Introduction to Unix

M. PAVEL

New York University, New York, New York 10003

The Unix operating system, developed by Bell Laboratories and distributed by Western Electric, represents a new direction in the design of an operating system. This paper describes the system and its utility in the field of psychological research.

Unix is an operating system developed at Bell Laboratories and distributed by Western Electric. It is enjoying increasingly wider acceptance among diversified groups of users, which is particularly significant because the amount of available software support is very limited. Among the reasons for its popularity is that it offers a user a nice working environment and many powerful software tools. The most comprehensive description of the Unix operating system can be found in a special issue of the *Bell System Technical Journal* (1978, Vol. 57, No. 6, Part 2).

The Unix operating system was written to run on relatively small machines such as the Digital Equipment Corporation PDP-11 series of CPUs. Most of the system code is written in a high-level programming language, C. As a result, the system is fairly portable and easily modifiable, a nice, but potentially dangerous feature. Although Unix is a small system, it offers a surprising number of important and sophisticated features. For example, the hierarchical file system permits an easy and efficient organization of a large number of files for many different users. The fact that the data format of files, devices, and interprocess communications are compatible allows considerable freedom in software development. In addition, Unix provides the capability to initiate asynchronous processes.

The value of these features is apparent to anyone experienced in the development of nontrivial software. However, even novices can greatly benefit from the relatively friendly environment offered by Unix.

UNIX FILE SYSTEM

An essential aspect of the file system is the fact that all files are treated the same way. Thus the files containing data, program source, or executable binary code can be accessed in the same manner as are directories and devices.

The system is organized as a tree structure, with various directories as its nodes and other files as its leaves. An example of such a tree is shown in Figure 1. The root of the tree is a directory denoted by `"/`. In the example in Figure 1, the root directory contains a file, `"A,"` and three directories, `"B,"` `"C,"` and `"D."` Each file is uniquely identified by a path to it from the root, as shown in Figure 1.

Unix is a trademark of Bell Laboratories and Western Electric.

Figure 1. An example of a Unix file system. Each node of the tree represents a file. For clarity, directories are distinguished from the rest of the files by depicting them as circles. The subtree identified by the dashed line represents a removable file system mounted on directory `"D."` The character strings `"/A,"` `"/B/E/K,"` and `"/D/M/K"` uniquely identify the files to which they point.

In addition to files permanently residing in the system, it is possible to add (or mount) a removable file system that is stored on a separate disk. The removable file system can be attached to a directory (e.g., `"D"` in Figure 1), and it becomes a part of the existing system.

When a user logs on, Unix puts him into a predetermined directory. From then on, the user can move (depending on his authorization) from one directory to another. When in a given directory, the immediate children (files in that directory) can be accessed directly without specifying the complete path; for example, while in the directory `"/B/E,"` the file `"/B/E/K"` can be accessed by specifying only `"K."`

Security of the system is implemented using passwords and user identification numbers. Each file can be protected independently from reading, writing, and executing by other users.

COMMAND INTERPRETER

The user interface in Unix is implemented by a program called `"Shell."` It is a kind of command interpreter with programming features. A basic command recognized by Shell consists of a string of the form `"file_name arg1 arg2 arg3. . .,"` where `file_name` is the name of an executable file and `arg1`, `arg2`, and so on, are

optional arguments. For example, "pr fil1 fil2" prints files fil1 and fil2. The specification of file names as arguments can be simplified by using the wild-card construction, string_1*string_2, where "*" is replaced by any number of characters. For example, "pr fil*" can replace the form above (pr fil1 fil2) if the directory contains only two files with the initial letters "fil."

A convenient feature of Unix is I/O redirection. The programs that assume a standard input and output can be made to read (write) arbitrary files as follows: command <fin>fout, where "command" is an executable file, "fin" is an input file, and "fout" is an output file. The redirection is accomplished by the symbols > and <.

Another very useful feature of Unix is "Pipe." Pipe is a mechanism for interprocess communication that has the following form: command_1 | command_2, where the output of command_1 is taken as an input to command_2. Thus for example, "sort fil* | pr" first sorts files fil1 and fil2 and then prints them.

Most of the jobs that do not require an interactive input can be run in the background by appending "&" after a command line: command arg_list &, where the system returns before the command finishes executing.

"Shell," however, is more than a command interpreter. It has many constructs usually found in algorithmic languages. It allows passing parameters, evaluates variables, and recognizes a number of control-flow primitives: "while" and "for" loops, "if-then-else," "case," and so on. These primitives enable a user to create files containing Shell commands that are capable of making decisions by conditional execution of various commands. The files can then be executed, and so they provide a powerful instrument for combining other software tools.

UTILITIES

Most of the utilities and subsystems are independent of the system kernel. Because of this modular design, they are easily modifiable and maintainable and can therefore be tailored to many specific applications and particular users.

Among the utilities that are standard with Unix are many programs for manipulation of files, such as "copy," "move," "remove," "concatenate," and "list." A number of different editors are written for Unix, including the line-oriented editor originally supplied with the system (ed) and a number of screen editors.

Additional utilities are provided for interuser communication (mail, write), status information, and many administrative functions. The system also has a convenient software for communication between any two Unix machines. Finally, the system contains programs for system generation and maintenance.

LANGUAGE SUBSYSTEMS

Unix supports a large number of languages; the most important and convenient is C. C is a general-purpose

programming language with modern control-flow and data structures, in many ways similar to PASCAL.

Besides C, many traditional languages are supported under Unix: FORTRAN, BASIC, SNOWBOL, and even LISP. Unix also supports several compiler-compilers that are used to write parsers and lexical analyzers.

TEXT PROCESSING

One of the most widely used features of Unix is its text processing environment. Documents are prepared using one of the available editors. Error correction is facilitated by programs that either use a dictionary look-up or a statistical method to detect probable spelling mistakes. A corrected manuscript is processed by formatter programs and then sent to an output device (a typewriter or phototypesetter). In addition to the standard text formatter (nroff or troff), there are special-purpose programs particularly suitable for scientific writing: a preprocessor for mathematical typing (eqn), a generator of tables (tbl), a reference processor (refer), and a permuted index generator (ptx).

CONCLUSION

The Unix operating system represents a new direction in the design of an operating system. Its utility has been further increased because the system has been implemented on a large number of different computers. At present, it is possible to operate Unix on various PDP-11s, VAX, Perkin-Elmer (Interdata), Honeywell Level 6, Zilog 8000, Motorola 68000, and others.

One aspect of Unix that detracted from its appeal was the lack of software support. Neither Bell Laboratories nor Western Electric offered software assistance to users outside the Bell System. However, several software companies have recently offered to support a number of different versions of Unix. This service is costly, but it may well be worthwhile.

Unix is a very appropriate tool for many applications in the field of psychological research. These applications range from preparation of stimuli and experimental software to data analyses and text processing. Unix is also suitable for running certain experiments that are not too time critical.

Unix is more difficult to apply in real-time experimental work requiring fast machine response and high data rates. The standard versions do not allow sufficient temporal promptness, although there are solutions to this problem. Some laboratories developed satellite systems in which the software is developed under Unix and then downloaded to a satellite processor that mediates all time-critical functions. Other laboratories have modified the system to minimize the system overhead.

REFERENCE

UNIX Time-Sharing System, *Bell System Technical Journal*, 1978, 57, 1897-2305.