I/O Systems

The two main jobs of a computer are I/O and processing. In many cases, the main job is I/O, and the processing is merely incidental. For instance, when we browse a Web page or edit a file, our immediate interest is to read or enter some information, not to compute an answer.

The role of the operating system in computer I/O is to manage and control I/O operations and I/O devices. Although related topics appear in other chapters, here we bring together the pieces to paint a complete picture of I/O. First, we describe the basics of I/O hardware, because the nature of the hardware interface places constraints on the internal facilities of the operating system. Next, we discuss the I/O services provided by the operating system and the embodiment of these services in the application I/O interface. Then, we explain how the operating system bridges the gap between the hardware interface and the application interface. We also discuss the UNIX System V STREAMS mechanism, which enables an application to assemble pipelines of driver code dynamically. Finally, we discuss the performance aspects of I/O and the principles of operating-system design that improve I/O performance.

Bibliographical Notes

[Vahalia (1996)] provides a good overview of I/O and networking in UNIX. [Leffler et al. (1989)] detail the I/O structures and methods employed in BSD UNIX. [Milenkovic (1987)] discusses the complexity of I/O methods and implementation. The use and programming of the various interprocess-communication and network protocols in UNIX are explored in [Stevens (1992)]. [Brain (1996)] documents the Windows NT application interface. The I/O implementation in the sample MINIX operating system is described in [Tanenbaum and Woodhull (1997)]. [Custer (1994)] includes detailed information on the NT message-passing implementation of I/O.

For details of hardware-level I/O handling and memory-mapping functionality, processor reference manuals ([Motorola (1993)] and [Intel (1993)]) are among the best sources. [**?**] describe multiprocessor systems and cache-consistency issues. [Tanenbaum (1990)] describes hardware I/O design at a low level, and [Sargent and Shoemaker (1995)] provide a programmer's guide to low-level PC hardware and software. The IBM PC device I/O address map is given in [IBM (1983)]. The March 1994 issue of **IEEE Computer** is devoted to I/O hardware and software. [Rago (1993)] provides a good discussion of STREAMS.

Bibliography

[Brain (1996)] M. Brain, *Win32 System Services,* Second Edition, Prentice Hall (1996).

[Custer (1994)] H. Custer, *Inside the Windows NT File System*, Microsoft Press (1994).

[IBM (1983)] *Technical Reference*. IBM Corporation (1983).

[Intel (1993)] *Pentium Processor User's Manual, Volume 3: Architecture and Programming Manual*. Intel Corporation (1993).

[Leffler et al. (1989)] S. J. Leffler, M. K. McKusick, M. J. Karels, and J. S. Quarterman, *The Design and Implementation of the 4.3BSD UNIX Operating System*, Addison-Wesley (1989).

[Milenkovic (1987)] M. Milenkovic, *Operating Systems: Concepts and Design*, McGraw-Hill (1987).

[Motorola (1993)] *PowerPC 601 RISC Microprocessor User's Manual*. Motorola Inc. (1993).

[Rago (1993)] S. Rago, *UNIX System V Network Programming*, Addison-Wesley (1993).

[Sargent and Shoemaker (1995)] M. Sargent and R. Shoemaker, *The Personal Computer from the Inside Out,* Third Edition, Addison-Wesley (1995).

[Stevens (1992)] R. Stevens, *Advanced Programming in the UNIX Environment*, Addison-Wesley (1992).

[Tanenbaum (1990)] A. S. Tanenbaum, *Structured Computer Organization,* Third Edition, Prentice Hall (1990).

[Tanenbaum and Woodhull (1997)] A. S. Tanenbaum and A. S. Woodhull, *Operating System Design and Implementation,* Second Edition, Prentice Hall (1997).

[Vahalia (1996)] U. Vahalia, *Unix Internals: The New Frontiers*, Prentice Hall (1996).