

ter
Technology Electronic Reviews
Formerly *Telecommunications Electronic Reviews*

Volume 10, Number 4, December, 2003

Technology Electronic Reviews (TER) is a publication of the Library and Information Technology Association.

Technology Electronic Reviews (ISSN: 1533-9165) is a periodical copyright © 2003 by the American Library Association. Documents in this issue, subject to copyright by the American Library Association or by the authors of the documents, may be reproduced for noncommercial, educational, or scientific purposes granted by Sections 107 and 108 of the Copyright Revision Act of 1976, provided that the copyright statement and source for that material are clearly acknowledged and that the material is reproduced without alteration. None of these documents may be reproduced or adapted for commercial distribution without the prior written permission of the designated copyright holder for the specific documents.

REVIEW OF: David A. Karp, Tim O'Reilly, Troy Mott. (2002). *Windows XP in a Nutshell*. Sebastopol, CA: O'Reilly.

by Tim Daniels

As computers are replaced and new operating systems are installed, Windows XP is becoming the operating system if not of choice, then of convenience for many libraries. Windows XP brings changes and added features to the windows operating system. The one thing you need to help navigate these changes is a good guide, *Windows XP In a Nutshell* is that guide. No matter if you are a power user or a regular user, this is the book for you.

Windows XP in a Nutshell has three impressive authors. David A. Karp, who has a degree in mechanical engineering from the University of California at Berkeley, is an expert in user-interface design and software development. He is the author of the best-selling Windows Annoyances books, and maintains the www.annoyances.org (<http://www.annoyances.org>) web site. He also writes articles on computing for magazines such as *Windows Sources*, *InfoWorld* and *PC Computing*. Tim O'Reilly is the founder and president of O'Reilly and Associates and co-authors many of the books published by O'Reilly. Troy Mott is a

corporate services agent for Studio B. He is a co-author of O'Reilly's *Palm OS Network Programming*, and *Photoshop in a Nutshell: A Desktop Quick Reference*. With this depth of experience and O'Reilly's format, *Windows XP In a Nutshell* is an excellent companion to the XP operating system.

The well-organized contents of this extensive reference manual (600+ pages) allow readers to find the information they need quickly. The book is broken down into nine chapters and four parts. The nine chapters include: The Lay of the Land, Using Windows XP, The User Interface, Windows XP Applications and Tools, Task and Setting Index, The Command Prompt, Networking, The Registry, and The Windows Script Host. The first of four parts, The Big Picture, is an overview chapter that explains not only how to use *Windows XP in a Nutshell* but also gives an overview of the Windows operating system. Part II, Alphabetical References, contains alphabetically organized references for each major element of Windows XP. Part III, Advanced Topics, covers more advanced topics including networking and the registry. Part IV, the Appendices are one of the most valuable parts of the book: Appendix A, Installing Windows XP; Appendix B, Migrating to Windows XP; Appendix C, Keyboard Shortcuts; Appendix D, Power Toys and TweakUI; Appendix E, Keyboard Equivalents; Appendix F, Common Filename Extensions; and Appendix G, Services.

The first part of the book discusses the history of the Windows operating system. The authors explain that until the release of XP all previous versions were just graphical ways to launch DOS applications. Unlike the previous incarnations of the Windows operating system, XP is developed around the popular Windows NT operating system that was marketed primarily to business users. This change makes Windows XP far more stable than any of the other versions of the Windows operating systems targeted to the public.

At the heart of *Windows XP in a Nutshell* is an alphabetic 250-page directory of all the applications included in Windows XP and directions on how to use each one. XP's new user interface with its cartoon like presentation is also discussed. The theory behind the new interface is it is designed to be more like a web site and therefore easier to navigate. Users do have the option of changing the interface to look like the classic Windows operating system.

For advanced users there is a complete discussion of networking and how to configure Windows XP to work in a networked environment. The registry is also discussed. Many of Windows XP's features are controlled by the registry, and a great deal of data about how XP is being used is collected in the registry. The third section covers how to edit the registry to make XP even more useful. It should also be noted that early in part three a discussion on the possible dangers of editing the registry is also included.

The Appendices of *Windows XP in a Nutshell* could be a complete book unto itself. This section covers the installation process, migration procedures, and has a great introduction to "TweakUI" as well as a list of shortcut keys and file extension names.

I have been using O'Reilly books for several years now and have yet to find one that I did not find to be well written and informative; *Windows XP in a Nutshell* is no exception. Clear writing, targeted towards computer users, combined with screen shots makes this book usable for readers who are comfortable with computers and even for some who are not. *Windows XP in a Nutshell* is highly recommended to anyone who uses or administers Windows XP.

Tim Daniels, (mailto:tim.daniels@library.gatech.edu)tim.daniels@library.gatech.edu
(mailto:tim.daniels@library.gatech.edu), Information Services Department, Georgia Tech
Library and Information Center in Atlanta.

Copyright © 2003 by Tim Daniels. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at tim.daniels@library.gatech.edu (mailto:tim.daniels@library.gatech.edu).

REVIEW OF: Dave Taylor and Jerry Peck. (2002). *Learning Unix for Mac OS X*. Sebastopol, CA: O'Reilly.

by Janet A. Crum

What is the most widely distributed Unix-based operating system in use today? Surprise: it's Mac OS X, Apple's newest operating system for the Macintosh (The power of Unix, 2003). First released in 2001, OS X consists of a graphical user interface called Aqua running on a Unix core called Darwin. Darwin is an open source operating system that combines elements of Berkeley Software Distribution (BSD) Unix with other Unix services such as the Apache Web server (Darwin FAQ, 2002). Aqua continues Apple's longstanding tradition of shielding Mac users from the nuts and bolts of their operating system. However, the power of Darwin's Unix command line is available via the Terminal program in OS X, and *Learning Unix for Mac OS X* provides a concise, effective introduction to Unix for Mac users who wish to launch Terminal and explore the mysterious world of `grep` and `vi`.

The purpose of this book is simple and straightforward, "to teach the basics of Unix to Macintosh users" (vii). The authors, both longtime Unix users and developers, fulfill this purpose admirably, packing a solid introduction to Unix in just 139 pages: a volume slim enough to tote around with an iBook or PowerBook for quick reference while on the go. They cover lots of material in such a small package by focusing on the most commonly used commands, programs, features, and options, with enough detail to be useful without overwhelming the user. Mac specific information is included where appropriate, but most of the content applies to other Unix environments, too.

Early chapters cover basic principles; while more sophisticated functions appear in the latter half of the book. Chapter 1, Getting Started, explains how to launch Terminal, introduces the Unix shell prompt, and explains how to enter, recall, and correct commands. In Chapter 2, Using Unix, the authors introduce the file system, covering the home directory concept, relative pathnames, changing the working directory, listing files, viewing files, setting file permissions, and changing your password. Chapter 3, File Management, provides more detail on interacting with files, including file and directory names and wildcards, text editing with an emphasis on `vi`, creating directories, and moving, copying, renaming, finding, and deleting files. Chapter 4, Customizing Your Session, covers customizing the Terminal application and the shell environment. Chapter 5, Printing, will likely prove the most daunting to new Unix users. It includes several pages on configuring an `lpr` printer, as well as how to format files for printing and how to print to an AppleTalk printer. In Chapter 6, Redirecting I/O, the authors cover several powerful Unix tools for manipulating data in files. These tools include `cat`, `grep`, and `sort`, along with instructions for using `i/o` redirection, pipes, and filters with these commands.

Chapters 7 and 8 (Unix-Based Internet Tools) cover Internet access in the Unix environment. In Chapter 7, Accessing the Internet, the authors explain remote logins using `telnet`, `rlogin`, and `ssh` (secure shell), and cover file transfer with `scp` (secure copy), `rcp` (remote copy), and `ftp` (file transfer protocol). Chapter 8

addresses Web browsing with Lynx, reading e-mail and Usenet news with Pine, and chatting with Talk and IRC (Internet Relay Chat). Readers should note that several of these programs, e.g. Pine, are not included with Mac OS X. The authors explain how to download and install them.

Chapter 9, Multitasking, explains how to run commands in the background, monitor the status of running processes, and kill processes. Chapter 10, Where to Go from Here, points the reader to additional sources of information, including man(ual) pages and Web sites, and suggests areas for further investigation, such as shell programming and creating aliases for commands. The book concludes with a brief appendix on configuring Sendmail to allow you to send e-mail from the Unix command line. Also included is a pullout quick reference card, featuring common commands and functions with their correct syntax.

The authors convey information effectively without intimidating new Unix users or insulting their intelligence. The writing style is typical of O'Reilly books, somewhat informal but reasonably serious, with clear, well-written instructions, well-chosen examples, and no hype or excessive use of exclamation points or distracting icons.

Learning Unix for Mac OS X is highly recommended for intermediate and advanced Mac users who would like to get acquainted with Unix and discover the power of OS X's Unix underpinnings. Experienced Unix users making the switch to OS X will find this volume too basic; they would do much better with a standard OS X reference, such as David Pogue's *Mac OS X: The Missing Manual* (2002, 2nd ed., O'Reilly). Mac users who already know the basics of Unix but would like to dig deeper than what is covered in this book should read one or more good Unix books and check out the following sites:

Apple's Unix and Open Source Downloads - http://www.apple.com/downloads/macosx/unix_open_source/ (http://www.apple.com/downloads/macosx/unix_open_source/) (including MySQL, Pine, and OpenOffice.)

The Fink Project - <http://fink.sourceforge.net/> (<http://fink.sourceforge.net/>) (A site which ports Unix open source software to Darwin for use with Mac OS X. This site includes games, editors, Perl libraries, Internet applications, graphics applications, and lots more.)

References:

Darwin FAQ. (2002). Retrieved January 25, 2003, from <http://developer.apple.com/darwin/projects/darwin/faq.html> (<http://developer.apple.com/darwin/projects/darwin/faq.html>).

The power of Unix. (2003). Retrieved January 25, 2003, from <http://www.apple.com/macosx/jaguar/unix.html> (<http://www.apple.com/macosx/jaguar/unix.html>).

Janet A. Crum (crumj@ohsu.edu (<mailto:crumj@ohsu.edu>)) is Head, Library Systems & Cataloging at Oregon Health & Science University in Portland, Oregon.

Copyright (c) 2003 by Janet A. Crum. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at crumj@ohsu.edu (<mailto:crumj@ohsu.edu>).

REVIEW OF: D. Scott Brandt. (2002). Teaching Technology: A How-To-Do-It Manual for Librarians. New York: Neal-Schuman Publishers.

by Stacey Greenwell

Creating a technology training program or improving upon an existing one will be made easier by using *Teaching Technology: A How-To-Do-It Manual for Librarians*. From developing an individual technology-training course, to implementing a successful training program in the library, this book covers the methods, gives numerous specific examples, and provides samples throughout. For anyone considering a training program, to anyone with an established program in place, this book will provide thorough methodology and helpful examples to build or improve a training program.

The first half of the book is devoted to the ADDIE method of developing a technology-training course. ADDIE consists of Analysis, Design, Development, Implementation, and Evaluation. Each chapter explains a step of the process quite thoroughly, providing abundant examples.

The first chapter, Analysis, covers the basic issues in developing a training course-- what does the learner need or want from the training? The author encourages the trainer to consider different learning styles in developing training. Some ways to uncover those styles include self-assessment of the students. Examples of what to survey and how to do this in the planning stages are included in this chapter.

In the Design chapter, the author explains the need to determine prerequisite learning for the class participants as part of building course objectives. The author provides an example of determining prerequisites - in this example, that motor skills are involved in pointing, clicking, and selecting, and then goes as far as including a section, "Why determining prerequisites is important." This straightforward approach to design clearly illustrates each step and makes this a very hands-on, practical manual for planning a technology-training course.

The Development chapter covers the importance of objectives and offers suggestions for building course objectives. The author discusses strategies for facilitating learning in the classroom, from lecture to group interaction, and offers examples and suggestions for each method. Also provided in the chapter are suggestions for building handouts and exercises, and the author even includes a sample handout and exercise for searching journal articles.

From preparing for a session to dealing with problem participants, the Implementation chapter covers both what to prepare before entering the classroom and what to do during the session. Some topics in this section include benefits of an assistant or co-teacher, dealing with failing technology in the class, and preparing a back up plan.

While reading this book, I had the opportunity to "test" some of the methods in the book, as I was planning a training course on the use of Microsoft Outlook. Particularly helpful for me was the Evaluation section, as I had always felt that evaluation of previous training that I done was quite lacking. In this section, the author starts from the beginning, defining evaluation, and then moving to the why, when, and where of evaluation. The section includes several example evaluations, including an example of a particularly "bad" evaluation which offered the most generic questions and went so far as to offer responses as smiley faces or sad faces--a good example of an evaluation that would tell the instructor quite little about the outcome of the course! The author offers ideas on what you need to evaluate and provides suggestions for evaluation techniques.

The second half of the book is devoted to building an effective technology-teaching program from start to finish. First steps that are covered include analyzing the current environment in the library-from the library's mission and goals to the number of workstations and the number of technical support calls in a particular

area. Other important topics covered include budgeting, tips for getting buy-in from the administration, promotion (marketing and advertising), and implementation (from scheduling and registration to evaluation).

The book concludes with an exploration of successful training programs. The programs featured are from a wide range of libraries, including the New York Public Library, Science, Industry, and Business Library; University of Texas at Austin General Libraries; Ithaca College Library; Business Research Center, Andersen (Atlanta); Arlington Heights Memorial Library (Illinois); St Joseph County Public Library (Indiana); and Staff Development and Training Program, Purdue University Libraries (the author's own training program). The programs featured, highlight the details of the training program, including the type of training offered, as well as when, where, and to whom it is offered. The programs also point out their most successful techniques as well as the classes that are most popular. In continuing with the copious examples provided throughout the book, the appendix includes a one-hour lecture on searching indexes, an instructor's guide and sample handouts for searching EBSCO, as well as sample exercises and a course evaluation for the session.

For current trainers as well as those investigating a technology-training program, this book will be a well-used, welcome addition to one's bookshelf.

Stacey Greenwell (staceyg@email.uky.edu (mailto:staceyg@email.uky.edu)) Stacey Greenwell is Desktop Support Librarian for the University of Kentucky Libraries in Lexington, Kentucky.

Copyright (c) 2003 by Stacey Greenwell. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at staceyg@email.uky.edu (mailto:staceyg@email.uky.edu).

REVIEW OF: David Tansley. (2002). Create Dynamic Web Pages Using PHP and MYSQL. London, England: Pearson Education Ltd.

by John Wynstra

PHP is a server-side scripting language that, when combined with a database (in our case mySQL) and a web server (in our case Apache), can be used to create dynamic and powerful web sites. Most of the dynamic web sites that we encounter today, including search engines, e-commerce sites, and product catalogs, are built using server-side scripting languages such as PHP, ASP (Microsoft), JSP (JAVA), CGI, and Cold Fusion. PHP currently accounts for roughly 15 percent of this dynamic content. One of the great advantages of PHP, mySQL, and Apache is that they are all open source and free. MySQL does offer a commercial license, but for many applications it is free.

In this book, the author David Tansley establishes himself as a down to earth programmer who has a lot of wisdom to impart in a very understandable manner. Unfortunately, this book has many serious proofreading errors that make it difficult to recommend. If these errors were related merely to bad grammar or punctuation, I could almost look the other way, due to the very practical and easy to understand approach the author takes. Unfortunately, the book is aimed at new and inexperienced users and is intended to function as a hands-on tutorial. Therein lays my dilemma. The book contains many working sample scripts,

as well as detailed instructions for loading software and running the scripts. Those instructions and a few of the scripts contain a number of errors that make it very difficult for the inexperienced but aspiring programmer to successfully test the examples in this book.

The most significant problems come right in Chapter 1. This chapter gives a concise introduction to how the web works, and provides step-by-step instructions for loading the software that is included on the CDROM that comes with the book. The software includes Netscape Navigator Version 4.76, MySQL Version 3.23.33, Apache Version 1.3.17, and PHP Version 4.04. Correctly installing this software is essential to running the hands-on exercises contained in this book. The author even states this at the end of Chapter 1, when he tells the reader that they can't really proceed until successfully installing the software. Unfortunately, the instructions as written in Chapter 1 do not work for installing MySQL Apache, or PHP. The instructions contain numerous errors including, incorrect character case (Linux is case sensitive), syntax errors, and even incorrect keyword usage (e.g., use of activate rather than enable (p.10)). In each case, I had to do something other than what the instructions indicated in order to install the software package successfully. An inexperienced user would easily get frustrated and give up before page 16 of the book. While there is a continual pattern of noticeable grammatical errors in this book, once you do get a working PHP environment set up and begin running the exercises, there is a lot of valuable information here. I make no claim to being an expert in grammar, but the errors in this book were sloppy and distracting.

The author is systematic in his approach; he builds new concepts upon previous concepts and always works with one idea at a time. He does a good job of covering the right amount of PHP language syntax, MySQL, SQL language, HTML forms processing, and Apache in order to give the reader the necessary background to understand and implement the examples in the book. This is no small feat, since each of these individual topics already has numerous in-depth books written about it. The author successfully gives practical coverage to broad concepts, leaving the fine details to other, more focused books. This is a positive and refreshing aspect to this book.

The book progresses from software installation in Chapter 1 to language syntax in Chapters 2-7. It is filled with PHP scripts. As a matter of fact, you can't turn more than a few pages without coming across a script, and most scripts are complete and ready to be plugged in and run. This makes the language syntax chapters interesting and practical as opposed to long and theoretical. Every concept can be tested immediately. Chapters 8-12 cover forms, form processing, email, web server variables, as well as maintaining state, and they include many more useful examples to test. Chapters 13-14 introduce MySQL to the equation, and Chapters 15-17 put everything together, with each chapter serving as a stand-alone sample application. The sample applications are nicely done and serve as models for applications that the reader would likely want to build anyway. The last two chapters cover Apache authentication mechanisms and PHP utilization of these mechanisms.

The sample scripts listed in the book are also available on the CD. I came across a number of small errors in the scripts in the book (a missing quote or tab, a misspelling), but most of the copies on the CD were functional and ran without a problem when copied into the test environment. There were one or two PHP scripts that were not on the CD, and none of the SQL scripts were included on the CD. The SQL scripts would have been useful, but it is possible that the author wanted the reader to have a little more command line experience with MySQL.

I hesitate to point out the negative aspects of this book so strongly, since I personally found it to be one of the more useful PHP books that I have come across. The problem that I see is that Chapter 1 will be a showstopper for many inexperienced users. I had enough background working with Linux to see the errors in the instructions and find my way around them. I cannot assume that this will be the case for other

readers. With that in mind, I hope that there is a second edition to this book that is carefully proof read and tested before being placed on the market. My hats off to the author for writing a book that is so practical and easy to understand. It is a shame that poor proof reading had such a detrimental impact on this book.

John Wynstra (wynstra@uni.edu (mailto:wynstra@uni.edu)) is Library Information Systems Specialist for the Rod Library, University of Northern Iowa in Cedar Falls.

Copyright (c) 2003 by John Wynstra. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at wynstra@uni.edu (mailto:wynstra@uni.edu).

About TER

Editor is Sharon Rankin, McGill University (sharon.rankin@mcgill.ca (mailto:sharon.rankin@mcgill.ca)).

Editorial Board Members are: Linda Robinson Barr, Texas Lutheran University (lbarr@tlu.edu (mailto:lbarr@tlu.edu)); Paul J. Bracke, Arizona Health Sciences Library (paul@ahsl.arizona.edu (mailto:paul@ahsl.arizona.edu)); Brad Eden, University of Nevada, Las Vegas (beden@ccmail.nevada.edu (mailto:beden@ccmail.nevada.edu)); Kathlene Hanson, California State University, Monterey Bay (kathlene_hanson@csumb.edu (mailto:kathlene_hanson@csumb.edu)); Adriene Lim, Wayne State University (ab7155@wayne.edu (mailto:ab7155@wayne.edu)); Michelle Mach, Colorado State University (mmach.lib.colostate.edu); Florence Tang, Mercer University, Atlanta (tang_fy@mercer.edu (mailto:tang_fy@mercer.edu)); Stacey Voeller, Minnesota State University (voeller@mnstate.edu (mailto:voeller@mnstate.edu)); Laura Wrubel, University of Maryland (lwrubel@umd.edu (mailto:lwrubel@umd.edu)).