The Encyclopedic Reference as a Model for Print Documentation

David K. Farkas

A NEW WAY TO DOCUMENT COMPUTER APPLICATIONS SOFTWARE *is the encyclopedic reference* (ER), *a single alphabetically organized manual that can replace the traditional user's guide and command reference and possibly other standard documents as well. This article reviews the standard documentation set, points out strengths and weaknesses of the ER, and offers some thoughts on the future of computer documentation.*

The encyclopedic reference, or ER, is an innovative design for documenting applications software. Its defining traits are (1) that procedures, concepts, and commands are presented in the form of alphabetically organized entries, and (2) that the entries are not intended exclusively as reference material but rather are written so that broad classes of users can learn the product from them. ERs have been prepared for a variety of major software products, including *WordPerfect** and *WordStar Professional**. Furthermore, since 1987 the ER has been a prevalent design at Microsoft, the largest microcomputer software publisher, and has been used to document *Works**, *Excel**, *Word**, and as-yet-unreleased products.

The ER has important implications for the computer industry. It offers the user significant advantages over the traditional user's guide and command reference, though there are drawbacks as well. As a replacement for these two volumes, it offers software publishers the potential of reducing page count and saving on printing and binding costs. It offers writers the opportunity to produce both print and procedural online documentation as variations upon one another, and indeed the ER is the print cousin to procedural online documentation. Although this study deals only with computer documentation and specifically with applications software, the ER almost certainly can be adapted for the documentation of complex systems of many kinds.

In this study I review the components of the standard documentation set, discuss the strengths and weaknesses of the user's guide, and compare the ER to the user's guide, pointing out some promising design possibilities. I draw extensively on excellent papers by four Microsoft designers **[1-4],** but my overall approach and my assessment of the strengths of the ER are different in many respects. Two key differences are these: First, the Microsoft designers tend to assume that users will begin with the product's tutorial, whereas I see many users bypassing tutorials whenever possible; thus, I treat the ER's introductory material as a vital precursor to the alphabetically arranged information. Second, I place special value on multiple tables of contents as a means of exploiting the ER design.

THE STANDARD DOCUMENTATION SET

In its complete form, the standard documentation set includes these components: a print or online *tutorial*, a *user's guide (UG)*, a comprehensive *command reference*, and *online help*.

The *tutorial* explains at least the fundamentals of the product in the manner of a human tutor. The pacing is slow, and there are extended ex-

*Works, Excel, and Word are trademarks of the Microsoft Corporation; WordPerfect is a trademark of the WordPerfect Corporation; WordStar is a trademark of the MicroPro International Corporation.

About the Author . . .

DAVID K. FARKAS *is an associate professor in the Department of Technical Communication, College of Engineering, University of Washington. He has also been a faculty member at West Virginia University and Texas Tech University, and he holds a Ph.D. in language and literature from the University of Minnesota. At UW, he teaches computer documentation and a graduate seminar in document design. He received awards from the National Council of Teachers of English for the year's best research article (1986) and pedagogical article (1988) in the field of technical communication. An article in* Technical Communication *received the Award for Distinguished Technical Communication in 1982.*

amples for the user to follow. Tutorials are relied upon by many computer novices and by those who want a very comfortable way of learning new product features. Many users, however, seek to bypass tutorials altogether, preferring a faster pace and the opportunity to actually get work done as they learn the product.

These characteristics are found in the *UG*. Terminology in computer documentation is far from standard, and the UG is not always labeled as such. Often, it is simply regarded as a form of reference manual **[5,**12]. But however labeled, the UG is a very prevalent piece of documentation, consisting of the procedures and their attendant concepts representing the full (or nearly full) functionality of the product. The information is presented in a straightforward manner and is sequenced to correspond as closely as possible to the tasks the user performs with the product. The UG is a versatile piece of documentation, supporting users seeking an introduction to the product, more experienced users who are learning a new procedure, and users who have forgotten procedures they once knew.

Very often there is also a comprehensive *command reference* offering a terse presentation of the product's commands and other relatively atomistic kinds of information. In varying degrees, however, procedures based on the commands are also treated, resulting in overlap with the UG. Reference information is most often organized alphabetically, though it can also be organized according to a product's menu structure. The reference is most suited to users who already have a good mental model of the product. Quick-reference cards, quick-start manuals, and in-depth technical notes are other prevalent print components.

Online help can take various forms. Often information is provided for error recovery. Also information may be provided on specific commands and procedures. These last two forms of help, since they usually are written as self-contained units and designed for nonsequential access, resemble the entries of an ER.

STRENGTHS AND LIMITATIONS OF THE TRADITIONAL USER'S GUIDE

While a "flat" UG can be written for a very simple product, almost all UGs are organized hierarchically. The functionality of the product is divided into chapters, sections, and subsections. There may be, for instance, a chapter on file management containing a section on loading files and a subsection on how to load several files simultaneously. Larger-than-chapter units, often called "Parts," may also appear.

Hierarchical organization is, of course, extremely familiar and functional. Most of our conceptual thinking and most of our books are organized hierarchically. Physiologists, for example, divide the human body into systems (sensory, digestive, reproductive, etc.) and subsystems, and so a physiology textbook could well mirror this hierarchical organization.

Despite limitations, hierarchical organization will not only endure, it will flourish; and it will flourish not only in print documentation but as one form of online documentation.

But even though hierarchical organization is a venerable model, the traditional UG may not be ideally suited for all forms of computer documentation. Borland argues that as software applications grow more complex, they usually offer many more pathways through the product, making the possibility of sequencing the chapters to match the user's work agenda increasingly unlikely **[4].**

There is also reason to question the efficacy of the UG's hierarchical structure as a conceptual model of the product. Again as Borland argues, the hierarchical breakdown of large products is apt to be arbitrary and unhelpful **[4].** First, the hierarchical "tree" is apt to be very large—with as many as 20 or 30 chapters. Also, there may be product features that do not fit naturally and inevitably into higher-level categories, and there may be higher-level categories—for example, "Other Formatting Considerations" or "Tools"—that are too general to tell the user much about the product or the topics contained within. Furthermore, it is possible that one writer's hierarchical breakdown will conflict with the view of the product that the user has obtained through training or experience with one or more comparable products **[6].** Even the best hierarchical breakdown, it should be noted, does not lead, directly at least, to improved task performance **[7].**

Finally, hierarchical organization does not indicate to the user whether any particular chapter (or section) is self-contained or whether it is meant to be read after one or more of the preceding chapters (or sections). Users learn to make intelligent guesses about this, and forewords and introductions often state explicitly which chapters are prerequisite to which. But the lack of inherent cues handicaps the user.

Despite these limitations, hierarchical organization will not only endure, it will flourish; and it will flourish not only in print documentation but as one form of online documentation. It is especially suitable when there is one main pathway through a product or when one particular breakdown of product functionality seems natural and inevitable. But the UG is not the only model for computer documentation; the ER, I believe, has an important role to play.

THE STRUCTURE OF THE ENCYCLOPEDIC REFERENCE

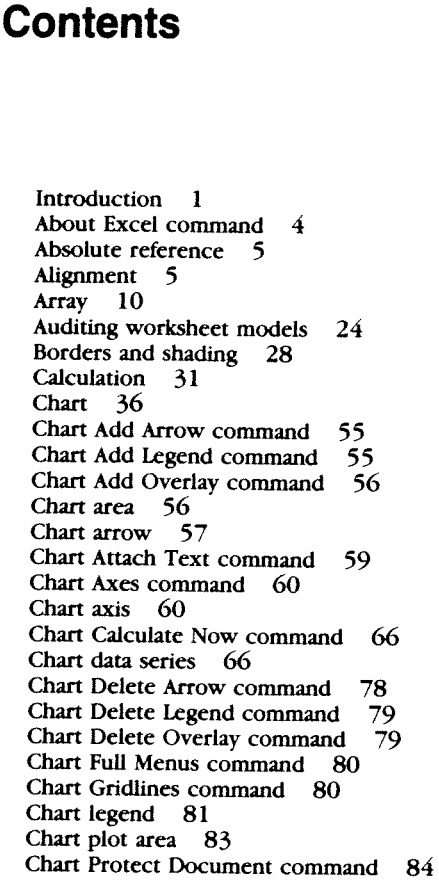
The encyclopedic reference is from a structural standpoint an expansion of traditional references. It differs, however, from the traditional alphabetical reference in important respects. Whereas the traditional reference generally consists of atomistic items of information, typically a set of commands, the ER consists of entries on procedures and concepts as well as commands, and these entries range in length from several para-

graphs to 10 or more pages. (Figure 1, a portion of the table of contents of *Excel DOS version 2.0,* shows the general structure of an ER.) Extensive cross-referencing between entries is used to direct the user to related entries—for example, from a command entry to a procedure using that command.

Whereas the traditional reference is intended primarily for users who have already attained a solid grasp of the product, the ER can serve the needs of the novice and intermediate user in the manner of a UG, making it possible to eliminate the comprehensive UG from the documentation set.

Few users, however, will choose to learn an unfamiliar piece of software using nonsequential documentation. Most want to begin with basic information—key concepts, fundamental procedures, and general principles about the function of the user interface. Special introductory material is therefore necessary. The ER for *Excel* (DOS version 2.0) contains "core topics," which, though arranged alphabetically, are singled out as introductory. *Works* (DOS version 1.0) is an integrated four-component product, and the manual consists of four separate ERs, each beginning with an introductory section. The manual for *Word* (Macintosh version 3.0) is purely alphabetical, but is accompanied by a short, noncomprehensive UG that is, in essence, an introductory entry bound separately from the rest of the manual. Without some sort of "UG-type" introductory material, the user must go to the tutorial to learn the basic concepts and procedures of the product, which is the case in *WordPerfect (DOS version 5.0).*

An idea we can call "forced graduation" arises here. Without a comprehensive UG in the documentation set, the user is required to "graduate" to the ER's nonsequential (alphabetic) presentation—or else fall back upon the tutorial. Ideally, the nonsequential entries should be comprehensive, and the introductory, UG-type material should be as brief as possible, since it duplicates the nonsequential material. But only experience and usability testing can determine how short the introductory material can be and how soon the user should be graduated to nonsequential presentation. If, as in the case of the *Wordstar Professional* documentation set (version 5.0), a comprehensive UG precedes an ER, the user receives two complete manuals and there is no forced graduation.



Contents

Figure 1. A portion of the table of contents of an encyclopedic reference.

In one respect the UG and ER lie opposite one another, near the poles of a single continuum. UGs with extensive appendixes are less fully hierarchical and are more like ERs. ERs which have more comprehensive introductory material or which are composed of broader entries are more like UGs. Indeed, a UG with alphabetized chapters is a kind of ER, though probably not an effective implementation.

STRENGTHS AND LIMITATIONS OF THE ENCYCLOPEDIC REFERENCE

As noted above, one advantage of the ER, when it fulfills the functions of both a UG and a traditional reference, is that the user need work with only one major piece of print documentation; the concomitant benefit to the software publisher is the potential for reduced page count and reduced printing and binding costs **[1].**

Another advantage, the one stressed by the Microsoft authors, is faster access to information. In the best case, the user, knowing the relevant term for the target information, thumbs directly to the entry of interest, without even consulting the table of contents (TOC). If this search proves to be a near miss, a cross reference should direct the user to the exact information being sought. Even with these two look-ups, the user has done as well as the UG user, who, accessing information by concept, must scan the 1st-level TOC entries looking for one that encompasses the target information and must then scan the 2nd- or even 3rd-level entries under this 1st-level entry to confirm that the target information in indeed there.

But as Prekeges notes, access by terminology requires that the product terminology be familiar or highly intuitive **[1].** Otherwise, users will have to go to the index, which presumably will list all synonyms for the entries in the ER. This requirement suggests that ERs are especially well suited for products in which much of the terminology is drawn largely from a task domain already familiar to the user, such as accounting or music composition. It also suggests the value of adding a hierarchical TOC (as discussed below) so that users can also search by concept.

Another benefit of the ER is that the entries are self-contained units, and any entry's relationship to other entries is spelled out explicitly in cross references. In contrast, the tendency in a UG is for one section to presuppose familiarity with earlier sections, and UG users must often judge how far they need to backtrack to pick up the prerequisite information for the section they are reading.

Perhaps the most important benefit of the ER is flexibility in design. This documentation model, which is still being explored, is capable of many variations, and these variations should

solve certain documentation problems and prove especially advantageous for certain documentation situations.

The ER for *Excel* (DOS version) utilizes several very intriguing design features, including broad "megatopic" entries that overlap other entries and offer minimalist versions of the procedures contained in these entries **[2].** Though care must be taken not to make the internal logic of the manual too complex for the user, the idea of overlapping entries exploits the fundamental nature of encyclopedic organization: entries can be added directly to meet user needs, as index entries are, without the need to conform to an hierarchical breakdown of the product or to any other formal requirement. Similarly, a general purpose encyclopedia will have an entry for George Washington and another for the American Revolution.

A design alternative that I believe should be explored carefully is to provide multiple tables of contents, each made distinctive through color coding or similar means. A hierarchical TOC (with some special treatment if there are overlapping entries) would afford the user many of the benefits of a UG. Another TOC could be based on the menu structure of the product, and there might be separate TOCs for commands, concepts, procedures, and error messages. A very successful reference book that makes good use of multiple TOCs is Brusaw, Alred, and Oliu's *Handbook of Technical Writing* **[8].**

One advantage of online documentation over print is that various electronic indexing schemes can be used to provide access to the same online database in a variety of ways. The use of multiple TOCs affords some of this power to the ER. Multiple TOCs can be used with a traditional user's guide as well; but they work better when the manual consists of self-contained entries.

CONCLUSION: THOUGHTS ON THE FUTURE OF PRINT DOCUMENTATION

Much computer documentation has moved and will continue to move online. Tutorials are one excellent candidate for online presentation: the large number of pages required by print tutorials ceases to be an issue, and tutorials in particular benefit from sound, animation, and two-way interaction with the user. Error recovery information also belongs online, because this kind of documentation is so intimately connected to the current state of the computer.

Print manuals, however, will remain important, not only outside the realm of the computer, but as a form of computer documentation. A likely role for print is presenting straightforward procedural information, the province of the UG and ER. Online presentation offers less compelling advantages here, and users value portability, the superior legibility of print, and the fact that print documentation doesn't cover any portion of the screen. Users also like the physicality of print: whereas online information appears as disembodied screens that rise out of the computer and go back again, manuals reveal their structure and the location of the user within that structure more readily and provide tactile access to the information.

But which kind of procedural manual is best? The ER offers the possibility of a single multipurpose print volume, and multiple TOCs can provide an equivalent to one important online capability—access to the information through a variety of powerful electronic indexes. Also, software publishers will benefit from the similarity between the self-contained units of the ER and online procedural information. In fact, even if a software publisher were to go entirely to online documentation, a sound plan would be to allow the user to print off paper copies of the procedural files in the form of an ER.

On the other hand, many products can be effectively explained with sequential and hierarchical information. Sequential and hierarchical presentation is exactly what print has most often been used for and, arguably, what it does best.

Ultimately, the choices between print and online documentation and between the UG and ER are highly situational—they depend on the nature of the product and its users. The ER, however, is an intriguing option that designers can utilize as they face the challenge of creating the best possible documentation set. Ω

REFERENCES

1. James Prekeges, "The Once and Future Paradigm: Alphabetic References Return," *Proceedings* of the International Professional Communication Conference (New York: IEEE Professional Communication Society, 1988): 11-13. The papers by Prekeges, Slivinksi, and Grimm also appear in the 1989 ITCC *Proceedings*.
2. Jo Slivinski, "Paradigm Revisited: The Documentation Odyssey of Microsoft Excel," *Proceedings* of the International Professional Communication Conference (New York: IEEE Professional Communication Society, 1988): 15-17.
3. Rick Grimm, "Guesswork and Common Sense: The Alphabetic Reference Design Process for Microsoft Works," *Proceedings* of the International Professional Communication Conference (New York: IEEE Professional Communication Society, 1988): 19-20.
4. Russell Borland, "Documentation Paradigms" (internal Microsoft document), March 1987.
5. R. John Brockmann, *Writing Better Computer User Documentation: From Paper to Online* (New York: John Wiley & Sons), 1986.
6. Donna E. Alvermann, Lynn C. Smith, and John E. Readance, "Prior Knowledge Activation and the Comprehension of Compatible and Incompatible Text," *Reading Research Quarterly* 20, no. 4 (Summer 1985), 421-435.
7. David E. Kieras and Susan Bovair, "The Role of a Mental Model in Learning to Operate a Device," *Cognitive Science* 8, no. 3 (July-Sept. 1984), 255-273.
8. Charles T. Brusaw, Gerald J. Alred, and Walter E. Oliu, *Handbook of Technical Writing*, 3rd ed. (New York: St. Martin's Press), 1987.

THE PROOFREADER'S NIGHTMARE:
A spelling error as big as a barn.

From the collection of Herbert B. Michaelson