

Introduction



An **operating system** is a program that manages a computer's hardware. It also provides a basis for application programs and acts as an intermediary between the computer user and the computer hardware. An amazing aspect of operating systems is how they vary in accomplishing these tasks. Mainframe operating systems are designed primarily to optimize utilization of hardware. Personal computer (PC) operating systems support complex games, business applications, and everything in between. Operating systems for mobile computers provide an environment in which a user can easily interface with the computer to execute programs. Thus, some operating systems are designed to be *convenient*, others to be *efficient*, and others to be some combination of the two.

Before we can explore the details of computer system operation, we need to know something about system structure. We thus discuss the basic functions of system startup, I/O, and storage early in this chapter. We also describe the basic computer architecture that makes it possible to write a functional operating system.

Because an operating system is large and complex, it must be created piece by piece. Each of these pieces should be a well-delineated portion of the system, with carefully defined inputs, outputs, and functions. In this chapter, we provide a general overview of the major components of a contemporary computer system as well as the functions provided by the operating system. Additionally, we cover several other topics to help set the stage for the remainder of this text: data structures used in operating systems, computing environments, and open-source operating systems.

Bibliographical Notes

[Brookshear (2012)] provides an overview of computer science in general. Thorough coverage of data structures can be found in [Cormen et al. (2009)].

[Russinovich and Solomon (2009)] gives an overview of Microsoft Windows and covers considerable technical detail about the system internals and components. [McDougall and Mauro (2007)] cover the internals of the Solaris operating system. Mac OS X internals are discussed in [Singh (2007)]. [Love

(2010)] provides an overview of the Linux operating system and great detail about data structures used in the Linux kernel.

Many general textbooks cover operating systems, including [Stallings (2011)], [Deitel et al. (2004)], and [Tanenbaum (2007)]. [Kurose and Ross (2013)] provides a general overview of computer networks, including a discussion of client-server and peer-to-peer systems. [Tarkoma and Lagerspetz (2011)] examines several different mobile operating systems, including Android and iOS.

[Hennessy and Patterson (2012)] provide coverage of I/O systems and buses and of system architecture in general. [Bryant and O'Hallaron (2010)] provides a thorough overview of a computer system from the perspective of a computer programmer. Details of the Intel 64 instruction set and privilege modes can be found in [Intel (2011)].

The history of open sourcing and its benefits and challenges appears in [Raymond (1999)]. The Free Software Foundation has published its philosophy in <http://www.gnu.org/philosophy/free-software-for-freedom.html>. The open source of Mac OS X are available from <http://www.apple.com/opensource/>.

Wikipedia has an informative entry about the contributions of Richard Stallman at http://en.wikipedia.org/wiki/Richard_Stallman.

The source code of Multics is available at http://web.mit.edu/multics-history/source/Multics_Internet_Server/Multics_sources.html.

Bibliography

- [Brookshear (2012)] J. G. Brookshear, *Computer Science: An Overview, Eleventh Edition*, Addison-Wesley (2012).
- [Bryant and O'Hallaron (2010)] R. Bryant and D. O'Hallaron, *Computer Systems: A Programmers Perspective - Second Edition*, Addison-Wesley (2010).
- [Cormen et al. (2009)] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Third Edition*, MIT Press (2009).
- [Deitel et al. (2004)] H. Deitel, P. Deitel, and D. Choffnes, *Operating Systems, Third Edition*, Prentice Hall (2004).
- [Hennessy and Patterson (2012)] J. Hennessy and D. Patterson, *Computer Architecture: A Quantitative Approach, Fifth Edition*, Morgan Kaufmann (2012).
- [Intel (2011)] *Intel 64 and IA-32 Architectures Software Developer's Manual, Combined Volumes: 1, 2A, 2B, 3A and 3B*. Intel Corporation (2011).
- [Kurose and Ross (2013)] J. Kurose and K. Ross, *Computer Networking—A Top-Down Approach, Sixth Edition*, Addison-Wesley (2013).
- [Levy (1994)] S. Levy, *Hackers*, Penguin Books (1994).
- [Love (2010)] R. Love, *Linux Kernel Development, Third Edition*, Developer's Library (2010).
- [McDougall and Mauro (2007)] R. McDougall and J. Mauro, *Solaris Internals, Second Edition*, Prentice Hall (2007).

- [**Raymond (1999)**] E. S. Raymond, *The Cathedral & the Bazaar*, O'Reilly & Associates (1999).
- [**Russinovich and Solomon (2009)**] M. E. Russinovich and D. A. Solomon, *Windows Internals: Including Windows Server 2008 and Windows Vista, Fifth Edition, Fifth Edition*, Microsoft Press (2009).
- [**Singh (2007)**] A. Singh, *Mac OS X Internals : A Systems Approach*, Addison-Wesley (2007).
- [**Stallings (2011)**] W. Stallings, *Operating Systems, Seventh Edition*, Prentice Hall (2011).
- [**Tanenbaum (2007)**] A. S. Tanenbaum, *Modern Operating Systems - Third Edition*, Prentice Hall (2007).
- [**Tarkoma and Lagerspetz (2011)**] S. Tarkoma and E. Lagerspetz, "Arching over the Mobile Computing Chasm: Platforms and Runtimes", *IEEE Computer*, Volume 44, (2011), pages 22–28.

