

PREFACE

A Practical Guide to UNIX® for Mac OS® X Users explains how to work with the UNIX operating system that is the foundation of Mac OS X. This book 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 UNIX.

Command line
interface (CLI)

In the beginning UNIX had a command line (textual) interface. There was no mouse to point 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. In addition, reverse video helped separate areas of the screen. UNIX was born and raised in this environment.

Naturally, all of the original UNIX tools were invoked from the command line. The real power of UNIX, and of Mac OS X, lies in this environment, which explains why many UNIX 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 UNIX-based Mac OS X system using the command line interface. The first few chapters quickly bring readers with little computer experience up to speed. The rest of the book is appropriate for more experienced computer users.

Audience

This book is designed for a wide range of readers. It does not require programming experience, but assumes a basic familiarity with the Macintosh GUI. It is appropriate for the following readers:

- **Beginning Macintosh users** who want to know what UNIX is, why everyone keeps saying that 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 that underlies Mac OS X
- **Students** who are taking a class in which they use Mac OS X
- **Power users** who want to explore the power of Mac OS X from the command line
- **Professionals** who use Mac OS X at work
- **UNIX users** who want to adapt their UNIX skills to the Mac OS X environment
- **Computer science students** who are studying the Mac OS X operating system
- **Programmers** who need to understand the Mac OS X programming environment
- **Technical executives** who want to get a grounding in Mac OS X

Benefits *A Practical Guide to UNIX® for Mac OS® X Users* gives you an in-depth understanding of how to use the UNIX operating system that is the foundation for Mac OS X.

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 aspects of Mac OS X, this book throws open to Macintosh users the vast store of free and low-cost software available for UNIX, Linux, and other UNIX-like systems.

Regardless of your background, this book offers the practical knowledge you need to get on with your work. You will come away from this book understanding how to use the UNIX operating system that underlies OS X, and this text will remain a valuable reference for years to come.

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 about the UNIX aspects of Mac OS X from the ground up. Alternatively, once you are comfortable using OS X, 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 VI, “Command Reference.” You can also think of this book as a catalog of Mac OS X topics: Flip through the pages until a topic catches your eye. If you are familiar with UNIX or a UNIX-like operating system such as Linux, refer to Appendix C, “Mac OS X for UNIX Users,” which lists some of the differences between Mac OS X and traditional UNIX systems. The book also includes many pointers to Web sites where you can get additional information: Consider the Web to be an extension of this book.

A Practical Guide to UNIX® for Mac OS® X Users offers these 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 situations in which 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 that 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.
- Important command line utilities that were developed by Apple specifically for Mac OS X are covered in detail, including `GetFileInfo`, `SetFile`, `nidump`, `otool`, `launchctl`, `diskutil`, and `plutil`.
- Descriptions of Mac OS X **extended attributes** include **file forks**, **file attributes**, **attribute flags**, and **Access Control Lists (ACLs)**.
- The relationships between **GUI tools** and their **CLI counterparts** are discussed in depth.
- Information that will help you set up servers includes sections on **property lists**, the **launchd** superserver, and **DHCP**.
- A section on `NetInfo` discusses the **NetInfo database** and ways to work with it.
- Concepts are illustrated by **practical examples** found throughout the book.
- Many useful **URLs** (Internet addresses) identify **Web sites** where you can obtain software and information.
- **Chapter Summaries** review the important points covered in each chapter.
- **Review exercises** at the end of each chapter allow readers to hone their skills. Answers to even-numbered exercises are available at www.sobell.com.
- Important **GNU tools**, including `gcc`, `gdb`, 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.

CONTENTS

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

- **Chapter 1 Welcome to Mac OS X**
Presents background information on Mac OS X. This chapter covers the **history of Mac OS X**, explains the connection between OS X and open-source software including GNU and BSD software, and discusses some of **OS X's important features** that distinguish it from other operating systems, including other versions of UNIX.

PART I: THE MAC OS X OPERATING SYSTEM

Experienced users may want to skim Part I

tip If you have used a UNIX-like 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 18), which explains the typographic conventions that this book uses, and “Getting the Facts: Where to Find Documentation” (page 27), which points you toward both local and remote sources of Mac OS X documentation.

Part I introduces UNIX on a Mac OS X system and gets you started using it from the command line.

- **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, **display a shell prompt**, give OS X commands using the shell, and **find system documentation**.
- **Chapter 3 The Mac OS X 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 UNIX and introduces some of the tools you will use day in and day out. Chapter 3 also introduces pipes, which allow you to combine utilities on the command line. The utilities covered in this chapter include
 - ◆ **grep**, which **searches through files** for strings of characters;
 - ◆ **ditto**, which **copies files and directories** (folders);
 - ◆ **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 Mac OS X Filesystem**
Discusses the OS X hierarchical filesystem from a UNIX perspective, covering files, filenames, **pathnames**, working with directories (folders), **access permissions**, and hard and **symbolic links**. This chapter also discusses **extended attributes** including file forks, attribute flags, type codes, creator codes, and **Access Control Lists (ACLs)**. 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 the Bourne Again Shell (bash) and the TC Shell (tcsh). This chapter discusses
 - ◆ Using **command line options** to modify the way a command works;
 - ◆ How a minor change in a command line can **redirect input** to a command to come from a file instead of from the keyboard;
 - ◆ How to **redirect output** from a command to go to a file instead of to the screen;
 - ◆ Using **pipes** to send the output of one utility directly to another utility so that you can solve problems right on the command line;
 - ◆ Running programs in the **background** so that you can work on one task while UNIX is working on a different one; and
 - ◆ Using the shell to **generate filenames** to save you time spent on typing and help you when you do not recall the exact name of a file.

PART II: THE EDITORS

Part II covers two classic, powerful UNIX command line text editors. Mac OS X includes 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.

- **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 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. The chapter also covers key bindings, buffers, and **incremental and complete searching** for both character strings and regular expressions. In addition, 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 (tsh).

- **Chapter 8 The Bourne Again Shell**

Picks up where Chapter 5 leaves 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 (preference) files**, shell options, and shell features to **customize your 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 can execute more quickly;
- ◆ Write and execute simple **shell scripts**; and
- ◆ **Redirect error messages** so that they go to a file instead of the screen.

- **Chapter 9 The TC Shell**

Describes tsh and covers features that are common to and differ between bash and tsh. This chapter explains how to

- ◆ Run tsh and **change your default shell** to tsh;
- ◆ **Redirect error messages** so that they go to files instead of to the screen;
- ◆ Use **control structures** to alter the flow of control within shell scripts;
- ◆ Work with **tsh array and numeric variables**; and
- ◆ Use shell **builtin commands**.

PART IV: MORE ABOUT MAC OS X

- **Chapter 10 Networking**
Discusses networks, network security, and the Internet. This chapter
 - ◆ Covers types of networks, subnets, protocols, addresses, hostnames, and various network utilities;
 - ◆ Discusses distributed computing including the client/server model; and
 - ◆ Describes briefly some of the servers you can use on a network.
- **Chapter 11 System Maintenance**
Discusses core system concepts including Superuser and the use of `sudo` as well as system operation. This chapter
 - ◆ Provides general information about how to set up a server, including information on property list and other configuration files;
 - ◆ Describes how the `launchd` superserver works;
 - ◆ Discusses the NetInfo database and delves into how it differs from the classic UNIX system databases;
 - ◆ Describes briefly many important standard directories and files;
 - ◆ Explains how to set up a `chroot` jail; and
 - ◆ Describes how DHCP works.

PART V: PROGRAMMING TOOLS

Part V covers programming under Mac OS X. It discusses the C programming environment, the use of `bash` as a programming language, and ways to write programs using `awk` and `sed`.

- **Chapter 12 Programming Tools**
Introduces Mac OS X's exceptional programming environment. This chapter focuses on the traditional UNIX command line development environment because this environment is the one you will likely have to work in with free software developed for UNIX systems. This chapter
 - ◆ Discusses the Carbon, Cocoa, and UNIX APIs;
 - ◆ Explains how to invoke the GNU `gcc` compiler;
 - ◆ Describes how to use `make` to keep a set of **programs up-to-date**;
 - ◆ Explains how to **debug a C program** using `gdb`;
 - ◆ Describes how to work with **shared libraries**;
 - ◆ Explains how to set up and use CVS to **manage and track program modules** in a software development project; and
 - ◆ Discusses **system calls** and explains how you can use them to initiate kernel operations.

- **Chapter 13 Programming the Bourne Again Shell**

Once you have mastered the basics of OS X, you can use your knowledge to build more complex and specialized programs, by using the shell as a programming language. Chapter 13 shows you how to use `bash` to write advanced shell scripts. It 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`.

Chapter 13 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 and shows you how to set up a shell script that **interacts with a user** and how the script processes data. (The examples in Part VI also demonstrate many features of the utilities you can use in shell scripts.)

- **Chapter 14 The awk Pattern Processing Language**

Explains how to write programs using the powerful `awk` language that **filter data** and **write reports**. The advanced programming section describes the powerful `getline` statement, which gives you better control over bringing data into `awk` and the `system` statement, which allows you to run external commands from inside an `awk` program.

- **Chapter 15 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.

PART VI: COMMAND REFERENCE

Mac OS X includes hundreds of utilities. Chapters 14 and 15 as well as Part VI provide extensive examples of the use of more than 90 of the **most important utilities** with which you can solve problems without resorting to programming in C. If you are already familiar with UNIX, 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 14 and 15 and Part VI are presented in a format similar to that used by the Mac OS X manual (`man`) pages, they are much easier to read and understand. These utilities were chosen 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 your Mac OS X system** (`ps`, `kill`,

sysctl, and GetFileInfo), or because they enable you to **communicate with other systems** (ssh, scp, and ftp). Part VI also covers several utilities that were developed by Apple for OS X—they are not found on other UNIX systems (nidump, otool, launchctl, and plutil).

Each utility description includes complete explanations of its most useful options. The “Discussion” and “Notes” sections present **tips and tricks** for using the utility to full advantage. 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 609), **ftp** (page 738), and **sort** (page 837) to see how extensive these sections are.

PART VII: APPENDIXES

Part VII includes the appendixes and the glossary.

- **Appendix A Regular Expressions**

Explains how to use **regular expressions** to take advantage of the **hidden power of UNIX**. Many utilities, including **grep**, **sed**, **vim**, and **awk**, 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 Mac OS X/UNIX system. This appendix also includes many **links to Web sites** that offer **documentation**, useful UNIX information, mailing lists, and **software**.

- **Appendix C Mac OS X for UNIX Users**

This appendix is a brief guide to Mac OS X features and quirks that may be unfamiliar to users who have been using other UNIX or UNIX-like systems.

- **Glossary**

Defines more than 500 terms that pertain to the use of Mac OS X/UNIX.

The index helps you quickly find the information you want.

SUPPLEMENTS

Mark Sobell’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 UNIX-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 FROM MARK G. SOBELL

First and foremost I want to thank my editor at Prentice Hall, Mark L. Taub, who encouraged me and kept me on track. His comments and direction were invaluable. Thank you, Mark T.

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: Thank you to David Chisnall, computer scientist extraordinaire; Sean Fagan, Apple Computer; Paul M. Lambert, Apple Computer; Nicolas Roard; Stephen Worotyneć, Engineering Services, Alias Systems; Gretchen Phillips, Independent Consultant, GP Enterprises; and Peggy Fenner.

Thanks to Molly Sharp of ContentWorks, the production manager who made sure the book came together as it was supposed to; Jill Hobbs, the copy editor who kept the various parts of the English language in their proper relative places; and Andrea Fox, the most amazing proofreader I have ever worked with. Thanks also to the folks at Prentice Hall who helped bring this book to life: Kim Silvestro, Publicist; Stephane Nakib, Marketing Manager; Julie Nahil, Full-Service Production Manager; Noreen Regina, Editorial Assistant; and everyone else who worked behind the scenes to make this book happen.

I am also indebted to Denis Howe, the editor of *The Free On-Line Dictionary of Computing* (FOLDOC). Dennis 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).

Parts of *A Practical Guide to UNIX® for Mac OS® X Users* have grown from my UNIX and Linux books, and I want to thank the people who helped with those books.

Thank you to 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; Jim Dennis, Starshine Technical Services; Carsten Pfeiffer, Software Engineer and KDE Developer; Aaron Weber, Ximian; Matthew Miller, Boston University; Cristof Falk, Software Developer at Critter-Design; Scott Mann, IBM, Systems Management and Integration Professional; 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, Capitol One Finances; Karel Baloun, Senior Software Engineer, Looksmart, Ltd.; Matthew Whitworth; Dameon D. Welch-Abernathy, Nokia Systems; Josh Simon, Consultant; Stan Isaacs; and 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 Hagenaaars; and Casper Dik.

A Practical Guide to UNIX® for Mac OS® X Users is based in part on two of my early 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, and 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, for reviewing the book.

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.

We take responsibility for any errors and omissions in this book. If you find one or just have a comment, let us know (mgs@sobell.com) and we will fix it in the next printing. Mark Sobell's 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 Mac OS X Web sites.

