co.uk**

Computer Manuals 0121 706 6000; http://www.compman.co.uk for Microsoft Windows NT Resource Kit version 3.51 (ISBN: 1556159269). £187.99 (incl. VAT)