

NTegration

by Eileen Frisch



Making yourself Feel at Home

In the previous two columns, we've taken some initial steps into the Windows NT world. This month, we'll look at some freely available and commercial software packages that can make this new place look and feel more like what we're used to: the familiar UNIX environment.

NT Resource Kit's POSIX Utilities

The Windows NT Resource Kit is an add-on product to the basic operating system sold by Microsoft. There are separate kits for the workstation and server versions of the operating system. The Resource Kit has two components: supplementary printed documentation and additional software, including many important administrative programs. Although Microsoft charges money for the Resource Kit, the software contained in it is unsupported.

You should consider the Resource Kit a required part of any Windows NT installation. It's unfortunate that

there is additional cost associated with it because its contents really ought to be part of the normal Windows NT product. You can purchase the Resource Kit at most large bookstores (in the computer section), at many retail software stores and from mail order hardware and software suppliers. The street price for the workstation and server versions is about \$55 and \$150, respectively.

One of the software items is a set of command-line utilities known as the POSIX Utilities. When installed, the executable files for these commands are located in the POSIX subdirectory of the Resource Kit installation directory (usually `C:\NTResKit`). The following utilities are included: `cat`, `chmod`, `chown`, `cp`, `find`, `grep`, `ln` (for hard links only), `ls`, `mkdir`, `mv`, `rm`, `rmdir`, `sh`, `touch`, `vi` and `wc`. Most of them behave as expected, but `find` is somewhat eccentric. When using them, be aware that they treat filenames as case-sensitive and that filenames on FAT (Windows-format) file systems are converted to uppercase.

GNU Utilities for Win32

The GNU utilities collection has been ported to Win32 systems (in other words, to Windows 95 and NT). It is available free of charge (as always) from Sunnyvale, CA-based Cygnus Solutions at `ftp://ftp.cygnus.com/pub/gnu-win32/latest`. You can choose to download some or all of the available software, ranging from user utilities to the GNU development environment. The former consist of most of the commonly used UNIX commands and utilities.

Listing 1 contains some examples that will give you a flavor of what running these utilities on a Windows NT system is like. As you can see, `grep` and `ls` work as expected. The `ps` command is more limited.

The `find` command is somewhat nonstandard in that it is the usual GNU version of this utility, but it works the same way on Windows NT systems as it does on other systems where the GNU utilities are installed. For example, the following command is one way to find

NTegration

all files in the current directory having the extension `html`:

```
C:\> find . -regex .*\.html -print
./TIPS.html
./TIPS2.html
./TIPS3.html
```

You can combine these commands with Windows NT commands as desired. For example, the first command in Listing 2 extracts a `ps`-style process list from the (overly verbose) output of the `pstat` command (included in the Resource Kit). In the same vein, the second command in Listing 2 displays the five highest priority processes currently running on the system.

The Hamilton C Shell

The Cygnus collection of GNU utilities includes the `bash` shell. Users who prefer a C shell may be interested in the excellent Hamilton C Shell package from Hamilton Laboratories, Sudbury, MA (it costs \$350). It includes a C shell supporting all of the usual features (aliases and command completion are the most important to me); most existing C shell scripts will run correctly without modification.

The package also includes a variety of UNIX-style utilities that may be run from within the C shell or from the Windows

NT command line. Among them are the following utilities, which are missing from the GNU set:

- A `cron`-style facility (named “cron” but implemented somewhat differently).
- A `df` command, which is inexplicably named `du` (see Listing 3).
- An `mt` command for manipulating tapes.
- The `uudecode` and `uuencode` utilities.
- A shell script functioning as a `whereis` command.

The package also includes a number of “extras,” including many cool sample scripts and the following useful utilities:

- `des` – A DES encryption utility/filter.
- `binedit` – A binary file editor (`sed`-like).
- `xd` – A hexadecimal dump tool.
- An enhanced version of the `pwd` command, which displays the current directory on every disk drive on the system:

```
C:\> pwd
c:\ntreskit\perl
d:\aefrisch\columns
e:\
...
k:\
l:\hamilton\bin
```

Listing 1. Sample Commands from the Cygnus GNU Utilities

```
C:\> grep "u.* .*see[^\ ].* .*p" *.html
TIPS3.html:submit the solution to a problem you've seen occur, please

C:\> ls -l *.html
-rw-r--r--  1 544      everyone    22256 Mar 29 18:54 TIPS.html
-rw-r--r--  1 544      everyone    12843 Mar 29 15:55 TIPS2.html
-rw-r--r--  1 544      everyone     3980 Mar 28 19:05 TIPS3.html

C:\> ps -ef
  PID      PPID      WIN32-PID  UID  COMMAND
  1000     1000           327   500  C:\\ps.exe
```

Listing 2. Combining Windows NT and GNU Commands

```
C:\> pstat | awk "/Pri/ || /^ 0:00/ {print $0}"
  User Time   Kernel Time   Ws   Faults   Commit Pri Hnd Thd Pid Name
  0:00:00.000 20:58:41.945   16     1         0   0   0   1   0 Idle Process
  0:00:00.000 0:00:56.240   120    2016      36   8 251 30   2 System
  0:00:00.180 0:00:00.220   120    2032     164  11  36   6  23 smss.exe
  0:00:03.575 0:00:21.420  1304   1899    1220  13 246   7  31 csrss.exe
  0:00:02.012 0:00:08.722   568   15470    668  13  67   2  37 WINLOGON.EXE
  ...

C:\> pstat | awk "/Pri/ || /^ 0:00/ {print $0}" | sort +5 | head -6
  User Time   Kernel Time   Ws   Faults   Commit Pri Hnd Thd Pid Name
  0:00:00.000 21:00:11.243   16     1         0   0   0   1   0 Idle Process
  0:00:00.000 0:00:00.000    52     10        108   8   5   1  196 sort.exe
  0:00:00.000 0:00:00.010    52     10        104   8   5   1  236 head.exe
  0:00:00.020 0:00:00.010   524    128       280   8   8   1  230 awk.exe
  0:00:00.010 0:00:00.020   892    222       332   8  14   1  237 PSTAT.EXE
```

Listing 3. The Hamilton C Shell's du Command

```
C:\> du
c: 612.832 M Total = 561.337 M Used + 51.495 M (08.40%) Free ariadne
d: 511.784 M Total = 469.248 M Used + 42.536 M (08.31%) Free ananke
e: 620.206 M Total = 620.206 M Used + 0.000 M (00.00%) Free ntsrv40a
g: 1044.095 M Total = 4.421 M Used + 1039.674 M (99.58%) Free aporia
h: 522.112 M Total = 4.419 M Used + 517.693 M (99.15%) Free acrasia
i: 700.287 M Total = 4.379 M Used + 695.908 M (99.37%) Free aveya
j: 1189.888 M Total = 91.325 M Used + 1098.563 M (92.32%) Free amelia
k: 308.208 M Total = 286.457 M Used + 21.751 M (07.06%) Free amanda
l: 449.788 M Total = 165.800 M Used + 283.988 M (63.14%) Free anitra
```

Demonstration versions of the Hamilton C shell are available from the Hamilton Laboratories Web site, <http://www.hamiltonlabs.com>.

MI/X X Server Software

If you'd like to be able to use a Windows NT system as the display for X windows initiated on a UNIX system, then the free MI/X X Server software from MicroImages Inc., Lincoln, NE, may be just what you're looking for. You can download it from the Internet at <http://www.microimages.com/freestuf/mix.htm>. Once it is installed, the simplest way to use it is to initiate a telnet session from the Windows NT system to the desired UNIX system and start the X Server executable (`xs`). Then, issue the following commands on the UNIX system (where `vala` is the name of the Windows NT system):

```
unix-102>> setenv TERM xterm
unix-103>> setenv DISPLAY vala:0
unix-104>> xterm &
```

These C shell commands will cause an `xterm` window to appear within the X Server application window (as you know, the Bourne shell commands for setting environment variables are slightly different). Once it is functioning, the telnet session may be terminated.

Figure 1 illustrates the X Server environment with several X-based applications running. The X Server includes the `twm` window manager, which may be fully customized, including a startup file to automatically initiate X applications when the server is started.

OpenNT: Going All the Way

If a few user utilities are not enough to satisfy your UNIX appetite while working on a Windows NT system, consider the OpenNT package from Softway Systems Inc., San Francisco, CA. It provides an impressively rich UNIX-style working environment under Windows NT. Architecturally, OpenNT is structured as an enhanced POSIX subsystem, and so it is able

Figure 1. The MicroImages X Server



to provide not only user-level utilities but a full POSIX programming environment designed to make porting UNIX applications to Windows NT simple.

OpenNT includes four shells—`sh`, `csh`, `ksh` and `tcsh`—all of which support full job control. It also includes various UNIX commands and facilities, including some that are generally missing from the free software collections (for example, `strings` and `umask`).

The following simple examples illustrate some features of the OpenNT environment. First, here is a `find` command (which, unlike other available versions, conforms to the standard syntax):

```
> find . -mtime -1 -name \*y\* -print
/OpenNT/usr/lib/perl5/auto/DynaLoader
/OpenNT/usr/lib/perl5/opennt/5.00305...
/OpenNT/usr/lib/perl5/Sys
```

Most of the included commands work as well as this one does.

However, there are a few glitches. For example, while the `ps` command uses the proper display format, it still does not work as expected:

```
> ps -ef
USER      PID      PPID  ELAPSED  TTY      TIME    CMD
197108   1638400    1      0:54.95  n00      0:00.45  tcsh
197108   6422529  1638400  0:00.03  n00      0:00.02  ps -ef
```

This is not a bug but simply a limitation of using a POSIX subsystem as the product's base: Only those processes running under the subsystem will appear in the list.

Mixing Windows NT commands and OpenNT is possible but sometimes a bit tedious. For example, consider this command to set the C shell prompt:

```
> set prompt = "`HOSTNAME.EXE`-!\!>> "
vala^M-58>> _
```

The `hostname` command is a Windows NT command (located in `C:\WinNT\System32`). In order for it to be found, its location must be in the search path, and its full name must be entered in the correct case (uppercase in this instance). Extensions are not applied by default to command names, and all pathnames are truly case-sensitive, both of which are completely in line with standard UNIX but are inconvenient when you have gotten used to Windows NT's laxness.

OpenNT includes many X clients, and the server version also includes an X Server—which can be purchased separately. Figure 2 illustrates a Windows NT desktop when the X Server is running. As Figure 2 demonstrates, this product makes it easy to run UNIX, X and Windows NT applications simultaneously. In this example, we have an `xterm` window on a remote UNIX host, a Windows NT file browsing window, a Microsoft Word session, two other X-based utilities (one of which is running from the remote UNIX system), as well as the usual Windows NT icons on the desktop. Each process runs in its own window, and multiple processes of any type are supported.

Prices for the OpenNT products start at \$229 for the Lite

version (no X Server) and \$379 for the single-system Workstation version. Server versions (which include a `telnet` server product) range from \$979 for one to 25 users to \$1,899 for unlimited users. The Software Development Kit costs \$199.

Other Stuff

Almost every UNIX facility that you might want to put on a Windows NT system is out there somewhere. Here are sources for two of the most requested classes of tools:

- Perl for Win32 systems is available at <http://www.perl.com/perl/info/software.html>.
- Various TCP/IP daemons and World Wide Web-related facilities can be obtained at Jim Buyens' amazing software site, see <http://www.primenet.com/~buyensj/ntwebserv.html>. ✍

Eleen Frisch is systems administrator for a very heterogeneous network of UNIX and NT systems. She is also the author of the book Essential System Administration (O'Reilly & Associates Inc., now in its second edition) and is currently finishing up another on Windows NT systems administration. In her (almost nonexistent) spare time, she enjoys painting and lounging around with her cats, Daphne and Sarah. Email: aefrisch@lorentzian.com.

Figure 2. The OpenNT X-Based Environment

