Is the Help Helpful? How to Create Online Help That Meets Your Users' Needs
Jean Hollis Weber. 2004. Whitefish Bay, WI: Hentzenwerke Publishing. [ISBN 1-930919-60-3. 224 pages, including index. $39.95 USD (softcover).]

Jean Hollis Weber, a well-known authority on technical editing, has written a very worthwhile book on designing and developing online help systems. *Is the Help helpful?* is brief, clearly written, and amply illustrated. It covers a broad range of topics in a prescriptive manner, generally moving from a few paragraphs of general discussion to guidelines and tips. A great deal of practical, hard-won experience comes shining through in these pages. Those new to online help will learn a great deal. Those with a strong background in help development will find the book interesting—and, in a sense, provocative.

The world of software user assistance is complex and fluid. There are many help file formats with their own features. There are also competing models of help design and varied approaches to the help development process. Therefore, any book on help will necessarily reflect particular views and commitments. This book exhibits a strong loyalty to the model of help design that arose in the early and mid-1990s during the era of WinHelp (the .hlp file). Furthermore, *Is the Help helpful?* is deeply rooted in the craft approach to help development.

Weber touches on user assistance for e-commerce Web sites, explains the distinctive navigation features of Microsoft's HTML help (.chm) and its cousins, and takes note of embedded help—with some skepticism. But WinHelp is the bedrock of her thinking. Many of her screen captures are from WinHelp help systems, and she explains and endorses almost all the features and design concepts of the WinHelp era: pop-up definitions, browse sequences, bookmarks, extensive cross-referencing among help topics, the traditional table of contents and index, and more. Weber is especially committed to F1 key context sensitivity (and equivalent techniques) for displaying procedures and conceptual information relevant to the currently displayed dialog. All of these features and concepts, integral to WinHelp, have been rejected or significantly de-emphasized in at least some of Microsoft's more recent designs. Is this, then, a backward-looking book, or is Weber elucidating enduring design ideas that are pertinent to past, present, and future help systems? Is the answer somewhere in the middle? Your answers to these questions may largely determine your judgment of the book. I chose to assign the book in my course on computer documentation.

Weber also takes a traditional stance regarding the kinds of help topics to provide and how to write them. She believes in overview and conceptual topics, reference topics, example topics, and more. She also believes in providing conceptual information in procedure topics. She is no minimalist. Furthermore, *Is the Help helpful?* does not mention journalism-inspired help articles or help that comes from bulletin boards and blogs. Ultimately, Weber puts her faith in expert technical communicators who follow a systematic, user-centered help development process.

Weber's help development process consists of these steps:
1. Analyze the audience, plan the project, and write the plan.
2. Develop high-level specifications.
3. Develop detailed specifications.
4. Perform a detailed task analysis.

5. Build and evaluate a prototype help system.
6. Develop an outline and a map of the help project.
7. Write, index, and edit the help topics.
8. Review the help topics, index, and table of contents.
9. Test the help both by itself and with the product.
10. Release the help with the product.
11. Evaluate the help and plan for improvements.

Broadly speaking, the book chapters map to this process. Weber explains each step admirably, packing lots of guidance into short chapters. For example, her chapter on analyzing audiences and tasks includes not just basic principles but also personas and use cases. Her chapter on prototyping covers the pros and cons of paper prototypes. The chapter on writing specifications includes an explanation of localization and accessibility issues. Weber departs from her organizational scheme to include a chapter setting forth common user complaints about help and a chapter on writing for diverse audiences—two very welcome topics.

In discussing the help development process, Weber draws examples from both commercial and internal software development projects. Weber's help development process, however, is decidedly craft-oriented and assumes a lot of manual checking and rechecking. It may, therefore, be more applicable to small and medium-sized help projects. Weber explains single sourcing, but single sourcing is not incorporated into her development process. Content management systems are not discussed. In short, this is a fine book, but one that does not address emerging design ideas or development methods.

The book's appendixes provide a sample documentation plan and specifications for documenting TreeLine (a personal information management application), a list of help file formats and tools, an annotated list of resources (books, Web sites, listservs, and so on), a glossary, and checklists. You can download an executable file (in Windows and Linux versions) of TreeLine along with a PDF version of the entire book.

David K. Farkas
David K. Farkas is an STC fellow and a Professor in the Department of Technical Communication at the University of Washington.

Forthcoming *Technical Communication* August, 2005