PREFACE

Linux

A Practical Guide to Linux® Commands, Editors, and Shell Programming, Second Edition, explains how to work with the Linux operating system from the command line. The first few chapters of this book quickly bring readers with little computer experience up to speed. The rest of the book is appropriate for more experienced computer users. This book does not describe a particular release or distribution of Linux but rather pertains to all recent versions of Linux.

Mac OS X

This book also explains how to work with the UNIX/Linux foundation of Mac OS X. It looks "under the hood," past the traditional graphical user interface (GUI) that most people think of as a Macintosh, and explains how to use the powerful command-line interface (CLI) that connects you directly to OS X. As with the Linux releases, this book does not describe a particular release of OS X but rather pertains to all recent releases. Where this book refers to Linux, it implicitly refers to Mac OS X as well and makes note of differences between the two operating systems.

Command-line interface (CLI)

In the beginning there was the command-line (textual) interface (CLI), which enabled you to give Linux commands from the command line. There was no mouse to point with or icons to drag and drop. Some programs, such as emacs, implemented rudimentary windows using the very minimal graphics available in the ASCII character set. Reverse video helped separate areas of the screen.

Linux was born and raised in this environment, so naturally all the original Linux tools were invoked from the command line. The real power of Linux still lies in this environment, which explains why many Linux professionals work *exclusively* from the command line. Using clear descriptions and lots of examples, this book shows you how to get the most out of your Linux system using the command-line interface.

Linux distributions

A Linux distribution comprises the Linux kernel, utilities, and application programs. Many distributions are available, including Ubuntu, Fedora, Red Hat, Mint, OpenSUSE, Mandriva, CentOS, and Debian. Although the distributions differ from one another in various ways, all of them rely on the Linux kernel, utilities, and applications. This book is based on the code that is common to most distributions. As a consequence you can use it regardless of which distribution you are running.

New in this edition

This edition includes a wealth of new and updated material:

- Coverage of the Mac OS X command-line interface (throughout the book). Part V covers utilities and highlights the differences between utility options used under Linux and those used under Mac OS X.
- An all-new chapter on the Perl scripting language (Chapter 11; page 485).
- New coverage of the rsync secure copy utility (Chapter 14; page 583).
- Coverage of more than 15 new utilities in Part V, including some utilities available under Mac OS X only.
- Three indexes to make it easier to find what you are looking for quickly. These indexes indicate where you can locate tables (page numbers followed by the letter **t**) and definitions (italic page numbers). They also differentiate between light and comprehensive coverage (page numbers in light and standard fonts, respectively).
  - The File Tree index (page 989) lists, in hierarchical fashion, most files mentioned in this book. These files are also listed in the Main index.
  - The Utility index (page 991) locates all utilities mentioned in this book. A page number in a light font indicates a brief mention of the utility; use of the regular font indicates more substantial coverage.
  - The completely revised Main index (page 995) is designed for ease of use.

Overlap

If you read *A Practical Guide to Red Hat® Linux®: Fedora™ and Red Hat Enterprise Linux, Fourth Edition,* or *A Practical Guide to Ubuntu Linux®, Second Edition,* or a subsequent edition of either book, you will notice some overlap between those books and the one you are reading now. The introduction, the appendix on regular expressions, and the chapters on the utilities (Chapter 3 of this book—*not* Part V), the filesystem, the Bourne Again Shell (bash), and Perl are very similar in the books. Chapters that appear in this book but not in the other two books include those covering the vim and emacs editors, the TC Shell (tcsh), the AWK and sed languages, the rsync utility, and Part V, which describes 97 of the most useful Linux and Mac OS X utility programs in detail.

Audience

This book is designed for a wide range of readers. It does not require programming experience, although some experience using a computer is helpful. It is appropriate for the following readers:

- **Students** taking a class in which they use Linux or Mac OS X
- **Power users** who want to explore the power of Linux or Mac OS X from the command line

- **Professionals** who use Linux or Mac OS X at work
- **Beginning Macintosh users** who want to know what UNIX/Linux is, why everyone keeps saying it is important, and how to take advantage of it
- **Experienced Macintosh users** who want to know how to take advantage of the power of UNIX/Linux that underlies Mac OS X
- **UNIX users** who want to adapt their UNIX skills to the Linux or Mac OS X environment
- **System administrators** who need a deeper understanding of Linux or Mac OS X and the tools that are available to them, including the bash and Perl scripting languages
- **Computer science students** who are studying the Linux or Mac OS X operating system
- **Programmers** who need to understand the Linux or Mac OS X programming environment
- **Technical executives** who want to get a grounding in Linux or Mac OS X

Benefits

A Practical Guide to Linux® Commands, Editors, and Shell Programming, Second Edition, gives you an in-depth understanding of how to use Linux and Mac OS X from the command line. Regardless of your background, it offers the knowledge you need to get on with your work: You will come away from this book with an understanding of how to use Linux/OS X, and this text will remain a valuable reference for years to come.

A large amount of free software has always been available for Macintosh systems. In addition, the Macintosh shareware community is very active. By introducing the UNIX/Linux aspects of Mac OS X, this book throws open to Macintosh users the vast store of free and low-cost software available for Linux and other UNIX-like systems.

In this book, *Linux* refers to *Linux* and *Mac OS X*

tip The UNIX operating system is the common ancestor of Linux and Mac OS X. Although the graphical user interfaces (GUIs) of these two operating systems differ significantly, the command-line interfaces (CLIs) are very similar and in many cases identical. This book describes the CLIs of both Linux and Mac OS X.To make it more readable, this book uses the term *Linux* to refer to both *Linux* and *Mac OS X.* It makes explicit note of where the two operating systems differ.

FEATURES OF THIS BOOK

This book is organized for ease of use in different situations. For example, you can read it from cover to cover to learn command-line Linux from the ground up. Alternatively, once you are comfortable using Linux, you can use this book as a reference: Look up a topic of interest in the table of contents or index and read about it. Or, refer to one of the utilities covered in Part V, "Command Reference."

You can also think of this book as a catalog of Linux topics: Flip through the pages until a topic catches your eye. The book also includes many pointers to Web sites where you can obtain additional information: Consider the Internet to be an extension of this book.

A Practical Guide to Linux® Commands, Editors, and Shell Programming, Second Edition, offers the following features:

- **Optional sections** allow you to read the book at different levels, returning to more difficult material when you are ready to tackle it.
- **Caution boxes** highlight procedures that can easily go wrong, giving you guidance *before* you run into trouble.
- **Tip boxes** highlight places in the text where you can save time by doing something differently or when it may be useful or just interesting to have additional information.
- **Security boxes** point out ways you can make a system more secure.
- The **Supporting Web site** at www.sobell.com includes corrections to the book, downloadable examples from the book, pointers to useful Web sites, and answers to even-numbered exercises.
- Concepts are illustrated by **practical examples** found throughout the book.
- The many useful **URLs** (Internet addresses) identify sites where you can obtain software and information.
- **Chapter summaries** review the important points covered in each chapter.
- **Review exercises** are included at the end of each chapter for readers who want to hone their skills. Answers to even-numbered exercises are available at www.sobell.com.
- Important **GNU tools**, including gcc, GNU Configure and Build System, make, gzip, and many others, are described in detail.
- Pointers throughout the book provide help in obtaining **online documentation** from many sources, including the local system and the Internet.
- Important command-line utilities that were developed by Apple specifically for Mac OS X are covered in detail, including diskutil, ditto, dscl, GetFileInfo, launchctl, otool, plutil, and SetFile.
- Descriptions of Mac OS X **extended attributes** include **file forks**, **file attributes**, **attribute flags**, and **Access Control Lists** (**ACLs**).
- Appendix D, "Mac OS X Notes," lists some differences between Mac OS X and Linux.

CONTENTS

This section describes the information that each chapter covers and explains how that information can help you take advantage of the power of Linux. You may want to review the table of contents for more detail.

- **Chapter 1—Welcome to Linux and Mac OS X**
  Presents background information on Linux and OS X. This chapter covers the **history of Linux**, profiles the OS X **Mach kernel**, explains how the GNU Project helped Linux get started, and discusses some of **Linux's important features** that distinguish it from other operating systems.

PART I: THE LINUX AND MAC OS X OPERATING SYSTEMS

Experienced users may want to skim Part I

tip If you have used a UNIX/Linux system before, you may want to skim or skip some or all of the chapters in Part I. All readers should take a look at "Conventions Used in This Book" (page 24), which explains the typographic conventions that this book uses, and "Where to Find Documentation" (page 33), which points you toward both local and remote sources of Linux documentation.

Part I introduces Linux and gets you started using it.

- **Chapter 2—Getting Started**
  Explains the **typographic conventions** this book uses to make explanations clearer and easier to read. This chapter provides basic information and explains how to log in, **change your password**, give Linux commands using the shell, and **find system documentation**.
- **Chapter 3—The Utilities**
  Explains the **command-line interface** (CLI) and briefly introduces **more than 30 command-line utilities**. Working through this chapter gives you a feel for Linux and introduces some of the tools you will use day in and day out. The utilities covered in this chapter include
  - grep, which **searches through files** for strings of characters;
  - unix2dos, which **converts Linux text files** to Windows format;
  - tar, which **creates archive files** that can hold many other files;
  - bzip2 and gzip, which **compress files** so that they take up less space on disk and allow you to transfer them over a network more quickly; and
  - diff, which **displays the differences** between two text files.

- **Chapter 4—The Filesystem**
  Discusses the Linux hierarchical filesystem, covering files, filenames, **pathnames**, working with directories, **access permissions**, and hard and **symbolic links**. Understanding the filesystem allows you to **organize your data** so that you can find information quickly. It also enables you to **share some of your files** with other users while **keeping other files private**.

- **Chapter 5—The Shell**
  Explains how to use shell features to make your work faster and easier. All of the features covered in this chapter work with both bash and tcsh. This chapter discusses

  - Using **command-line options** to modify the way a command works;
  - Making minor changes in a command line to **redirect input** to a command so that it comes from a file instead of the keyboard;
  - **Redirecting output** from a command to go to a file instead of the screen;
  - Using **pipes** to send the output of one utility directly to another utility so you can solve problems right on the command line;
  - Running programs in the **background** so you can work on one task while Linux is working on a different one; and
  - Using the shell to **generate filenames** to save time spent on typing and help you when you do not remember the exact name of a file.

PART II: THE EDITORS

Part II covers two classic, powerful Linux command-line text editors. Most Linux distributions include the vim text editor, an "improved" version of the widely used vi editor, as well as the popular GNU emacs editor. Text editors enable you to create and modify text files that can hold programs, shell scripts, memos, and input to text formatting programs. Because Linux system administration involves editing text-based configuration files, skilled Linux administrators are adept at using text editors.

- **Chapter 6—The vim Editor**
  Starts with a **tutorial** on vim and then explains how to use many of the **advanced features** of vim, including special characters in search strings, the General-Purpose and Named buffers, parameters, markers, and execution of commands from within vim. The chapter concludes with a **summary of vim commands**.

- **Chapter 7—The emacs Editor**
  Opens with a **tutorial** and then explains many of the features of the emacs editor as well as how to use the META, ALT, and ESCAPE keys. In addition, this

chapter covers key bindings, buffers, and **incremental and complete searching** for both character strings and regular expressions. It details the relationship between Point, the cursor, Mark, and Region. It also explains how to take advantage of the extensive **online help** facilities available from emacs. Other topics covered include cutting and pasting, using multiple windows and frames, and working with emacs modes—specifically **C mode**, which aids programmers in writing and debugging C code. Chapter 7 concludes with a **summary of emacs commands**.

PART III: THE SHELLS

Part III goes into more detail about bash and introduces the TC Shell (tcsh).

- **Chapter 8—The Bourne Again Shell**
  Picks up where Chapter 5 left off, covering more advanced aspects of working with a shell. For examples it uses the Bourne Again Shell—bash, the shell used almost exclusively for system shell scripts. Chapter 8 describes how to
  - Use shell **startup files**, shell options, and shell features to **customize the shell**;
  - Use **job control** to stop jobs and move jobs from the foreground to the background, and vice versa;
  - Modify and reexecute commands using the **shell history** list;
  - Create **aliases** to customize commands;
  - Work with **user-created and keyword variables** in shell scripts;
  - Set up **functions**, which are similar to shell scripts but are executed more quickly;
  - Write and execute simple **shell scripts**; and
  - **Redirect error messages** so they go to a file instead of the screen.
- **Chapter 9—The TC Shell**
  Describes tcsh and covers features common to and different between bash and tcsh. This chapter explains how to
  - Run tcsh and **change your default shell** to tcsh;
  - **Redirect error messages** so they go to files instead of the screen;
  - Use **control structures** to alter the flow of control within shell scripts;
  - Work with tcsh **array and numeric variables**; and
  - Use shell **builtin commands**.

PART IV: PROGRAMMING TOOLS

Part IV covers important programming tools that are used extensively in Linux and Mac OS X system administration and general-purpose programming.

Chapter 10—Programming the Bourne Again Shell
Continues where Chapter 8 left off, going into greater depth about advanced shell programming using bash, with the discussion enhanced by extensive examples. This chapter discusses

- **Control structures** such as **if...then...else** and **case**;
- **Variables**, including locality of variables;
- **Arithmetic and logical** (**Boolean**) **expressions**; and
- Some of the most useful **shell builtin commands**, including exec, trap, and getopts.

Once you have mastered the basics of Linux, you can use your knowledge to build more complex and specialized programs, using the shell as a programming language.

Chapter 10 poses two complete **shell programming problems** and then shows you how to solve them step by step. The first problem uses **recursion** to create a hierarchy of directories. The second problem develops a quiz program, shows you how to set up a shell script that **interacts with a user**, and explains how the script processes data. (The examples in Part V also demonstrate many features of the utilities you can use in shell scripts.)

- **Chapter 11—The Perl Scripting Language**
Introduces the popular, feature-rich Perl programming language. This chapter covers
  - Perl **help tools** including perldoc;
  - Perl **variables** and **control structures**;
  - **File handling**;
  - **Regular expressions**; and
  - Installation and use of **CPAN modules**.

  Many Linux administration scripts are written in Perl. After reading Chapter 11 you will be able to better understand these scripts and start writing your own. This chapter includes many examples of Perl scripts.

- **Chapter 12—The AWK Pattern Processing Language**
Explains how to write programs using the powerful AWK language that filter data, **write reports**, and **retrieve data from the Internet**. The advanced programming section describes how to set up **two-way communication** with another program using a **coprocess** and how to obtain input over a network instead of from a local file.

- **Chapter 13—The sed Editor**
  Describes sed, the **noninteractive stream editor** that finds many applications as a filter within shell scripts. This chapter discusses how to use sed's buffers to write **simple yet powerful programs** and includes many examples.
- **Chapter 14—The rsync Secure Copy Utility**
  Covers rsync, a secure utility that copies an ordinary file or directory hierarchy locally or between the local system and another system on a network. As you write programs, you can use this utility to back them up to another system.

PART V: COMMAND REFERENCE

Linux includes hundreds of utilities. Chapters 12, 13, and 14 as well as Part V provide extensive examples of the use of 100 of the **most important utilities** with which you can solve problems without resorting to programming in C. If you are already familiar with UNIX/Linux, this part of the book will be a valuable, **easy-to-use reference**. If you are not an experienced user, it will serve as a useful supplement while you are mastering the earlier sections of the book.

Although the descriptions of the utilities in Chapters 12, 13, and 14 and Part V are presented in a format similar to that used by the Linux manual (man) pages, they are much easier to read and understand. These utilities are included because you will work with them **day in and day out** (for example, ls and cp), because they are **powerful tools** that are especially useful in shell scripts (sort, paste, and test), because they help you **work with a Linux system** (ps, kill, and fsck), or because they enable you to **communicate with other systems** (ssh, scp, and ftp). Each utility description includes complete explanations of its most useful options, differentiating between options supported under Mac OS X and those supported under Linux. The "Discussion" and "Notes" sections present **tips and tricks** for taking full advantage of the utility's power. The "**Examples**" **sections** demonstrate how to use these utilities in real life, alone and together with other utilities to generate reports, summarize data, and extract information. Take a look at the "Examples" sections for AWK (more than 20 pages, starting on page 541), ftp (page 707), and sort (page 819) to see how extensive these sections are.

PART VI: APPENDIXES

Part VI includes the appendixes, the glossary, and three indexes.

- **Appendix A—Regular Expressions**
  Explains how to use **regular expressions** to take advantage of the **hidden power of Linux**. Many utilities, including grep, sed, vim, AWK, and Perl, accept regular expressions in place of simple strings of characters. A single regular expression can match many simple strings.

- **Appendix B—Help**
  Details the steps typically used to **solve the problems** you may encounter with a Linux system. This appendix also includes many **links to Web sites** that offer **documentation**, useful Linux and Mac OS X information, mailing lists, and **software**.
- **Appendix C—Keeping the System Up-to-Date**
  Describes how to use tools to download software and **keep a system current**. This appendix includes information on
  - yum—Downloads software from the Internet, keeping a system up-to-date and **resolving dependencies** as it goes.
  - apt-get—An alternative to yum for keeping a system current.
  - **BitTorrent—**Good for distributing large amounts of data such as Linux installation CDs.
- **Appendix D—Mac OS X Notes**
  This appendix is a brief guide to Mac OS X features and quirks that may be unfamiliar to users who have been using Linux or other UNIX-like systems.
- **Glossary**
  Defines more than 500 terms that pertain to the use of Linux and Mac OS X.
- **Indexes**
  - **File Tree Index—**Lists, in hierarchical fashion, most files mentioned in this book. These files are also listed in the Main index.
  - **Utility Index—**Locates all utilities mentioned in this book.
  - **Main Index—**Helps you find the information you want quickly.

SUPPLEMENTS

The author's home page (www.sobell.com) contains downloadable listings of the longer programs from this book as well as pointers to many interesting and useful Linux- and OS X-related sites on the World Wide Web, a list of corrections to the book, answers to even-numbered exercises, and a solicitation for corrections, comments, and suggestions.

THANKS

First and foremost I want to thank my editor at Prentice Hall, Mark L. Taub, who encouraged me and kept me on track. Mark is unique in my experience: He is an editor who works with the tools I am writing about. Because Mark runs Linux on

his home computer, we share experiences as I write. His comments and direction were invaluable. Thank you, Mark T.

Molly Sharp of ContentWorks worked with me during the day-by-day production of this book, providing help, listening to my rants, and keeping everything on track. Thanks to Jill Hobbs, Copyeditor, who made the book readable, understandable, and consistent; and Andrea Fox, Proofreader, who made each page sparkle and found the mistakes that the author left behind.

Thanks also to the folks at Prentice Hall who helped bring this book to life, especially Julie Nahil, Full-Service Production Manager, who oversaw production of the book; John Fuller, Managing Editor, who kept the large view in check; Brandon Prebynski, Marketing Manager; Curt Johnson, Executive Marketing Manager; Kim Boedigheimer, Editorial Assistant, who attended to the many details involved in publishing this book; Heather Fox, Publicist; Cheryl Lenser, Senior Indexer; Sandra Schroeder, Design Manager; Chuti Prasertsith, Cover Designer; and everyone else who worked behind the scenes to make this book come into being.

A big "Thank You" to the folks who read through the drafts of the book and made comments that caused me to refocus parts of the book where things were not clear or were left out altogether: David L. Adam; Joe Barker, Ubuntu Forums LoCo Admin; Mike Basinger, Ubuntu Forum Administrator; Michael Davis, Systems Analyst, CSC; Andy Lester, author of *Land the Tech Job You Love: Why Skill and Luck Are Not Enough,* who helped extensively with the Perl chapter; John Peters, J&J Associates; Rich Rosen, Interactive Data; Max Sobell, New York University; Leon Towns-von Stauber; and Jarod Wilson, Senior Software Engineer, Red Hat, Inc.

I am also indebted to Denis Howe, the editor of *The Free On-Line Dictionary of Computing* (FOLDOC). Denis has graciously permitted me to use entries from his compilation. Be sure to visit the dictionary (www.foldoc.org).

Dr. Brian Kernighan and Rob Pike graciously allowed me to reprint the **bundle** script from their book, *The UNIX Programming Environment* (Prentice Hall, 1984).

Thanks also to the people who helped with the first edition of this book: Lars Kellogg-Stedman, Harvard University; Jim A. Lola, Principal Systems Consultant, Privateer Systems, LLC; Eric S. Raymond, cofounder, Open Source Initiative; Scott Mann; Randall Lechlitner, Independent Computer Consultant; Jason Wertz, Computer Science Instructor, Montgomery County Community College; Justin Howell, Solano Community College; Ed Sawicki, The Accelerated Learning Center; David Mercer, Contechst; Jeffrey Bianchine, Advocate, Author, Journalist; John Kennedy; Chris Karr; and Jim Dennis, Starshine Technical Services.

Parts of *A Practical Guide to Linux® Commands, Editors, and Shell Programming, Second Edition,* have grown from my previous Linux books and I want to thank the people who helped with those books. Thank you to John Dong, Ubuntu Developer, Forums Council Member; Matthew Miller, Senior Systems Analyst/Administrator, BU Linux Project, Boston University Office of Information Technology; George Vish II, Senior Education Consultant, Hewlett-Packard; James Stockford, Systemateka, Inc.;

Stephanie Troeth, Book Oven; Doug Sheppard; Bryan Helvey, IT Director, OpenGeoSolutions; Vann Scott, Baker College of Flint; David Chisnall, computer scientist extraordinaire; Thomas Achtemichuk, Mansueto Ventures; Scott James Remnant, Ubuntu Development Manager and Desktop Team Leader; Daniel R. Arfsten, Pro/Engineer Drafter/Designer; Chris Cooper, Senior Education Consultant, Hewlett-Packard Education Services; Sameer Verma, Associate Professor of Information Systems, San Francisco State University; Valerie Chau, Palomar College and Programmers Guild; James Kratzer; Sean McAllister; Nathan Eckenrode, New York Ubuntu Local Community Team; Christer Edwards; Nicolas Merline; Michael Price; Sean Fagan, Apple Computer; Paul M. Lambert, Apple Computer; Nicolas Roard; Stephen Worotynec, Engineering Services, Alias Systems; Gretchen Phillips, Independent Consultant, GP Enterprises; Peggy Fenner; Carsten Pfeiffer, Software Engineer and KDE Developer; Aaron Weber, Ximian; Cristof Falk, Software Developer at CritterDesign; Steve Elgersma, Computer Science Department, Princeton University; Scott Dier, University of Minnesota; and Robert Haskins, Computer Net Works.

Thanks also to Dustin Puryear, Puryear Information Technology; Gabor Liptak, Independent Consultant; Bart Schaefer, Chief Technical Officer, iPost; Michael J. Jordan, Web Developer, Linux Online Inc.; Steven Gibson, owner of SuperAnt.com; John Viega, founder and Chief Scientist, Secure Software, Inc.; K. Rachael Treu, Internet Security Analyst, Global Crossing; Kara Pritchard, K & S Pritchard Enterprises, Inc.; Glen Wiley, Capital One Finances; Karel Baloun, Senior Software Engineer, Looksmart, Ltd.; Matthew Whitworth; Dameon D. Welch-Abernathy, Nokia Systems; Josh Simon, Consultant; Stan Isaacs; Dr. Eric H. Herrin II, Vice President, Herrin Software Development, Inc. And thanks to Doug Hughes, long-time system designer and administrator, who gave me a big hand with the sections on system administration, networks, the Internet, and programming.

More thanks go to consultants Lorraine Callahan and Steve Wampler; Ronald Hiller, Graburn Technology, Inc.; Charles A. Plater, Wayne State University; Bob Palowoda; Tom Bialaski, Sun Microsystems; Roger Hartmuller, TIS Labs at Network Associates; Kaowen Liu; Andy Spitzer; Rik Schneider; Jesse St. Laurent; Steve Bellenot; Ray W. Hiltbrand; Jennifer Witham; Gert-Jan Hagenaars; and Casper Dik.

A Practical Guide to Linux® Commands, Editors, and Shell Programming, Second Edition, is based in part on two of my previous UNIX books: *UNIX System V: A Practical Guide* and *A Practical Guide to the UNIX System.* Many people helped me with those books, and thanks here go to Pat Parseghian, Dr. Kathleen Hemenway, Brian LaRose; Byron A. Jeff, Clark Atlanta University; Charles Stross; Jeff Gitlin, Lucent Technologies; Kurt Hockenbury; Maury Bach, Intel Israel Ltd.; Peter H. Salus; Rahul Dave, University of Pennsylvania; Sean Walton, Intelligent Algorithmic Solutions; Tim Segall, Computer Sciences Corporation; Behrouz Forouzan, DeAnza College; Mike Keenan, Virginia Polytechnic Institute and State University; Mike Johnson, Oregon State University; Jandelyn Plane, University of Maryland; Arnold Robbins and Sathis Menon, Georgia Institute of Technology; Cliff Shaffer, Virginia Polytechnic Institute and State University; and Steven Stepanek, California State University, Northridge.

I continue to be grateful to the many people who helped with the early editions of my UNIX books. Special thanks are due to Roger Sippl, Laura King, and Roy Harrington for introducing me to the UNIX system. My mother, Dr. Helen Sobell, provided invaluable comments on the original manuscript at several junctures. Also, thanks go to Isaac Rabinovitch, Professor Raphael Finkel, Professor Randolph Bentson, Bob Greenberg, Professor Udo Pooch, Judy Ross, Dr. Robert Veroff, Dr. Mike Denny, Joe DiMartino, Dr. John Mashey, Diane Schulz, Robert Jung, Charles Whitaker, Don Cragun, Brian Dougherty, Dr. Robert Fish, Guy Harris, Ping Liao, Gary Lindgren, Dr. Jarrett Rosenberg, Dr. Peter Smith, Bill Weber, Mike Bianchi, Scooter Morris, Clarke Echols, Oliver Grillmeyer, Dr. David Korn, Dr. Scott Weikart, and Dr. Richard Curtis.

I take responsibility for any errors and omissions in this book. If you find one or just have a comment, let me know (mgs@sobell.com) and I will fix it in the next printing. My home page (www.sobell.com) contains a list of errors and credits those who found them. It also offers copies of the longer scripts from the book and pointers to many interesting Linux pages.

Mark G. Sobell
San Francisco, California