PROVIDING APPROPRIATE INFORMATION: AN ANALYSIS OF PROCEDURES IN COMPUTER USER DOCUMENTATION

By

ERIC J. RAY

Artium Baccalaeuri

Washington University

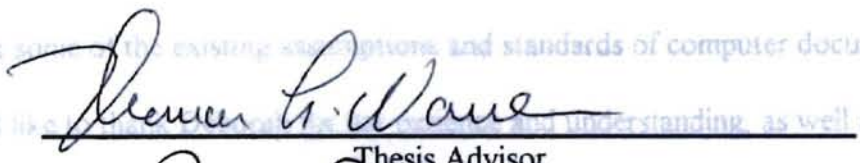
St. Louis, Missouri

1990

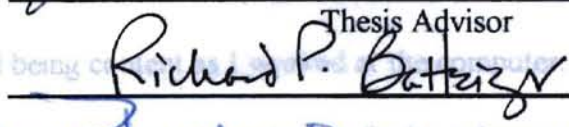
Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF ARTS
December, 1996

PROVIDING APPROPRIATE INFORMATION: AN ANALYSIS OF PROCEDURES IN COMPUTER USER DOCUMENTATION

Thesis Approved:



Thesis Advisor



Dean of the Graduate College

PREFACE

I developed this study in an effort to analyze some of the more obvious discrepancies I have observed between the literature and practice in the field of technical communication. As a practitioner and author of computer software documentation similar to the samples studied in this thesis, I believe that much research in technical communication remains transfixed by old paradigms of documentation. I hope this thesis may, in some small way, provoke a renewed analysis of computer documentation and a fresh look at some of the existing assumptions and standards of computer documentation.

I would like to thank Deborah for her patience and understanding, as well as Ashleigh for smiling and being content as I worked at the computer. Additionally, I sincerely want to thank Dr. Thomas Warren, my advisor, for his tolerance of my multiple and sometimes conflicting projects and directions. Thanks also to the remainder of my thesis committee — Dr. Robert Brown and Dr. Richard Batteiger — for their insightful and extremely helpful comments on the draft of this thesis. Thanks also to the librarians at the Oklahoma State University library and others, I am sure, whom I have forgotten at this time.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Writing instructions for computer users continues to challenge technical communicators, as it apparently has for several years, based on the steady flow of new publications about computer documentation. The ever-increasing complexity of computer software, coupled with the near-impossibility of targeting with focus and precision the needs of a heterogeneous audience, leads to continued discussion and recommendations of the best or most usable or highest quality documentation; yet the literature yields few new solutions for the technical communication community.

Much research and publication in the technical communication field continues to focus, as it has for the past several years, on how to provide "effective documentation" — whatever that might be — to computer users, including both novice users and others who need to use computers as a tool to accomplish specific tasks. Books, anthologies, periodicals, and articles about writing procedures, user's manuals, and usable documentation abound. As a technical communicator, facing the onslaught of too many how-tos about how-tos, the challenge becomes not only to provide information effectively to the audience, but to make sense out of the advice and recommendations from the technical communication literature.

In this chapter, I will outline my observations about commercially available computer documentation, in the context of frequent recommendations from the technical communication literature. I will then identify and describe a research problem stemming

from these observations. I will conclude this introductory chapter by identifying key aspects of the problem to consider throughout the subsequent review of literature, methodology, results and discussion, and conclusion.

Observations and Commentary on Computer Documentation

Two main schools of thought exist in the technical communication literature through apparent attempts to provide information to readers more effectively. As I will show more thoroughly in Chapter 2, the approaches to writing instructions in the technical communication literature can be divided grossly into the school of minimalism (John Carroll, others) and the more conservative or traditional school of providing clear and complete yet unadorned documentation sets (Carliner, Redish, others).

However, commercial computer user's manuals seem to completely fit neither of the above categories. For the purposes of this thesis, I consider commercial computer user's manuals to include most commonly-available, task-oriented, reference books designed to help people use software. For example, many books in the . . . *for Dummies* series, in the . . . *for Busy People* series, or other similar or competing books would fit this definition of commercial computer user's manuals, while tutorials or guided learning activities would not.

Far from minimalism, many commercial user's manuals include thorough instructions on all aspects of the computer software — more information, in some cases, than is contained in the manufacturer-provided documentation. Much commercial documentation also does not fit with a more traditional or conservative paradigm of technical communication. The periodical technical communication literature as well as many recent

technical communication books explicitly instruct writers to "provide only the information that is required, no more, no less" (Carliner 1993, 181). Rather than following this directive, commercial computer documentation often includes asides, commentary, advice, and other information embedded within sets of procedures.

While "the information that is required" in procedural sets certainly lends itself to loose interpretation, the overall sense of the professional literature about writing instructional text indicates that superfluous information would hinder, not help, readers. As a matter of fact, many technical communication researchers explicitly indicate that "the reader's goal is to get in, get the answer, and get out as quickly as possible" (Redish, Battison, Gold 1985, 139) with the admonition that non-essential information will hinder the reader.

However, rather than providing an organization with clearly defined and limited sets of procedures for the reader to access easily, many commercially available examples of computer documentation present an amorphous mixture of instructions, hints, background, and introductory information. The non-procedural information within the instructional steps I will call *elaboration*. Elaboration is additional, non-essential and non-critical information, buried in instructional steps.

The observation of elaboration within instructional steps raises the question of how frequently and consistently this elaboration appears in computer documentation. Interestingly, the concept of elaboration is rarely mentioned in the technical communication literature or in technical communication-related instructional text.

Thus, there appears to be a large variety of documentation that fits in neither the minimalist nor more traditional schools of procedural instruction style. Furthermore, this

commercial documentation appears to be fairly unlike many of the computer documentation recommendations from the technical communication literature as well.

Although I do believe that it is important to examine the relationship between recommendations in the literature and samples from the practice, I do not intend to imply that all new solutions should be preferred to the recommendations codified in the literature, nor even to state that any fraction of new solutions should be implemented blindly, without significant usability testing. However, a frequently or consistently observed discrepancy between the literature and the practice is, I believe, counter-intuitive and worthy of closer examination, particularly when the discrepancy might indicate that the current paradigm does not meet certain needs, either of the writer or the reader.

Types of Computer Documentation

In terms of computer documentation, Woolever and Loeb (1994, 2-10) provide a very useful and fairly typical characterization of four types of hard-copy computer documentation: *user guides*, *reference manuals*, *quick reference guides*, and *tutorials*. They identify a *user guide* as "a manual that gives step-by-step instructions for first-time users of a particular product." The user guide, they suggest, is usually task-oriented and increases in complexity through the book — new users should have an understanding and ability to use the product after working through a user guide. *Reference manuals*, on the other hand, assume knowledgeable users (although not expert) and are organized in modular units, often alphabetically by command. User's guides and reference manuals can be combined — either as two manuals in one or by making a user guide easier to use as a reference. *Quick reference guides* are short reminders of how to use the product, while

tutorials — as the name implies — teach how to do something and have some means with which users can evaluate their success.

Woolever and Loeb (1994, 9-10) clarify the difference between user guides and tutorials in terms of the additional features and unique characteristics that tutorials have, including an "informal tone, extended metaphors, pictures, reassurances, and other similar teaching tools." Woolever and Loeb also indicate that tutorials give the sensation that the user is being helped by a patient teacher, which seems to provide support to novice or unsure users. All these characteristics would seem ideal for computer user documentation, but Woolever and Loeb warn that tutorials do not work as well after users have mastered the product because users find the pace too slow.

These classifications, although fairly typical of the technical communication literature, do not seem to accommodate a significant amount of the commercial computer user's manuals available today. Much commercial documentation — characterized by the publishers as *user's guides* — blends traditional reference material with a vaguely tutorial tone and emphasis, yielding books that are not user's guides or tutorials in the traditional sense and are not designed strictly for reference. These new user's guides are appropriate for both novice and experienced users, and are distinctly different from much computer documentation as described in of the technical communication literature, as well as different from Woolever and Loeb's definition of user guides.

Selected samples of computer user documentation that blend reference documentation, user's guides, and some of Woolever and Loeb's characteristics of tutorials serve as the focus of my inquiry in this thesis.

Research Problem

I suspect, based on these preliminary observations, that a discrepancy exists between recommendations in the technical communication literature and the practice of technical communication, at least insofar as commercial computer user's guides reflect the practice. Commercial computer user's guides now exhibit — in a development that is apparently relatively independent from the technical communication literature — a composite of tutorial, user's guide, and reference manual. This composite comes complete with an apparently unusual, at least in the context of traditional technical communication, approach to writing instructional steps.

Although a discussion of the relationship between the literature and practice — if the literature prescribes the practice or describes what is already being done — exceeds the scope of this thesis, I believe that any observations from the practice that appear to be not addressed in or to be different from the literature merit further investigation.

An important step of investigation will be to determine if commercial documentation does or does not actually follow the recommendations from the research literature. Therefore, in this thesis, I will address the research question: Do computer user instructions from commercial documentation consistently follow the guidelines presented in the technical communication literature? If, in fact, the literature and practice do show substantive differences, I will explore possible reasons for the differences and call for additional inquiry into this aspect of technical communication.

Specifically, in this thesis, I will examine only sets of procedural instructions that give step-by-step instructions in finite processes, taken from commercial computer documentation, to determine if a basis exists to assert the need for further study of user

of the essence of David Dobrin's definition of technical communication: "Technical writing is writing that accommodates technology to the user." (Dobrin 1983, 242).

In arguing for providing only essential information, Carliner (1993, 163) specifically warns against including, for example, information about how a computer system processes information. On the other hand, in a more general context, Dobrin maintains that technical writers, when presenting information, ask first "who is the reader" not "what is the world" (247). Carliner appears to follow what Dobrin characterizes as the "what is the world" view, with the expressed concern about providing such esoteric information, rather than following a more practical "who is the reader" view. Certainly, for most audiences, Carliner's example reflects addressing a "what is the world" view, because few readers have need to know or use information about how computer systems process information. For writers who keep Dobrin's definition in mind, the additional information that Carliner, among others, is concerned about including becomes superfluous, thus paving the way to including useful, effective information within procedural steps, as a service to the reader.

Terminology and Issues

Elaboration

For the purposes of this document, *elaboration* refers to information included within instructional steps or within a procedure that provides information that is a) not essential to the completion of the procedure by the intended audience and is b) not a cautionary or warning statement. Clearly, "not essential" cannot be precisely defined and will vary based on the audience, but most instances of elaboration can clearly be identified as useful in a global context, although expendable on a procedure by procedure basis.

For example, in a document intended for highly experienced electricians about replacing a fuse box, one might find only the most basic steps, as shown in Figure 1. The basic procedure includes only steps, with no additional guidance for readers. In some documents, for their intended audience, that information might well be all that is necessary (presuming that all warnings and cautions have already been adequately addressed). The same document for less experienced audiences might include elaboration information and be more generally useful, as in Figure 2.

Replacing a Fuse Box

1. Shut off power to fuse box.
2. Cut all wires.
3. Remove the screws that connect the fuse box to the wall.

Figure 1: Basic Procedure.

As shown in Figure 2, the elaborated instructions address the same material, but with additional help and assistance, such as an experienced professional would provide. The information between Step 1 and Step 2 and that between Step 2 and Step 3 are possibly essential information for novice audiences because completely inexperienced readers might not be able to complete the procedure safely without that information. However, a slightly more experienced audience would quite possibly benefit from — but not really need — the tips. A truly experienced audience would likely not need the information at all.

Replacing a Fuse Box

1. Shut off power to fuse box.

   Break the circuit between the house main and the fuse box by throwing a switch, removing other fuses, or disconnecting the electric meter where the power enters the house.

2. Cut all wires.

   Cut cleanly and evenly, as close to the old connections as is feasible and perpendicular to the wire, to make it easier to strip and reattach the wires later.

3. Remove the screws that connect the fuse box to the wall.

   Save the screws for reuse later.

Figure 2: Procedure with Elaboration.

I consider elaboration information as information that purposefully stretches the bounds of audience appropriateness (in the strictest definition) with the objective of making the procedure more useful, effective, appropriate for learning, or otherwise more effective than it would have been without elaboration. Specifically, elaboration augments procedural steps with information that often facilitates understanding or learning, or with other information that is not essential to the reader for completing the procedure.

Elaboration expands on essential information, facilitating increased domain knowledge, without delving into extensive theoretical discussions and without requiring extensive prior knowledge to understand. The complexity of computer systems makes it impossible, within a manageable scope, to address all possibilities and contingencies of a system or even of a procedure within a system. The reasons for this complexity and the eventual results will be demonstrated throughout this discussion, but in short, suffice it to say that most computer processes and procedures and *the context within which they occur* cannot be adequately described with simple, unelaborated, procedural steps.

Sometimes elaboration information is already well-known or even considered common knowledge. In other cases, the elaboration provides insight or other assistance that only the author knows, or, in some cases, that the author feels will be appropriate additional information or a useful reminder for that reader at that time. What constitutes elaboration is certainly subjective, and what is elaboration in one context for one audience might well be essential information in the same procedure for a different audience.

While it would be easy to dismiss instructional text with "elaboration" or other apparently superfluous material as an anomaly or as an insignificant deviation from the accepted technical communication practices (outlined in Chapter 2), my preliminary observations indicate that elaboration text frequently or consistently appears in commercial (after-market) documentation — documentation for which users pay substantial amounts of money and documentation that they can review and consciously choose. Not to imply that people necessarily purchase the best or more useful documentation, but instances of readers consistently choosing documentation that deviates from recommendations in the literature would be a finding that should require further research and a closer look at the assumptions in the literature.

In this thesis, I will attempt to determine if elaboration actually occurs frequently in computer user documentation — too frequently to be only an anomaly. If it does frequently occur, that will indicate a discrepancy between the technical communication practice and the literature, which would then probably merit further inquiry.

Forecasting Information

In Chapter 2, I will review the available literature on computer user instructions and related topics that apply to the study of computer user instructions. Appropriately developing computer user documentation requires knowledge of several subject domains, including aspects of learning theory, instructional design, and more traditional areas of technical communication, including procedure writing and audience analysis. In an effort to adequately survey relevant areas of scholarship, this review of literature in writing instructional text draws primarily on technical communication literature, but writing procedural steps unfortunately remains fairly lightly studied; therefore aspects of other disciplines must be used as a tool with which to analyze how procedures can and should be written.

Chapter 3 will describe the methodology I use to assess sets of procedural steps from selected commercially available software documentation. Specifically focusing on commercial documentation for Microsoft Word for Windows, I will assess the frequency and kinds of elaboration used in the documentation. In Chapter 4, I will present my findings about how procedural steps in commercial documentation are written, then, in Chapter 5, I will discuss possible reasons for my findings, including elaborating possible causes for the disparity I expect to find. Finally, I will conclude with a discussion of my findings and their significance for technical communicators as well as for students and teachers of technical communication.

Chapter 2

REVIEW OF LITERATURE

Overview

To answer the question, "Do computer user instructions from commercial documentation consistently follow the guidelines presented in the technical communication literature?" requires not only a review of literature about writing procedures, but also a survey of more global concepts to provide the correct context for the discussions, including concepts such as instructional design and audience analysis.

My premise is that much computer user documentation incorporates elements of instructional design and of more traditional technical communication, while completely fitting the mold of neither. This section will outline some of the relevant aspects of the instructional, technical communication, and other professional literature that affect computer user documentation. Additionally, this section will establish a framework of concepts within which I can discuss my findings about procedural steps in software documentation.

The various areas of scholarship in instructional text and in creating procedural text address creating basic procedures and developing straightforward training assignments, but fail to address adequately the synthesis of procedural steps and instructional design that appears to characterize much commercial computer documentation today. Specifically, computer user documentation appears to require both procedural information to serve reference needs and information to help readers understand more global

considerations than a series of steps. The published literature comprehensively addresses, on the one hand, the instructional design considerations involved in teaching about computer programs or systems, and on the other hand, adequately instructs on the process of creating effective procedural steps. The literature does not adequately synthesize the two different areas.

Surveying the published scholarship that affects procedural sets in computer documentation requires a survey of learning theory from education and psychology, in addition to audience analysis and writing procedural instructions from technical communication.

Throughout this thesis, it is important to note that most of the published scholarship in this area is anecdotal in nature. All empirical studies that I review will be explicitly called out as such in the text.

In this section, I will first review relevant literature about audience analysis, particularly as it directly relates to selecting information for procedural steps. Beyond the traditional areas of audience, this section will also review some more anecdotal aspects—particularly the "users do not read" premise. Next, I will review selected relevant literature about learning theory, including the aspects of behaviourism, cognition, and information processing concepts. These areas all directly affect issues of structuring information for learning or doing or both. The final section of this review of literature addresses the various theories about writing procedures, including reasons people read the written procedures. I begin with the minimalist school and proceed through the various levels of gray that accompany the "provide only essential information" edicts.

Audience

The concept of audience and audience analysis in technical communication is nothing new, but a review of some important aspects and issues will provide useful background and contextual information for further discussion of computer user documentation. I will begin the discussion with Pearsall, who provides a groundwork for the theory, then will proceed through several different aspects of audience analysis and definition.

These approaches develop an assumption — quite possibly an inappropriate assumption — that only specific, quantifiable, information is necessary in procedural steps. Further discussion of audience analysis delves more into the abstract and less clear issues of audience, from issues of psychological approaches to rhetorical aspects of audience. Pearsall's 5-part definition (1969) of audience, including the layman (sic), executive, expert, technician, and operator, provides a useful view into audience to determine the information that should or should not be included in instructional text. Instructional or procedural text occurs primarily in documents written for Pearsall's technician and operator audiences.

Pearsall indicates that for an audience of technicians the writer may assume a high-level of practical knowledge in the field and therefore should provide "lists of instructions written in the active voice, imperative mood" (1969, xx). Technical manuals need not be self-contained and may assume certain knowledge on the part of the reader, for example, in not instructing on the use of tools in the context of a procedure (1969, 41).

For an operator, who may have somewhat less knowledge in the subject domain than a technician — but no less need for thorough instructions — manuals should be self-contained, should provide background information where essential, and should include

complete procedural steps. Pearsall does not explicitly address elaboration or providing tips for the operator, although he does acknowledge that providing information about selecting tools for a task is sometimes necessary (1969, 47).

Modern computer user documentation presumably addresses an audience similar to Pearsall's operator, with the caveat that they might have little or no subject domain knowledge, yet need to use the tools nonetheless. However, other approaches to audience take the issue to another level of complexity.

In the context of instructions in the use of a product, the information provided in a set of instructions is based on "an assumption that the audience has been through all the earlier stages and is now ready to use the product to its fullest potential. This information solves the problem of how a particular application will work" (Caernarven-Smith 1983, 19). Theoretically, then, the writer must provide only the instructions and no other information, if the audience has already done everything else.

In terms of audience analysis as it is addressed in the literature and taught in advanced technical communication classes intended for prospective professional writers, readers' need for additional information, such as support, confirmation, and ongoing explanation is occasionally raised as an issue. For example, Alred, Oliu, and Brusaw recommend providing "reassurance at the outset" but fail to recommend ongoing reassurance, even for the anxious audience. (1992, 60).

Caernarven-Smith (1983, 195) addresses similar issues when she notes that reluctant audiences need to be

> coaxed and cared for to the point that they feel well educated and confident to undertake the job. This is accomplished by presenting a good deal of information, including many illustrations.

Additionally, Caernarven-Smith comments that authors should provide pacing information suggesting what to read first and should establish levels of prerequisite knowledge for specific topics. Both of these techniques serve to slow the eager audience as well as support the reluctant audience. It appears that this perspective, although reasonable, is de-emphasized by a large part of the technical communication literature.

Caernarven-Smith (1983, 141-3) also reiterates the importance of communication psychology in effectively addressing an audience. Grossly paraphrased, she sees the significant issue as how people respond to the overall communicative approach, not just the words on the page, including the level of language, of sophistication, and the sense of pace and style.

Further addressing technical communication from a psychological perspective can be used to continue to support a case for elaboration in user documentation. Sanders (1988, 65) builds on Carl Roger's argument and discusses the importance of the writer expressing a perspective as it would be seen by the reader.

A three-part synopsis of Rogerian argument involves 1) stating the facts neutrally, 2) demonstrating an understanding of the situation from the reader's perspective, and 3) stating aims and goals, including value judgments (Sanders 1988, 67). Although Sanders provides an explanation of how these stages map onto conventional expository text, they also map — at a micro level — easily onto an instructional step, with adjunct information about what the system does as well as an instance of elaboration.

Opening a File

1. Choose File → Open.

   The Open dialog box will appear. Don't worry about choosing the file type — the program guesses, generally accurately, for you.

Figure 3: Procedure in Rogerian Terms.

Figure 3 shows a procedure with Rogers' neutral statement (step 1), statement of understanding (first sentence after the step), and further elaboration including goals and aims. (Lest we assume that Sander's intent was only to apply the Rogerian argument to longer discourse objects, an example [69] applies it to a clever twelve line poem, with very good results.)

Developing rhetorical aspects of audience analysis further, technical writers should "acknowledge that their writing is . . . rhetorically engaged and that they can expressly and consciously practice the strategies that make their writing communicate" (De Beaugrande 1988, 84). While specific mention of instructional steps is not present, it seems that effective communication from the reader's perspective would more closely resemble Sander's Rogerian interpretation than mundane procedural sets.

Additionally, Sanders counsels technical writers against "trying to stay out of the way of the objective truth" and implies that a direct statement of evaluation or opinion, at rhetorically appropriate times, is to be recommended (1988, 74).

Returning to the topic of the Rogerian concepts of inferring the reader's needs and directly addressing them, some studies have analyzed the effects of anticipating questions. Odell (1985, 255-259) cites comments from a study about elaboration in text in which analysts (the writers, in this case) anticipated questions from their hypothetical readers and addressed them explicitly in the text. Odell's assessment, based on the relative lack of

contact the analysts had with their actual audience and the intermediate reviews of the analyst's writing, implies that perhaps the sense of audience is misplaced because the analysts did not really know (personally) their audience. The same procedure that Odell critically relates — that of imputing the response of a hypothetical audience — is generally recommended to technical writers of all sorts creating a variety of different documents. In general, I would suggest that any attempts to anticipate and answer questions from the audience should be encouraged, not discouraged.

Overall, the psychological approaches to audience and understanding the reader seem to be addressed at the theoretical level, but not often applied to procedural steps, at least in the technical communication literature. Were they consistently applied to the literature, elaboration would probably be more frequently recommended.

Asking Questions

In an article about the Five W's, Hart (1996, 139) tangentially addresses the need for clarification and elaboration in technical documentation. Hart identifies five considerations for technical communicators in the context of the journalistic maxim of *who*, *what*, *why*, *where*, *when* (and *how*). Hart defines *what* as the task users might try to do and the steps they must follow to do it (140), and *why* as the explanation of why anyone would choose to follow that (or those) steps. In further clarification, Hart states that *why* information in a manual would include explanatory components, apologies, and justifications — potentially all issues that would fall under the more inclusive rubric of elaboration within procedural sets (144).

Additionally, using the questions of *what*, *how*, *when*, *where*, *who*, and *why* can help determine the appropriate level of detail in a procedure (Zimmerman and Campbell 1988, 49). As with Hart's five question assessment, the *why* corresponds most closely to elaborative information. Zimmerman and Campbell state, however, that writers should "resist the temptation" of answering all possible *why* questions in a procedure and reserve more detailed *why* answers for training documentation (1988, 50). It is not clear if this warning not to include too much information refers to a reading to do approach or to something similar to Redish's "reading to learn to do" concept (1993).

In a final assessment, Hart claims that much technical documentation addresses the reader's need to know *what* but "either ignores or undervalues the remaining four W's." (1996, 145). Hart further points out that failure to provide information to answer all these questions leaves meeting the reader's needs at least somewhat to chance.

Certainly, anything that can be done to answer questions for the reader is to be recommended. Meeting audience needs, of course, assumes that the audience actually reads the material provided — an assumption that many researchers call into question, as well as the what and how of reading instructions in general.

Reading, or Not

A commonly expressed opinion in much of the technical communication literature is that users do not read documentation or at least do not thoroughly read documentation. This section will survey different schools of thought about how and when users actually read documentation.

Two kinds of reading tasks apply to this analysis of elaboration in user documentation. Reference reading is reading to find an answer, while performance reading is to carry out a set of procedures (Holland, Charrow, and Wright 1988, 34). Assuming readers read documentation specifically to complete a task, most readers of user documentation fall into the performance reading category. However, taken in a broader perspective, both tasks can easily apply to user documentation. In the context of computer user documentation, the immediate need is logically to carry out a set of procedures, while finding other information is secondary, yet important for many purposes.

As Wright (1994, 17) asserts, "the reluctance of people to actually read technical materials is well known." She further asks — rhetorically — "if the quality solution is to show [meaning tell] people or let them find out." Letting them find out would be most similar to a minimalist perspective and Wright's own minimalist recommendation, while the alternative solution she rejects fits a more traditional model.

Some studies show that few readers will read continuous discourse — apparently users want to spend their time working on the computer, not reading about it. The users "scan down the page looking for specific steps that they can follow" (Ramey 1988, 148)

In an attempt to address the issue of readers reading or not, Weiss recommends that the best approach is to "devise documents that compel readers to find what they need, in the most efficient sequence, and with a level of effort that neither discourages them nor lowers their productivity" (1991, 16). Although Weiss stops short of recommending specific steps to take, information embedded within series of instructional steps would logically be a starting point with which to control the reader's experience.

Brockmann asserts that increasing the reader's sense of interactivity, either through minimalist design or online documentation, will improve the effectiveness of computer documentation. In an effort to encourage users to read the documentation, he recommends using conversational style and avoiding humor (1990, 214-215).

A possible origin of the presumption that users do not read documentation comes from Redish, who cites a study (Sullivan and Flower, 1986, 170-171) showing that reading documentation, in a particular task-oriented situation, was sporadic at best. Readers apparently refer to the documentation only when they are looking for specific information or when seeking the answer to a specific problem. Redish correctly summarizes that these findings are not universally applicable because some people tend to read more often or willingly than others (1993, 20).

However, many other sources in the literature note the apparent reluctance of readers to read. Brockmann (1990, 31) states that adult learners resist reading by trying to use the system without reading the documentation, or that they guess about what should or should not be happening with the system as soon as they start learning it. Additionally, "readers of user manuals only come to a manual when they have a problem . . . " (Brockmann 1990, 53). Carroll reiterates the renowned user reluctance to read training material carefully, preferring to skip around to find the answers to specific issues (1990, 8). Chisholm (1988, 318) also asserts that documentation is rarely read and, when it is read, that reading normally occurs in certain limited situations, particularly with inexperienced users, unique products, in cases of concern about damaging the product, and if the product is perceived as dangerous.

However, the very act of referring to the documentation indicates that the readers have some confusion or uncertainty. Similarly, if they read the documentation, they expect to be able to follow the directions and to succeed at completing the procedure. If for some reason they cannot succeed, their expectation of success in following a procedure was violated and they seek an explanation or clarification. If writers provide only basic procedural information, without error correction information, the reader may or may not find the clarification or information to produce a favorable communication outcome.

The additional information necessary to help readers correct their errors or to help them understand possible reasons for problems, at the points at which the problems arise, could be examples of elaboration. Perhaps the issue of elaboration can be considered a case of multiple audiences — one that wants just the basic instructional steps, and the other, more of a learning to do audience that seeks additional information or that encounters problems with the procedure.

Although content probably causes most problems with users' failure to read instructions, other reasons that users might not read instructions may exist. Cohen compares reasons for selecting online or printed documentation in the context of software documentation. She refers to Weiss and others who contend that "only wierdos read technical manuals" (1991, 127) and itemizes (anecdotal) reasons for choosing alternative (online) documentation, including the allegations that technical users hate to leave the screen, that novices hate to read too, that poor readers are also less successful at finding information, and that the static structure of printed literature is less accommodating than it should be. The quest to find the reasons that users do not (apparently) read documentation touches on audience reading skills, a desire to maintain focus on the computer screen, and

limitations of printed documentation. However, although Cohen concludes that online documentation has particular advantages, she does not address issues of content and writing that could eliminate much of her perceived problem with traditional documentation.

Cohen considers reasons for selecting alternative media to more effectively address the audience. She does not address the possibility that the well-known and often discussed problems with software documentation result from a shifting paradigm of software documentation, rather than from problems inherent in either online or paper documentation, nor does she raise the possibility that most "technical manuals" do not meet user's needs because of the information presentation, rather than because of faults with the medium. She states that

> many computer users must either enjoy reading or believe that they need to read in order to keep up with their systems. Nothing else can explain the proliferation of the many magazines, newsletters, reference books, and instruction books that are so clearly aimed at a computer-using audience." (128)

With that, I would point out that the readers of much computer documentation perhaps strive to find documentation that meets their needs as computer users. Specifically, however, exactly what characterizes effective documentation remains open to question. Certainly computer user documentation can only succeed if the actual procedures - the raison d'être for the documentation — are effective. However, beyond this aspect, it could be that another characteristic of effective documentation for users would be that the documentation helps the readers easily gain additional information beyond just the fundamental buttons to push.

Learning Theory Background

To provide a starting point for discussion, I will begin with an assumption that one of the fundamental reasons for which procedural text is created is to control actions, and, among other purposes, computer user documentation is procedural text. Procedures, from any source including computer user documentation, assembly instructions, or how-to guides for any purpose, exist only as a means with which the actions of the reader can be controlled or directed to accomplishing a specific task.

In computer user documentation, one significant consideration should be retention of certain material. For example, while assembly instructions or annual maintenance instructions need not be retained, specific information about procedures and practices in computer software used frequently should be retained — users want to remember how to accomplish a procedure without having to look everything up each time they need to do something.

Based on those two assumptions, I will discuss some elements of learning theories in the context of computer user documentation.

Two primary schools of thought exist regarding what brings about what behavior. The *behaviorist school* (dating to the post-War period) uses a programmed learning approach in which tasks are reduced to the smallest possible steps and users are trained — in the strictest interpretation — to follow the steps, just as rats are trained to follow a maze. The *cognitive school*, currently preeminent in the pedagogic literature, addresses holistic concerns and regards the human mind as an information processing system, rather than as an organism to be trained (Seels and Glasgow 1983, 34-5).

Behaviorism

The behaviorist school, developed out of B. F. Skinner's psychological experiments, provides one of the key distinctions in terminology that applies throughout this discussion. *Training* will be used to describe learning in a strictly behavioristic sense — that is, observable performance of behavior. *Learning*, in this thesis, applies to competence in a domain, to use Chomsky's term. More broadly stated, *learning* is acquisition of knowledge. People who acquire knowledge can deduce specific procedural steps from the larger body of domain knowledge, while the possibility of inducing the larger body of knowledge from trained steps is questionable at best.

Although the behaviorist school is nearly 50 years old, many of the fundamental assumptions of the school remain evident in both instructional design and technical communication literature today. In particular, the assumption that users of instructional text need strictly procedural instructions (i.e., instructions yielding observable behavior) exists throughout the current corpus of technical communication literature. As I will show later in this section, much of the technical communication literature indicates that procedural steps should contain nothing but instructions (Carliner 1993, Horn 1980, Carroll 1990, and others) — that is, should focus on definable or measurable steps in a larger process.

By stating that procedural steps should contain no other information, the technical communication literature indicates that the steps themselves contain all the information that a reader needs. Many in technical communication know this as the "reading to do" approach, in which we as writers assume that readers will read only when necessary to

complete a procedure or to find specific information (Redish, Battison, and Gold 1985, Wieringa, Moore, and Barnes 1993, Zimmerman and Campbell 1988, others).

Although behaviorism might seem outmoded and somewhat archaic, most instructional objectives, even in the newest occurrences, are measured in terms of Magar objectives — strictly behaviorist and observable (Seels and Glasgow 1983, 35).

Instructional designers and learning theorists over time rebelled against the assumptions inherent in behaviorism. The theories developed in response to weaknesses in behaviorism are cognitive and focus on acquisition of competence, rather than measurable performance as the behaviorist school does.

Cognition

While at a superficial level the behaviorist assumption seems valid, the philosophy begins to break down in very complex systems or in cases in which documenting all possible cases is not possible. Although the behavioral view appears to claim that anything can be trained, the theory breaks down at higher levels of complexity and more complicated levels of thought. For example, the production of new and unique utterances by language learners precludes the possibility that they learned only by practicing.

As Chomsky states, in the context of children's language acquisition, "it is a mistake to assume that — past the very earliest stages — much . . . is acquired by imitation" (Chomsky 1971, 132). This concept applies equally well to acquisition of competence in use of computer software, as many of the procedures and sequences of procedures which a user performs have not been explicitly trained in that sequence.

In this section, I will review basic concepts of cognition to provide a context and perspective to the preceding discussion of behaviorism. While behaviorism would suffice as a model for simple steps — as in assembling a kit or following a map — more involved tasks require a more comprehensive and complex model to represent the process taken to evoke actions in the reader.

Bloom's Taxonomy of Educational Objectives, ranging from the first (lowest) level of knowledge to Comprehension, Application, Analysis, Synthesis, and Evaluation, provides a useful starting point for discussion (Bloom 1956, 201-207, Seels and Glasgow 1983, 26-27). While the specific levels are open to discussion and interpretation and many other, more developed taxonomies exist, the general idea of rote recall and basic manipulation of ideas constituting the lowest range of the taxonomy is commonly accepted. This lowest level is the more easily trained range, yielding observable behavior, and corresponds closely to behaviorist theory. Higher levels of the taxonomy require more complex thought processes than the recall of the lowest levels, and consequently also require higher levels of effort on the part of the learner and, when applicable, the teacher.

When writers create computer documentation — particularly reference materials — we generally address the lowest levels of these hierarchies. Reference material or strict procedural steps would correspond to the Knowledge and possibly, depending on the context, Comprehension levels. Tutorials and thorough training programs on computer software might address the Application or Analysis levels, but certainly no higher. As a rule, the technical communication literature specifies this type of behaviorist approach (Horn 1980 , Carroll 1990, and Carliner 1993).

Behaviorism per se is not an incorrect premise for technical communication, particularly for straightforward systems. However, when reason exists to believe that users will be trying to learn how to use software in a relatively global sense, rather than read how to accomplish a particular procedure, behaviorism may well be an oversimplification and more complex models may be required to represent the process accurately.

For example, a user who must perform a one-time setup task requires probably no more than the most perfunctory of instructions, containing only essential information. No retention or recall is likely necessary — the user completes the steps, the task, and likely never returns to the procedure. A behaviorist, observable, model would be ideal for this situation.

However, a user customizing the setup for the first of many times, or attempting to make changes to the configuration of a program in an effort to improve efficiency, would benefit from more information designed to help make connections, answer questions, and address potential problems or pitfalls. Useful additional information within the procedural set will help the user more easily perform the same task later. In this case, the writer must strive to provide the necessary information for observable results — performance — but also include information to improve the non-observable results — the competence.

In cases of documentation for a system that readers will be using extensively and creatively — as in a word processing program, spreadsheet, presentation software package, or almost any other program with more than a narrow and clearly defined application — gaining knowledge or learning is likely to be at least one of the readers' objectives, even if the reader might not articulate the objective in those terms.

Romiszowski, a cognitivist, in the context of performance and instructional systems, discusses four categories of knowledge (1981, 242-3) — facts, procedures (both grouped together as factual information), concepts, and principles (the latter two also grouped together as conceptual information). These categories offer a different way to approach technical communication and instructional steps.

Factual information (244) loosely corresponds to the levels of performance traditionally expected from much technical communication. For example, facts, fact systems, and procedures all fit into Romiszowski's factual information category. (1981) Therefore, steps from instructional text such as what to type, press, click, or the overall process involved in a mail merge would fit in the factual information category. Facts and procedures — factual information — include the information that would be trainable in a behavioral sense and lends itself to straightforward procedures. Conceptual information is a higher level of thought that requires cognitive manipulation of factual information.

The second category — requiring higher levels of thought and an additional level of abstraction (247) — might be addressed through explanations, exposition, or higher level training. Behaviorism cannot adequately explain the acquisition of skills in Romiszowski's conceptual information category because these skills are not observable, but this category would be necessary for users to extrapolate from procedures effectively and to begin to understand the software, rather than only remembering how to accomplish a specific task.

In other words, users must be able to construct a Plan, to use Miller, Galanter, and Pribram's term (1960, 16). The plan is a "hierarchical process" that includes multiple individual operations to accomplish a goal. For readers to synthesize individual procedural sets and construct a plan, the documentation must help them increase their competence

using the software. Miller, Galanter, and Pribram refer to the basis of information that allows individuals to develop a Plan as the Image (18). The role of computer user documentation, in this sense, is to add the necessary information to the Image to facilitate planning.

In further explanation of cognitive means of acquiring skills, Romiszowski explains a four-stage cycle of skills: Learners will 1) perceive a problem, 2) recall the prerequisites to solve the problems, 3) if necessary plan an approach to resolve the problem, and 4) perform that necessary acts to resolve the problem (1981, 254). In cases of problems of larger scope, this process might be iterative. This process is certainly more involved and complex than behaviorism can model, and necessarily involves instruction addressing higher level thought processes to follow the process.

The second and third steps in this cycle — recall and planning — are unique to cognitive theory. Behaviorism assumes perception and performance, without necessary intermediary steps.

In the context of the fundamental levels of user's manuals and user documentation, Romiszowski's (1981, 253) skills schema applies largely in the cognitive realm and primarily not in the psychomotor, reactive, or interactive realms. (Obviously, the actions of moving and clicking the mouse as well as typing on a computer keyboard fall into the psychomotor category, but the vast majority of user's guides do not explicitly instruct in the mechanics of pressing keys or clicking a mouse button — those skills, if not the specific applications, are taken as given capabilities for using the software. Sometimes users are reminded of the difference between clicking and double-clicking, but the physical press and release action need not be taught in most documentation.)

I believe that elaboration can assist with both the recall and planning stages of Romiszowski's skills cycle by providing hooks into existing schema in the readers' minds.

Both schools — behaviorist and cognitive — have their uses and advocates, but also each exhibits significant weaknesses. The behaviorist philosophy does not address mental processes or more complex levels of human performance, particularly those performance characteristics that are observable but have not been explicitly trained. The cognitive school discusses learning and gaining knowledge, but fails to address the adult learner's desire for specific instructions and procedures, without extraneous information.

Behaviorism can be summarized as evaluating observable behavior, while cognition is concerned with acquisition of competence in a subject. The following sections of this review of literature will refer to these fundamental concepts embodied by behaviorism and cognition to build further on these concepts.

This — relatively brief — review of behaviorism and cognition should suffice to show that behaviorism alone cannot account for all aspects of effective computer documentation. While a behaviorist model could represent the execution of step by step procedures, it cannot model the larger issues of acquired competence and knowledge required for any level of understanding of the software. For systems requiring a deeper understanding than rote recall of specific procedures, a model of learning more complex than strict behaviorism must be employed. In addition to modeling means of presenting information, specific models of information acquisition are necessary to see how readers must perceive and integrate information.

Processing Information — Schema

The preceding discussion of the two main schools of thought regarding learning and performance leads to the process by which users assimilate information or instructions. The cognitive process by which people learn new material is to fit new information into an existing paradigm or concept — otherwise known as schema. People can more easily learn new material when it fits into the corpus of knowledge they already possess. The use of metaphor is an example of using schema to help readers learn.

In the context of computer documentation, Carroll (1990, 38) notes that users refer to their prior knowledge and assume consistency in system behavior. Carroll (36) correctly identifies some user behavior as attempts "to construct their own ideas about what" was happening in the program. Users attempt to make connections throughout their activities at the computer, based on assumptions of consistent behavior on the computer's part. While the assumption of consistency is logical, it is also not necessarily adequate in the context of computer software. In cases in which acquired competence is important, writers must provide additional information or hooks to help users make their connections.

Jonassen (1985, 8) reminds us that a network of concepts is one possible representation of memory — new ideas must be related to the existing schema to be effectively used and readers' existing schema will determine to some extent how text may be interpreted. He suggests that activating schema in the reader can lead to better understanding and more effective learning (10). Jonassen explains that reader's knowledge "provides a framework within which new information can be integrated . . . (1985, 10)"

In the same vein, technical communicators should actively provide scenarios to help readers apply theoretical material to their own situations (Brockmann 1990, 216).

Brockmann goes on to state that readers often construct their own scenarios anyway, and providing the context makes it easier for readers to assimilate the new information.

In a related discussion of schema, Redish (1993) addresses Carroll and Rosson's 1987 "assimilation paradox." Readers apparently "actively interpret" as they work through software programs. The readers seek connections to their existing schema and concepts — connections that the writer should facilitate. Redish refers to the difficulty that readers have in moving beyond these connections to learn more about the program, and implies that the paradox indicates that making connections is not always to be recommended. Redish refers to Fisher and the need to determine precisely how or when to interrupt the reader with new information (1993, 30).

People constantly attempt to relate the information they are receiving to their existing knowledge and the overall context (Carroll 1990, 74). Or, in other words, they are "too busy learning to make much use of the instruction" Carroll (1990, 75). Carroll continues to paraphrase Anderson, Bartlett, and Piaget: "Having goals and coordinating prior knowledge with new experience is necessary for meaningful learning, indeed for meaningful experience at all."

According to this view, providing instructions alone is probably not adequate — readers require additional information to help them construct mental models.

Ramey (1988, 147) also strongly supports the concept of schema and mental models as aids to processing information. She indicates that users "frequently need for the documentation to provide the link between what they want to do and what the computer can do." She notes that documentation can exploit existing knowledge by helping to

organize the capabilities of the computer into the tasks, terminology, and reasoning skills of the user.

Readers should have a problem-solving schema in place when completing task instructions because following instructions without a schema appears to be more confusing and difficult for readers (Mulcahy 1989, RT-119). Mulcahy empirically assessed the effectiveness of instructional text with an explicitly stated goal, with "task rules," with both a goal and rules, and a control text with neither goal or rules. The rules correspond most closely to elaboration — they include explanatory text about the outcome of specific actions. Mulcahy found that the "most comprehensible, clear, coherent task instructions" provide both goal and causal information (RT-121). Both "local and global coherence" seem to be important to overall success in completing task instructions.

Elaboration theory, which states that content relationships should be explicitly addressed, also appears in instructional design literature. While the overall instructional design approach is not completely appropriate for this application, one of the central concepts is that "structural relationships in content should be explicitly taught and tested" (Sari and Reigeluth 1982, 60). Applying this concept to computer user documentation would require authors to make explicit connections to other aspects of the software. By making connections on a specific type of content relationship — e.g., conceptual, procedural, or theoretical — writers can help readers develop their cognitive structures in a specific field, which would in turn help with retention and retrieval of information according to schema theory. Although elaboration theory does not indicate that these connections should be made within procedural steps, it does, however, favor the inclusion of explicit connections at some point.

Certainly, if developing structural relationships in the material being presented will help readers learn more effectively, we as technical communicators should attempt to do so, through providing any necessary information to the readers. While developing structural relationships is not necessary for one time processes, repeated processes or sub-parts of a larger concept to be learned — like a computer program — should be taught in any way possible.

Jonassen (1984, 255) discusses theories of advance organizers in text and includes as an assumption that new information is meaningful only insofar as it can be attached to relevant existing concepts or schema. This assumption, while perfectly valid for organizers, could apply as well to information within the body of the text or procedural steps.

Further, organizers prevent the learner from having to make all the connections — from having to realize or discover how the information fits into a larger context (Jonassen 1984, 257). Presumably any information that meets that goal would be useful or effective. Jonassen additionally reports conclusions from research, including in particular that "organizers need not necessarily be presented in advance of instruction in order to be effective. Post-organizers have also facilitated learning" (263).

In support of schema and organizers, Jonassen (1985, 10-11) explains:

> The generative model asserts that learners, when faced with stimuli, construct and assign meaning to that information based upon prior learning. That is, existing knowledge structures are activated for the purpose of interpreting that incoming stimuli, which are in turn encoded as distinctive features of memory that may be later accessed to explain new information . . . . The more connections that we make, the more meaningful the text becomes.

Some of the theories that support generative processing as in the above example include (Jonassen 1985, 16-19) Ausubel's Theory of Cognitive Psychology, emphasizing organizers; Levels of Processing theories, and finally theories involving schema. Each of these separate but related theories supports the general concepts behind generative processing.

One significant point is that "if text content can be directly related to prior knowledge, it seems logical to presume that the interaction would be more productive" (Jonassen 1985, 32).

For this reason, it is important to control the communication process, or more precisely, to control for "the sources of noise and error" (Weiss 1991, 16). This reference to noise, as in the Shannon and Weaver communication model, refers to any stimuli that could interfere with effective communication. Weiss logically establishes a perspective on documentation in which extraneous or unnecessary stimuli must be excluded — or specific material must be linked with an existing schema — to communicate effectively. The idea of controlling both readers' actions — as in procedural sets — and controlling for errors is an interesting approach that calls for making connections explicit, to eliminate guesswork and unnecessary conjecture from the users' experience.

With that, it is clear that we, as technical communicators, must provide hooks to connect the new information we present to existing information that our readers already have. This concept of schema and the requirements for advanced cognitive models of learning theory seem to indicate that step by step instructions — behaviorism alone — cannot suffice to develop computer user documentation that really serves the reader.

An attempt to apply instructional design theories to procedural steps quickly shows that the appropriate emphasis on Redish's "reading to learn to do" (1993) is the reading to do part — the remaining "to learn" aspect must of necessity be of secondary importance. Instructional design theorists from Rogers to Piaget to Gagne emphasize the acquisition of domain knowledge rather than structuring a procedure for easy and effective completion. (West, Farmer, and Wolff 1991, Romiszowski 1981) While the "to learn" aspect of Redish's approach (1993) remains a significant consideration, of primary — but not solitary — importance for technical communicators must remain the steps in the procedures themselves.

Writing Procedures

Certainly, if computer documentation effectively and easily met the needs of most users, discussion of how best to prepare computer documentation would be superfluous at best. However, that is certainly not the case. This section reviews a broad spectrum of sources about writing procedures in an effort to determine the overall sense of the literature to writing procedural sets.

Overall, two schools of thought clearly emerge — the minimalist school, advocating providing only the most essential information, and what I will call the generalist school, advocating providing whatever information is necessary. However, even the generalist school seems to prefer to err on the side of providing less, rather than more, information, in an effort to reduce confusion and help the reader as much as possible. Much of the discussion of procedural sets in the literature seems to result from attempts to address the failings of existing documentation, as described below.

Oram (1992, 61) rationalistically and succinctly analyzes the failings of much computer documentation. He points out that current manuals are preoccupied with "getting users started" at the cost of not preparing the users for real situations and applications. Specifically, he asserts that a "popular style" of software documentation is to reduce everything to step-by-step procedures, possibly with background information included in other parts of the manual. He correctly identifies a significant problem as severing "the link users need between theoretical and applied knowledge" (61).

Oram further states (63) that

> *Step-by step procedures are useful in places, but they can easily hide as much as they reveal. The user can start out with a false sense of security in running the procedure and end up not knowing where to turn when something goes wrong.*
>
> An all-too-popular writing approach entails boiling information down to small chunks, in basic categories such as introductory background, procedures, and reference material. The evaporated by-products are worth more than the residue that remains. During later writing and review stages, the writer can totally miss the important large-scale issues, such as how the user should integrate a product into other activities.

The documentation Oram (1992) describes above is not only all-too-popular, but also all-too-familiar to many technical communicators. However, the "popular writing approach" that Oram castigates is unfortunately one born of many articles and books in the field of technical communication.

It is important to note that many slightly older approaches to documentation, particularly in the field of computer software, seem archaic in the context of current thought and in the context of the latest computer technology. For example, in the context of developing computer documentation in 1988, Holtz (23) identifies specific needs for

description, installation, operation, maintenance, and training documentation. These boundaries either blur or vanish in the context of modern software programs.

However, Carroll (1990, 71) accurately notes that, despite improvements in user interfaces and technology, users still are

> overwhelmed by complexity, misled by their expectations and prior knowledge, thwarted in trying to make sense, frustrated and unsuccessful with rote instructions, unable to coordinate multiple sources of information activity, inconsistently successful at reasoning on their own, and generally susceptible to problems with error recognition and recovery.

Additionally, Chisholm (1988, 308) effectively explains part of the difficulty in creating computer user documentation:

> Not only are computers complex, they are by far the most complex machines ever produced for widespread use . . . . The users of these increasingly complex machines are more varied than ever; more and more non-technical persons are using them. This is the first time in history that such complex machinery, with such complex capabilities, has been put into the hands of such varied users.

The traditional approaches to creating procedural text correspond closely to the behaviorist school. These approaches are behavioristic in the sense that they address the steps to be taken by the reader deliberately without providing information within the procedure about what, why or how the step is being taken. Technical communication literature tends to caution strongly that strictly step-by-step procedures (trainable) with no additional information, are preferable to any mixture of procedure and other information. While this particular approach is, in general, not incorrect, a behaviorist approach to computer user documentation is inadequate for the complexity of most computer systems today.

The Minimalist School

Starting with the rigorously minimalist approach, much of the technical communication literature speaks strongly to providing pure procedural steps, with only essential information.

John Carroll led the push toward minimalist documentation. He assessed the problems with computer documentation and identified minimal documentation as the ideal solution. I completely agree with his assessment of the problems in computer documentation — they persist today — but believe that other avenues, such as the computer user documentation discussed in this thesis, will more effectively address the problems. That said, a review of Carroll's statement of the problem and the context in which he developed the minimalist approach will provide a useful introduction to the minimalist school.

Carroll identifies the emergence of "user self instruction" in the 1980s, as computers became widely available and used as tools, rather than as objects of inquiry (1990, 4). The first examples of this class relied on drill and practice to train (in a behavioral sense) on using the software. However, Carroll accurately observed that behavioristic objectives failed to meet real user needs and were often considered obstacles or ends in and of themselves (7). In short, the objectives were tasks out of context, rather than paths to accomplishing a real goal. A significant goal of documentation should be making it easy for users to apply the knowledge that they already have to real tasks.

Carroll comments on a specific study that "participants' lack of knowledge about word processing can make it hard for them to interpret accurately what happens" (25). Based on the number of supplemental commercial software manuals on the market today, this lack of knowledge must continue to frustrate and confuse users.

Carroll (40) also notes the difficulty that some users have in identifying when they have made a mistake or committed an error, particularly if the consequences are not immediately obvious. Additionally, he identifies the lack of error recovery information as a problem in many manuals (9).

Explicit and implicit links between instructional materials and the system can help users perform better (Carroll 1990, 86). Carroll uses the example of a figure verifying that the user successfully completed the step as a case of effectively making this link. This link could also be a case of increasing competence on the part of the reader by explicitly making the mental connections.

In asserting that minimalist documentation is a possible solution for these expressed problems, Carroll asserts that the goal is to provide "less overt training structure." (78). Carroll also rejects comprehensive introductions to manuals or sections as obstacles to user's desire to get started and their need to accomplish something, and thus espouses minimalism as the answer (80-81). Carroll's answer cannot be immediately accepted as valid, however, because of a fundamental flaw in the methodology. Part of Carroll's study that claimed to show the effectiveness of minimal documentation re-labeled procedures with questions or gerund constructions that clarified the means of accomplishing recognizable goals. While this approach likely did improve the documentation, these and other developments would tend to make the manual more friendly and usable than the system manual in any case, minimal or not. Therefore, the findings of effectiveness for minimal documentation must be regarded with great caution, as the empirical evidence is flawed.

Finally, Carroll notes that forcing users to improvise helped them learn more quickly (1980, 180). Unfortunately, users of reference or user documentation do not want to improvise — they want to do.

"Because technical documents are tools that people use to do their jobs, they want to get in, get the information, and get out of the document," cautions Redish in Barnum and Carliner (1993, 19). Redish, Battison, and Gold (1985, 139) also remind us that "the reader's goal is to get in, get the answer, and get out as quickly as possible."

On the providing as little information as possible side of the coin, Robert Horn's Information Mapping™ principles clearly state that only essential information should be provided in any procedure (1980, 4-2 — 4-4, *passim*).

Similarly, Carliner (1993, 160-161) explains procedures as a set of instructions for performing a task, and explains specifically how to develop procedures. Additionally, Carliner (1993, 162) states that only "must-know" terminology or concepts should be presented in a procedure. The non-example Carliner uses, however, explains information that writers should not include with an example of how a computer might process an instruction, rather than possibly more immediately applicable information. Not including background information is a more clearly seen and easily made choice than not including an explicit comparison to related material.

The ambiguity introduced by Carliner's requirement that only "must-know" information should be included raises the question of what, exactly, must-know information is. While this will obviously depend on the circumstances, writer, and reader, Carliner makes it clear through further explanation and example that strictly procedural

information belongs in procedures and other information of all sorts should be removed to another section of the document.

Carliner summarizes (1993, 181) that technical communicators should "present only the information that people need — no more, no less." He refers to the fact that readers are "inundated" with information and it is essential to include only the information that readers need.

Similarly, in discussion of evaluating instructional text, Hartley cautions that instructional text must contain content that is "accurate, unbiased, up-to-date, and sufficient for the purpose at hand" (1985, 139). While that statement could easily be read as support for providing any additional information that readers need, Hartley immediately follows with the statement that readers sometimes complain of "too much information" — more than is actually necessary — in technical documents. (1985, 139).

While it is entirely possible that readers are on occasion provided with too much information, such as background information about how a system works, it is hard to imagine that elaboration or directed information about the process or expectations a reader should have while performing a specific task could contribute to a sensation of providing too much information.

We should note here that these examples, while quite clear and directive in their admonition about providing extraneous information, do not specifically either include or exclude computer user documentation from discussion; they do not consider computer user documentation separately from all other instructional text.

In the context of nuclear power plant procedures, Wieringa, Moore, and Barnes (1993, 57) discuss procedure writing and the need to write "at an appropriate level of

detail." They note that too little detail can prevent users from completing a procedure, while including excessive information (58) can also be problematic. Particularly given the possibility that procedure writers would try to be "safe" and include extensive explanation and detail, they caution against the danger of including too much and confusing the reader.

Additionally, procedure writers must assume a certain base level of knowledge on the part of users, from which way to turn a bolt to loosen it to how to communicate with others in the organization (Wieringa, Moore, and Barnes 1993, 61). Wieringa, Moore, and Barnes do acknowledge that non-emergency procedures can reasonably include more detail than emergency procedures, and note that less-frequently used procedures can reasonably include yet more information to help the reader.

These instructions provide useful guidelines for technical communicators in general, but do not necessarily speak directly to the issues in computer user documentation — questions of accuracy and speed must necessarily be handled differently in nuclear power plants than in word processor documentation.

In the context of general visual information presentation, Mackh and Rew (1991) include the benefits of using lists for sequential instructions. They do not address, either pro or con, the issue of interrupting the list with additional information that could, plausibly, not go in the list.

Wieringa, Moore, and Barnes (1993, 174-175) state that providing multiple equivalent steps from which the reader can choose should be minimized in procedures. They recommend presenting alternatives only when one of the choices might not be available or if the alternative would also be useful. If alternatives are presented, then information to help choose among them should also be available.

Similarly, Wieringa, Moore, and Barnes recommend not putting diagnostic information in procedural steps; rather diagnostics should be pulled out and into a flowchart. Only minimal diagnostic information should be included in procedures (1993, 176).

In a slightly different approach, Sebrechts, Deck, and Black (1983) address "mapping" as an non-traditional and possibly effective means of providing information to the novice computer user. They quantify the goals of a traditional instruction manual as telling a user how to accomplish each task by providing specific instructions. They state (1983, 200) that users want, at this point, a "cookbook" approach "without adornment" and allege that "extended text describing relations within the system can frequently lead to frustration." They further state that, for many, the more effective way of presenting procedures is to present only the steps, without commentary.

The Generalist School

Frequent recommendations in the technical communication literature call for providing all the information necessary for the readers to perform their tasks, with admonitions of varying degrees about the perils of providing too much additional information. This section surveys these generalist approaches to writing procedural sets.

Computers or other complex systems differ from other objects of documentation because the relationships between and among system components are important, and novice users frequently construct inaccurate mental models to represent the relationships (Sebrechts, Deck, and Black 1983, 200-201). In order to eliminate the problems with mental models, users need extra information about "the overall design of the computer

system, including a description of relations among the various commands and menus." Adding this type of additional information conflicts directly with the goal of "eliminating text in order to highlight procedures."

Additionally, Houp, Pearsall, and Tebeaux state that technical communicators, when writing instructions, should "include everything that is really relevant and nothing that is not relevant." They indicate that including theory, when necessary, is acceptable, but significant amounts of theory should be separate from the instructions. If "brief explanations of theory" would help the reader understand a "few steps" in the instructions, that it is acceptable to place those explanations with the steps (1995, 504).

In specific discussion of how-to procedures, Houp, Pearsall, and Tebeaux do state that including detailed instructions or theory, as needed and in moderation, can be acceptable (1995, 525). They do not address the need or desire for providing status check information (e.g., if you see certain information, you are on the right track) or other meta-discourse about the procedures, beyond the implication of omission that this information is not necessary.

Houp, Pearsall, and Tebeaux do discuss the acceptability of including "tips" either as a separate section or as part of the actual instructional step, if they "provide helpful tips on how to do a better job or that provide guidance when trouble occurs" (1995, 525).

Holtz, however, correctly identifies the need to provide "enough" information — not too little, not too much (1988, 56). A common error is apparently a result of not understanding the reader and the reader's need for information (Holtz 1988, 100), but it is left as an exercise in judgment how to determine the appropriate level of detail.

Holtz (1988, 106) draws a useful distinction between education — acquiring general knowledge — and training — acquiring specific skills. He casts, however, documentation firmly in the fields of training, in a strictly behavioristic and observable sense, complete with behavioral objectives, and does not regard documentation in the sense of facilitating acquisition of competence in a field.

Tutorials — step by step training — differ significantly from reference material, which provides generic procedures, including "exceptions, warnings, asides, and extra data" (Price 1984, 53). Writers should not mix the two, because of the danger of confusing or delaying the readers (54). Tutorials provide a "gradual progression" while reference material exists to allow people to retrieve information quickly(55). Certainly, as computer technology has changed since the 1984 date of publication, some of the considerations have changed too, particularly as the distinction between the two types of information has blurred.

With reference to tasks or procedural sets, Price asserts that writers should "put all the information needed to complete each task — and nothing irrelevant — into the reference sections" (1984, 57). Exactly what constitutes "irrelevant" is not defined. Additionally, Price notes that reference material must include "everything that a user could ever want to do with the software" (1984, 111). Certainly, with the expanding capabilities of most commercial programs, many people do not even know what they could do with the software.

In the context of writing technical, but not computer, documentation, Haydon notes only that operational documentation might include detailed instructions about performing certain procedures, with no guidance as to the format, organization, or structure of those

instructions (1995, 129). In the same context, Sides counsels the importance of erring on the side of providing too much information rather than too little (1991, 98). Additionally, Sides, stresses the importance of providing a "short purpose statement introduction" and indicates that the introduction is the most often overlooked aspect of writing procedures (99). Sides emphasizes that writers must inform the readers not only of *what* to do, but also of *why* to do it. However, as with many other published works on the subject, the emphasis on *why* lies in the introductory sections, not within procedural steps themselves.

The issue of presenting only the information that readers need is a murky one. While the technical communication literature often argues or implies that complete step-by-step instructions, in the simplest form, are all the information readers require, based on preliminary observation of the additional information included in many procedural steps, it appears that other information is often included.

Certainly, the weight of technical communication literature strongly emphasizes providing only essential information in procedural steps. Even in cases in which explanations, tips, warnings, or cautions must be included, the tenor of the literature is to minimize, minimize, minimize.

Holland, Charrow, and Wright (1988, 38-40) recommend, in the context of accommodating information to multiple audiences, compartmentalizing information to allow specific audiences to easily identify information (out of the main flow of text) that pertains to them, while the main flow of text contains the heart of the document content, appropriate for all audiences. While Holland, Charrow, and Wright recommend many different means of compartmentalizing information, they do not recommend any approach with extra text integrated into the procedural steps — all the compartmentalized

information is moved aside into an appendix, introduction, or, in some cases, into a sidebar.

Also in the context of providing additional information, Zimmerman and Campbell (1988, 114) explicitly address means of using "informational statements" appropriately. They state that informational statements many be necessary to supplement the procedural steps, but remind us that a procedure is not the same as a training manual and that it should include "little explanatory material." They recommend placing brief information about the procedure in the initial sections of the procedure, rather than in the body (steps).

Additionally, "beginning users may better perform a procedure that gives explanatory notes, while a more experienced user may skip over such information" (114). However, information as Cautions, Warnings, or Notes should be typographically or graphically differentiated (114-120).

Zimmerman and Campbell (1988, 36-7) also address user expectations — as established by headings and subheadings within the text. As long as the "standard" format is consistently followed and the labels accurately reflect the content to follow, the procedure may be successful. Additionally, Zimmerman and Campbell remind us that if a user has to ask *How*, the writer has not presented enough detail (1988, 38). However, they quickly follow this with a warning about the perils of presenting too much information — they assert that readers will skip items that contain too much detail. Unfortunately, they provide no means of gauging how much information is too much.

Caernarven-Smith (1983, 182) notes that audiences with little technical expertise pose additional problems for the technical communicator. She notes that the temptation is to

respond to lesser technical knowledge with increased explanation, but recommends instead short anthropomorphic explanations.

In a more global context, Warren (1993, 87) recommends selecting "an organization that supports the reader's needs, allowing immediate and unimpeded access to the specific information necessary." This organization could easily include elaboration, if necessary, to support the reader, although Warren does not explicitly address this issue.

Lazonder (1994, 95) demonstrates in an empirical study that minimal manuals with error information (defined as extra information to help readers identify and correct errors) demonstrate that the "conditions exist for the error information to have a facilitative effect on learning." While another study failed to conclude that the error information would help readers improve their demonstrable skills after practice (92), apparently readers do use error information provided.

In a discussion of documentation goals, Ummelen (1994, 117) explains the difference between the goals of procedural and declarative information in the context of the writer's rhetorical goals:

> "A procedural goal would be to instruct the reader, that is, to give specific directions for performing a task in a certain way. A declarative goal would be to inform the reader about certain facts, to transfer knowledge about facts to the reader. "

I maintain that computer user documentation attempt to or should attempt to meet both objectives — procedural first, then declarative as possible.

In a discussion of reader needs, Steehouder (1984, 134-5) identifies four situations, each with unique information requirements (italics all mine): *impasse*, in which procedural information is needed, *error*, in which a diagnosis and remedy are necessary, *dis-*

coordination in which the reader "may seek information that helps to provide an overall picture of the program," and finally *uncertainty*, in which the user needs confirmation of specific assumptions. I maintain that providing procedural information alone meets only the impasse needs and perhaps the error needs. While many documents attempt to address dis-coordination and uncertainty through overviews or other declarative forms of text, some documents might include information to meet those needs within procedural steps, with very good results.

Manuals are written "with the intention that the full possibilities of use are made clear to the users" (Mårdsjö 1994, 190). This comment does remind us of a key problem — users require information about the capabilities of the system or software or object. In response to users' demonstrated reluctance to read about the system, minimal manuals attempt to present just that information that is essential. However, a significant aspect of the instructional text is omitted when a purely minimal approach is taken. Effective computer user documentation might require more elaboration, as in the commercial user documentation evaluated in this study.

Van der Meij (1994, 203) identifies different sub-types of action information found in manuals as action information that is "meant to support hands-on experience," error information to help "detect, diagnose, and correct" mistakes; linkage information to help the reader make connections between text and screen, and finally action prompts, which are tips or shortcuts. Additionally van der Meij categorizes other information as declarative or text information. Everything except the action information could be considered elaboration in this context.

Reading to Learn to Do

Mirel, Feinberg, and Allmendinger (1991, 76) distinguish between the older model of "reading to do" and current models of "active learners." They identify three arguments against reading to do: First, "reading to do suffices only when the problem domains are unambiguous, goals are well-designed, and tasks require routine rather than complex, open-ended problem solving." It appears that reading-to-do does not describe GUI computer programs. Second, "reading-to-do learning often results in rote performance, and mechanical, albeit operational, instructions can even obstruct users from formulating the production rules they need to tackle unique, undocumented tasks." Finally, they identify the indication that users prefer to experiment or "play" with a program, particularly as they gain skill with the software. They close that "for active users, learning and performing are interdependent and dynamic"(77).

The "standard approach is to present background details and how-the-system-works information separately from and before step-by-step procedural directions." This view corresponds with a skill learning model (Mirel, Feinberg, and Allmendinger 1991, 77).

When designing documents for active users, Mirel, Feinberg, and Allmendinger (1991, 78) identify four areas of investigation, including content organizational issues, principles of design to help users "conceptualize the relation between the structure of a program and their tasks," elaboration on procedural steps to help users generalize, and troubleshooting help." Mirel, Feinberg, and Allmendinger identify Charney et al as the primary advocate of elaborated instructions and call for more research into how to create documentation to facilitate complex or higher order problem solving tasks (80-82).

Redish (1993, 20) suggests that some documents, including computer users' guides, serve a purpose of "reading to learn to do." Readers who "read to learn to do" may be characterized by the fact that they want to get on with their work, however, they also need to or want to learn as they do.

Other than implying that these combination purpose documents should include both numbered lists and similar task oriented information and structures to help recall, such as summaries, Redish offers no particular suggestions for addressing these needs (1993, 21). Specifically, in this context, Redish makes no recommendation, either pro or con, about including elaboration or additional material within procedural steps.

Fawcett, Ferdinand, and Rockley (1993, 52) emphasize the "reading-to-learn-to-do" concept and the reader's need for "conceptual information about the program as well as procedural information to help perform the tasks." Beyond that, they state that readers need "information that will help them transfer what they learn . . . to real tasks." However, they claim that examples and simulations are the key to accomplishing this objective.

On the other hand, a reader's desire to refer to documentation for specific procedures or information and then to exit the document as quickly as possible raises an additional argument in favor of providing elaboration information. As Warren (1993, 87) reminds us, readers of technical documents of all kinds do not read the documents linearly. He discusses the need to structure information accordingly, as in providing effective page layout and design to facilitate ease of entering and exiting the pages. By taking Warren's argument to the level of instructional text, providing information to ease the entry into and exit from procedures, through elaboration-type information, could also prove very useful to readers of instructional text.

Additionally, the writer must defer, at times, to the greater knowledge of the reader with the particular situation that the reader has. While it would be ideal to offer only a single way to do anything, a rule of thumb to follow is that the "writer should provide the reader with options only when the reader has information that informs decisions (or judgments) that the writer cannot obtain." The ideal choice is to present the best way, second is to present alternatives along with objective criteria to make a decision and third best would be to present choices and some information about how to make an informed judgment on the topic (Wieringa, Moore, and Barnes 1993, 190). This information to facilitate judgment could well be elaboration.

Computer documentation can be defined as "communication designed to ease interactions between computer software and the individuals who manage, audit, operate, or maintain it" (Brockmann 1990, 12). He identifies specific purposes of computer documentation as improving efficiency, overcoming users' fears, and selling the product. Elaboration logically fits in both the first and second categories. In the first, Brockmann states that "people need to understand the systems with which they are working." He recommends providing the assistance necessary to promote a "successful first encounter" with the software. Here too, although elaboration is not explicitly addressed, assistance could take that form.

Brockmann identifies only two types of computer user documentation: reference and tutorial (1990, 14-16). Reference material is an encyclopedic technical tome, as we would expect. However, his definition of tutorial is broader than most. He defines tutorials as material that "selects from the comprehensive reference material and presents information in a step-by-step fashion. It is usually organized around user tasks or around a hierarchy of

user needs." In this sense, Brockmann's definition of tutorial corresponds closely with the definition of user's guide used by commercial documentation publishers and with the definition I use in this thesis. Brockmann cautions that tutorials limit how users think they can use software (1990, 16).

An interesting dichotomy appears in the literature between books about procedure and technical information writing, which seem to be oriented to a variety of topics excepting computer procedures, and all other works with apparently generic titles but focused primarily on computer-related issues. While the literature rarely explicitly addresses elaboration in procedural sets, many of the recommendations made speak to issues that elaboration can help address.

Conspicuously absent from much of the technical communication literature is the question of organization, content, and structure at the sub-chapter and super-sentence level. These procedural design issues, while not addressed in most literature, are crucial in providing useful instructions to our readers.

Advocating Additional Information

One empirical study of users of computer documentation reading to learn a skill does address elaboration within procedures. Charney, Reder, and Wells identify the "expounder" school, in which adherents think that instruction must be as thorough as possible, and the minimalist school, which should be focused on facilitating the learners' creativity and exploration (1988, 47-48). They hypothesize that elaboration is not only useful for learning facts, but is also useful when learning skills.

The Charney, Reder, and Wells (1988) study appears to exist in a vacuum — it certainly does not appear to be propagated extensively through the technical communication literature. Among their findings, they observed that "learners were more greatly impeded by the 'under-elaborated' texts than . . . by the 'over-elaborated' one." (52), which they interpret as support for the expounder school.

Charney, Reder, and Wells (1988, 53) also note the deficiencies in Carroll's study, in which terminology and verbiage were changed, thereby clouding the results of a comparison of minimalism and traditional documentation.

In an empirical study of different types of elaboration that could be included in text, designed to determine if elaboration might address the concepts and procedures needed, when it might be used, and how it might be used, the samples elaborated on all possible permutations of conceptual and procedural information and found that elaboration on procedural aspects improved performance, while elaboration on conceptual aspects revealed little effect (Charney, Reder, and Wells 1988, 54-55).

Significant benefits can apparently be derived from elaborating on how to apply procedures, although the benefits of conceptual elaboration are non-existent, as are the benefits from advising on when to apply specific procedures (63-65). Additionally, readers following specific tasks need little elaboration, while readers without clear goals do benefit from elaboration. All users — novice and experienced alike — benefited equally from elaboration on how to apply procedures.

Separately, Ramey also indicates that users want strategies that "help them integrate the conceptual and keystroke-level information efficiently" (1988, 151). They also want

the manual to anticipate the errors they are likely to make and provide information to prevent or recover from the error.

Ramey (1988, 152) recommends that manuals include information in the user's terminology, to avoid confusion about the reasons for following a particular action. Additionally (153), she states that readers do not want multiple ways to do a single procedure — they want the basic approach up front and want to be able to refer to alternate ways of carrying out the same procedure.

Ramey distinguishes between a tutorial — designed to teach — and a procedure — focused on helping the reader do an action correctly "without worrying *primarily* [italics mine] about creating a lasting perception of the overall system workings" (1988, 153).

In an empirical study, LeFevre and Dixon address the use of examples in instructional text. While not quite the same as explanations or elaboration, the results about using examples are interesting. They demonstrate that people effectively use examples as a supplement to instructional text and, given a conflict between instructional text and examples, favor the example. More importantly, the users studied prefer examples, if asked to choose (1986, 29). LeFevre and Dixon conclude that their results *discourage* frequent examples in instructional text because, given examples, readers tend to disregard the text. Perhaps if the instructional text is particularly important, they do, but if completing a task — by whatever means — is the critical issue, good examples should be used frequently.

Smith and Goodman conducted a study that specifically addressed the use of explanations in instructional text. They compared the results of groups performing a procedure using linear instructions, using structural explanations (focused on the act) and

using functional instructions (focused on the object). In brief, they found that instructions with explanations to establish a schema were read faster, executed equally well, and with better recall and transfer than in the linear group (1984, 384-386).

Conclusion

Prescriptive comments about not including extra information in procedural steps appear to be designed at least partially to eliminate problems and compensate for errors of judgment on the part of the writer, rather than exclusively to help the reader. Several aspects of cognition, audience, and writing procedural steps call for the use of elaboration as an aid to the readers. Interestingly, it seems that the researchers who acknowledge the acceptability of including extra information within procedures are generally those who have explicitly — generally empirically — studied or assessed writing procedures.

Chapter 3

METHODOLOGY

Overview

The review of literature about developing procedural text indicates that the trend of scholarship in technical communication recommends providing instructional steps unadorned with other information or with only little additional information. As discussed above, some researchers recommend selective use of additional information, although primarily with a cautionary note about the perils of overusing additional information. Additionally, many defer to audience considerations, but again with the admonition to keep non-essential information out of the procedural steps.

However, many recent commercially available user's guides appear to exhibit consistent and plentiful additional information within procedural steps, contrary to recommendations from the technical communication literature. I will examine a selection of user's guides to discover how they reflect the recommendations for procedural text in the technical communication literature.

In this section, I will first present the process with which I restrict the scope of my inquiry, as the domain of all commercially available computer software documentation is not a manageable scope for study. The documents for evaluation were selected based on topic, genre (user's guide), intended audience, and availability, according to the criteria outlined below. Following that, I will explain the means with which I select and evaluate specific sets of procedural steps.

Academic Basis for Methodology

Assessing procedural sets in computer user documentation poses methodological challenges because the documentation of a single product can vary dramatically, according to many different factors — style and audience, among others — that cannot be effectively controlled. Therefore, I have chosen to use a qualitative approach to this problem. I will draw on several sources from published literature about qualitative research and develop my methodology from these sources.

In general, qualitative research in technical communication identifies three major kinds of qualitative research designs: ethnographic, descriptive, and case study (Morgan 1988, 27). Ethnographic research addresses people and their relationship with their environment, while descriptive studies attempt, on a large scale, to describe events or phenomena. Case study research falls in the middle — researchers are "interested in identifying features of a particular phenomenon by examining, in depth, few (perhaps only one) example of that phenomenon" (27).

It is just such a case study I will undertake, to assess the structure and application of procedural sets in computer user documentation.

Morgan (1988, 28-29) also relates several methods of collecting information, ranging from documentation, archival records, interviews and surveys, direct observation, participant observation, and physical artifacts. Physical artifacts include user instructions or documentation — precisely the type of data I will analyze.

I will use a qualitative process, including content analysis, to collect information. Content analysis, as defined by Marshall and Rossman (1989, 98), is "a technique for

making inferences by objectively and systematically identifying specified characteristics of messages" to produce countable results.

After I collect the data, I will use basic descriptive statistics to compare the use of procedural sets and any additional information embedded within those sets.

My qualitative analysis of procedural sets within computer use documentation fits well in Bereiter and Scardamalia's (1983: 3-5) six levels of inquiry in the writing or composing process. They state that their levels are of varying levels of abstraction, but are all equally valid for analysis of text at different levels. I will draw on the first two levels: Reflective Inquiry — thinking about and analyzing observations — and Empirical Variable Testing — quantitatively assessing characteristics to support observations. These two levels can also be used to analyze the products of the writer, rather than just the writing process. Beyond the second level, the observations are no longer verifiable and tend to the more abstract analysis or speculation about writer's intentions and cognitive processes, which are both out of the scope of this inquiry. I will use only the first two levels as a basis for my analysis of computer documentation.

Level 1 (5-6) requires no empirical research — rather, it demands cognitive inquiry into existing or commonly available information. Level 2, according to Bereiter and Scardamalia (6-10), supplements Level 1 inquiry by providing a sound and somewhat quantifiable — although not absolute — basis for Level 1 analysis.

Bereiter and Scardamalia (8) caution that, while Level 2 inquiry certainly is not in and of itself a basis for abandoning common sense and experience, a "reasonable person will not ignore Level 2 results. At the very least, Level 2 findings that run against common knowledge should serve as signals that something is more complex than we had assumed."

Bereiter and Scardamalia (9) comment that a good Level 2 study will "establish its findings within a context that has some representative significance." While there exist many disadvantages of Level 2 inquiry, particularly the lack of control of myriad independent variables, Level 2 inquiry can provide a more sound basis with which to supplement Level 1 reflection, as Bereiter and Scardamalia note (7).

My objective in this section, to recast it in terms of Bereiter and Scardamalia's assessment of Level 2 inquiry, is to attempt to identify characteristics of computer user documentation that run counter to recommendations from the technical communication literature. Although I do not maintain that deviation from the literature necessarily produces better or more usable documents, significant differences between practice and literature would call for some level of usability study of the documents created and used in practice.

Analysis Process

This section will specify the process with which I will analyze procedural sets in computer user documentation, including how I will identify procedural sets and what I will consider in determining the content of procedural sets. In the following section, I will identify the books I will analyze.

What is a Procedural Set?

The units within which I will be comparing instructions are procedural sets. A procedural set falls under a (generally action-oriented) heading in the books, follows any introductory material, begins with actual instructional steps, and concludes with the last information that is typographically part of the set. That is, a new heading signifies the end

of a procedural set, as does the end of indented text or a semantically concluding sentence, such as "With that, you've finished the procedure."

Typographically or visually distinct information between the beginning and end of the procedural set is not included in the discussion. Sidebars, screened tips and notes, and other distinct information will also not be considered a part of the procedural set.

What Information Do I Seek?

When summarizing the information, I will require three sets of information:

- Steps: procedural steps
- Adjunct information: useful but not absolutely essential information. This information could arguably be considered relevant, within loose interpretations of relevance.
- Elaboration: information that could be removed from the procedural steps without affecting the ability of the reader to complete the procedure in question.

Procedural steps are the actual actions the reader must follow to complete the procedure successfully. Procedural steps include actions, such as *choose*, *select*, *click*, *confirm*, or *accept*. Procedural steps are generally sequentially numbered, but could also occur with multiple steps within one number, or possibly not in a number at all. Likewise, the presence of a step number does not necessarily indicate an actual step.

Additional but not essential information includes cross-references, descriptions of what the reader should be seeing on the screen, or enumeration of information visible on the screen. This information is most likely helpful and reduces frustration, but should not

affect the ability actually to complete the task, assuming the procedural steps are correctly carried out.

Elaboration provides information useful to a global understanding of the program or operating system, information that explains what results a step or procedure will have, humor disconnected from the context or example, and explanations that add information that is not critical (e.g. the fact that something works as you would expect.)

Figure 4 shows a procedure with a step ("Choose File→ Open."), an item of adjunct information ("The Open dialog box will appear."), and an instance of elaboration ("Don't worry about choosing the file type — the program guesses, generally accurately, for you.").

Opening a File

1. Choose File → Open.

   The Open dialog box will appear. Don't worry about choosing the file type — the program guesses, generally accurately, for you.

Figure 4: Action, Adjunct Information, and Elaboration.

To put these categories in terms of conventional assumptions in the technical communication literature, I suggest that a manual written strictly to minimalist conventions would have values approaching zero for both Adjunct information and Elaboration. Additionally, a manual designed to provide only necessary information (albeit at greater levels of detail than a minimal manual) would contain virtually no information classified as Elaboration.

Therefore, my goal in the data collection phase is to identify the number of steps, number of occurrences of adjunct information, and number of instances of elaboration within each procedural set.

However, accurately and quickly determining what is adjunct information or what is elaboration requires smaller units of accurately quantifiable information. Making gross distinctions between adjunct information and elaboration, for example, would lead to the possibility of error, while identifying more readily defined sections of information can proceed without problems. Therefore, to facilitate data collection, I will further subdivide the information categories into the following categories:

- number of numbered steps
- number of actual steps
- alternate actions
- cross-references
- screen descriptions
- system action descriptions
- elaboration

The number of numbered steps is how many numbers for steps are used. Some books will divide a procedural set into very small units, thereby yielding a high number of steps. Others will provide the instructions in paragraph form, with no numbered steps at all.

Actual steps is the number of separate actions the instructions call for. *Click*, *select*, *choose*, and *drag and release* are all examples of actions. Sometimes these are combined within numbered steps, while at other times a numbered step contains no action

instruction, but rather only a description of what the system is doing. Actual steps are tallied individually, regardless of position within a paragraph or procedural step.

I identify actual steps semantically, rather than based on occurrences of specific words. That is, three separate steps, written to be complete, of "1) Click. 2) Hold. 3) Drag." would count as three actual steps, while an instruction stating that users should "click the icon and drag and drop it into the trashcan" would count only as one.

Alternate actions are provided as supplemental or alternative means of accomplishing the same goal. For example, applying bold face to text might instruct readers to click the B button on the toolbar, but also offer an alternate action of pressing Ctrl+B. Cross-references recommend that the reader reference a different section of the book or chapter to find additional or supplemental information about the procedure. Screen descriptions include all descriptions of what readers will see on the screen. For example, if the text states that "you will see the dialog box with three available options," that would be a screen description. Additionally, references to figures and pictures within the text fall into the screen descriptions category. Action descriptions tell what the system is doing, including information like "the system displays the dialog box" or "the system recalculates the total." Finally, elaboration is any additional information within the procedure that fits none of the above descriptions, including humor, comments related to a more global understanding of the software, and explanations.

In summary, information that is an actual action to be taken by the reader will fall into the step category, and completely extraneous to the procedure information will be considered elaboration. The remaining descriptions and alternate actions will yield the adjunct information category.

Presentation of Results

Effectively comparing sets of instructions, even on the same material, can be quite problematic. Different authors will choose to identify the beginning or end of procedural sets differently, or will subdivide lengthy procedures into differing sub-steps. For this reason, literal comparisons across multiple books would be tenuous at best.

However, a discussion of the amount of non-procedural information for each procedural step will be quite useful. By averaging non-procedural and procedural information across multiple sets of procedures, individual variance will be minimized and trends within each book to more or less additional information will be apparent.

By comparing procedural sets within books, I will be able to discern patterns, or lack thereof, in which books have consistent use of additional information within procedural sets. By comparing procedural sets across books, trends in providing more information in certain books or certain procedures will be evident.

Scope of Inquiry

Effectively assessing the structure and application of procedural sets in computer user documentation requires that I first restrict the scope of inquiry to a manageable number of sources — assessing samples from the full gamut of computer documentation obviously is not realistic.

Several criteria were used to progressively narrow the scope of research to a manageable level. To narrow the scope of investigation to a manageable, yet representative, sampling of computer user documentation, I made the following series of decisions.

Target Audience

The broad goal of my investigation is to assess documentation that addresses beginning users, although also addressing other users in addition is acceptable. Books targeting an audience of beginners are the most appropriate for this study because documentation for beginning users begins with roughly the same assumptions, prerequisites, and presumed ability. The needs of advanced users differ from the needs of novice users and are often, but not always, addressed through reference documentation.

Weiss (1991, 14) establishes two categories for instructional documents: orientation and guidance. Orientation documents are intended to "train neophyte users," while guidance documents are directed to a more experienced audience. The documents selected for this study fit approximately in the orientation category.

If procedural steps are actually found to be frequently interspersed with additional information for the reader, I assume that would be most commonly found in books that target beginners because beginners would require more advice and step-by-step assistance than more advanced users.

Furthermore, addressing books that target exclusively more advanced audiences would introduce additional and unnecessary complexity into the topic by blurring the audience's need or desire for additional information. Presumably, advanced audiences — for example Pearsall's technicians, operators, or possibly experts — require only specifically identifiable steps, rather than more broad-based (and appropriate for elaboration) information (Pearsall 1969, xvii-xxii).

Target Genre

I chose to assess user's guides, because they traditionally include numerous numbered procedural steps and attempt to provide information on the program's capabilities (rather than information focused on a specific user task). I excluded tutorials, reference manuals, and other documentation, based on the marketing characterization or book shelving information on the back of the books, for the reasons explained below.

Tutorials differ from user's guides because they have different constraints on their design and use. They are designed explicitly for the people who want to learn about the software, rather than just do a specific procedure. Additionally, because the intent of procedural steps in tutorials is different from a user's guide, existing studies and descriptions of procedural steps in the technical communication literature do not necessarily apply. Tutorials do not need to address the reading to do audience — their approach is inherently different from a user's guide.

Reference manuals were excluded from the study because their design, alphabetic organization, and strictly reference intent targets more advanced users and they generally focus more on specific tasks than on overall program capabilities — knowledge that is assumed of reference manual users.

Online documentation, video or multimedia documentation, and books heavily referencing an included CD-ROM or diskette, do not generally have a structure, organization, or audience that can readily be compared with different books. Specifically, they do not provide a valid object of comparison with user's guides and are not considered in this study. While each of these other forms of documentation would be appropriate for a further discussion of elaboration in documentation, they each represent a relatively

smaller and less heavily studied genre and are therefore less appropriate for this initial survey of elaboration in technical documentation. Additionally, visual guides and quick reference guides also follow unconventional and variable approaches to providing information. Neither category can accurately be represented in this study.

Therefore, I will only evaluate user's guides, to focus the research within a specific genre of technical documentation

Source for Documentation

Next, I determined that the process of assessing procedural steps would be most useful in user documentation intended for a wide audience — if the documentation were designed for a narrow or specialized audience, any findings would be less useful. Similarly, I required documentation for a software program, not an operating system, because operating systems have entirely different constraints and considerations from the applications that most technical writers document and most literature in technical communication considers. Therefore, I decided to seek out documentation for a popular, commercial, widely used, program.

Most people have to write (Anderson 1985 *passim*) at work, and word processing programs are frequently found on home computers as well, making the category of word processing programs a likely candidate for study.

According to IngramMicro (September 1996), different versions of Microsoft Word hold 4 of the top 5 places on the word processing bestseller list, so I selected documentation for the latest version of Microsoft Word. I choose the latest version as it would presumably address a large and varied audience (the potential market) and the

books about Word would presumably incorporate the latest trends and advances in commercial computer user documentation.

Selecting Specific Samples

The next step in developing my study was to select and obtain books. I selected a total of six samples, of which three met all selection criteria and were listed on the Ingram bestseller list, and three were not on the bestseller list, but met all selection criteria and were readily available in many bookstores. Specifically, I used the following process to select the books.

The materials for this study were purchased commercially, from readily available works on Microsoft Word for Windows 95 (sometimes also referred to as Word for Windows version 7.0). I did not assess books that would have been special-ordered or that were not commonly available. The decision to use only commonly available books reflects an assumption that well-known, stocked, and popular computer books reflect, to some degree, the preferences of consumers. While many other factors affect the availability and popularity of computer books, customer preference for specific styles, titles, series, and formats does to a degree dictate the books that are sold.

Of the top 50 word processing books in the week of September 23, 1996 (IngramBook 1996), 22 are for Word Perfect or Word Pro, while 24 are for Word 6.0, are tutorials, or are quick reference guides. Of the remaining 4, one book (Word for Windows for Dummies) is listed twice, so 3 bestsellers remain. Table 1 presents the three best-selling computer books selected for this study.

Table 1: Bestselling Books Meeting Evaluation Criteria

Title	Publisher	Author
Word for Windows 95 for Dummies	Foster City, California: IDG Books Worldwide, Inc.	Gookin, Dan
Running Microsoft Word for Windows 95	Redmond, Washington: Microsoft Press	Borland, Russell
Mastering Word for Windows 95	Alameda, California: Sybex, Inc.	Mansfield, Ron

Additionally, three other books on Microsoft Word were readily available in Barnes & Noble, Media Play, B. Dalton, Waldenbooks, and other general-interest bookstores that fit the specified selection criteria. These three additional books were also chosen for use in this study to provide a broader look at the use of procedural sets. Table 2 presents these three additional works.

Table 2: Related Books Meeting Evaluation Criteria

Title	Publisher	Author
ABCs of Word for Windows 95	Alameda, California: Sybex, Inc.	Hart-Davis, Guy
Word for Windows 95 for Busy People	Berkeley, California: Osborne McGraw-Hill.	Crumlish, Christian.
Microsoft Word for Windows 95 Made Easy	Berkley, California: Osborne McGraw-Hill.	Neibauer, Alan R.

Finally, I will also assess the procedural sets in the Microsoft User's Guide for Word for Windows version 6.0, to allow comparison between commercially available computer user's guides and documentation provided by software vendors. Microsoft did not publish a traditional user's guide for Word 95 that would fit the selection criteria for this study. However, because of the very minor software differences between Word version 6.0 and 7.0, the older user's guide can legitimately be compared to other user's guides for the current version of the software.

Table 3: Microsoft Word Documentation

Title	Publisher
Microsoft Word User's Guide (version 6)	Redmond, Washington: Microsoft

About the Samples

Each of the books selected is a user's guide intended for use with Word for Windows 95. The books selected are categorized by their publishers as addressing a beginner-level audience, although some books target a range from beginner to intermediate users (*Word for Windows 95 for Dummies, Running Word for Windows 95, ABCs of Word for Windows 95, Microsoft Word for Windows 95 Made Easy*) and some claim applicability for all users of Word (*Mastering Word for Windows 95, Word for Windows 95 for Busy People*).

Although each of the books (except for the actual program documentation) was written by an individual named author or authors, the books are all parts of a series of books from the publisher and appear to have been written to a series style guide. Although the author's voice and style is apparent in each of the books, issues such as consistent use of coaching and the general format of instructional steps appear to be common to all books within a series, such as the . . . *for Dummies* series or the . . . *for Busy People* series.

While the fact that the books are based on a corporate style guide does not imply that the findings in this study are universally applicable or generalizable to any degree, it does indicate that, where possible, books in a series will have similar use of procedural steps.

Although it could be argued that differences in audience or audience analysis could explain differences in writing style or structure, by restricting the scope of inquiry to mass market, beginner level, user documentation, the differences in possible audience are minimal. A look at the potential audience for any of these books would reveal people with an interest in or need to know more about the topic and with no readily identifiable experience or prior knowledge. In other words, these books address an audience that is so heterogeneous as to make effective analysis beyond the purpose of the book impossible.

I do not evaluate the accuracy, completeness, or overall quality of the documentation. I am not addressing the usability of these documents, nor do I intend to imply that differences between these documents and the technical communication literature indicates that these documents are somehow superior to other examples. All I am considering is how procedural sets are constructed.

Process of Inquiry

I will qualitatively assess selected procedural sets from several recent books about Microsoft Word in order to determine if procedural sets in these books reflect common recommendations from the technical communication literature. Because one of the fundamental points from the technical communication literature is to minimize extraneous information from procedural sets, I will count occurrences of different types of information from each set of steps and summarize the findings.

Rather than assessing all procedural steps in all books selected, I will focus on specific procedural sets that I select to represent the overall style of procedural steps in each of the

books. I will choose the steps according to the following process, which should ensure accurate results.

I will assess processes that reflect instruction in using Microsoft Word, rather than instruction in using the operating system itself. For example, copying and pasting or opening and closing a file is common across all (standard-compliant) Windows programs, so the depth and thoroughness with which this process is addressed will likely to be quite variable across the different books. I will select common or important processes, giving preference to those that will likely be used by or useful to many Word users. I will address processes that are discussed in most books, but will, in selecting procedures, focus on selecting a fairly representative selection of different procedures, processes, and levels of complexity, in order to see the range of treatment in the text.

As Spyridakis (1992, 616) accurately notes, "software manuals . . . are often hard to test if naturally occurring materials are used because of the number of variables that differ across different products." However, to test occurrences of a phenomenon in real documentation, I will not be able to follow Spyridakis' suggestion of creating custom materials. As a result of the difficulty in assessing the documentation, I chose a relatively small sample and manually — not randomly — selected the samples. If the results of this inquiry are promising, more rigorous studies might be called for in future research in technical communication.

In discussions, Spyridakis recommends a sample size of at least 10 subjects per condition (1992, 615) and preferably somewhere between 10 and 20. I initially select 25 items to assess, anticipating potential problems with collecting usable results from all.

In selecting procedures, I attempted to find procedures that

- would be addressed in most of the books I chose to analyze
- would cover a range from very elementary instructions for beginning users to somewhat advanced topics for more skilled users.
- would require a variety of actions
- would be representative of the book
- would be handled as a separate section.

The steps I initially choose to assess were:

- inserting a table using the toolbar
- checking spelling using the dialog box
- checking grammar
- inserting header/footer
- inserting word art
- rearranging in outline mode
- customizing page margins
- inserting newspaper-style columns in dialog box
- selecting a printer
- using print preview
- insert merge fields
- changing or creating style characteristics in dialog box
- creating indent with toolbar
- turning on revision marks
- inserting table of contents
- changing line spacing in dialog box
- inserting symbols
- adding borders to text using toolbar
- wrapping text around a frame
- changing alignment with toolbar
- sorting a table
- creating a multi-level list
- converting text to table
- formatting characters in dialog box
- using a document template

Four procedures in the above list (*creating a multi-level list, converting text to table, formatting characters in dialog box, using a document template)* were handled unusually, inconsistently, or were not addressed in most books. I therefore eliminated them from

consideration, resulting in 21 sets with usable results. These sets of procedural steps will be assessed according to the following analysis process.

I will create forms, one for each procedure under analysis. Each form will have a line for each book and categories in which to record the heading and page number of the procedure — to allow me actually to return and check results later. Additionally, each line will have categories for the quantity of

- numbered steps
- actual steps that the reader must perform
- alternate actions
- cross-references
- screen references or descriptions
- system action descriptions
- elaboration

I include a distinction between actual steps and numbered steps because a preliminary look at the books revealed significant differences between the number of numbered steps and the actual number of actions the reader must take — I choose to collect both values in case the information proves useful later. In this thesis I will be using only the actual steps to assess procedural sets.

Rather than collecting all information from each book in succession, I will collect all data on a given procedure at one, then move through each procedure in turn.

The following chapter presents and discusses the results of this inquiry.

Chapter 4

RESULTS AND DISCUSSION

The results of this analysis of procedural sets in computer user documentation are quite varied, and overall surprisingly unlike the recommendations from the literature. As a whole, I found that procedural sets in commercially available documentation for Microsoft Word deviate considerably from the recommendations in the technical communication literature, particularly in terms of frequent use of adjunct information and elaboration throughout most of the samples.

In this section, I will briefly outline some general considerations about my findings, particularly in areas that will set the context for further discussion. Next, I will present my findings about the use of adjunct information, elaboration, and total supplemental information in the books, followed by specific issues peculiar to individual books and general discussion of the results. In the second half of this chapter, I will discuss my findings and the implications for technical communicators.

Overview and Observations

In general, all the commercially available books on Microsoft Word selected for this study have consistent and systematic use of both adjunct information and elaboration throughout. The specific amounts and cases vary greatly among books, but elaboration is consistently found.

Overall Coverage and Treatment

In general, the books selected are fairly similar in the treatment of the material and approach to the information. The quality is by no means uniform — some books proved remarkably difficult to use from an information access standpoint, but an objective look at the results shows differences only in degree in use of adjunct information and elaboration.

As I collected the data, few trends and tendencies among the books were apparent. The process of collecting all the data on each procedural step from all books, then moving to the next procedural step, effectively yielded data.

Because the raw data as collected from the books is a step of processing (to combine the individual measures into adjunct information) away from the source data I will discuss, no trends or patterns were evident throughout the data collection phase, other than the observation that all the books showed some degree of both adjunct information and elaboration.

See Appendix A for the complete tabular results of the data collection phase.

Adjunct Information

Each of the seven books under consideration reveals a fairly significant amount of adjunct information for each step. This information, ranging from figure references (very common) to alternate ways of performing a task (fairly common), varies little across the books.

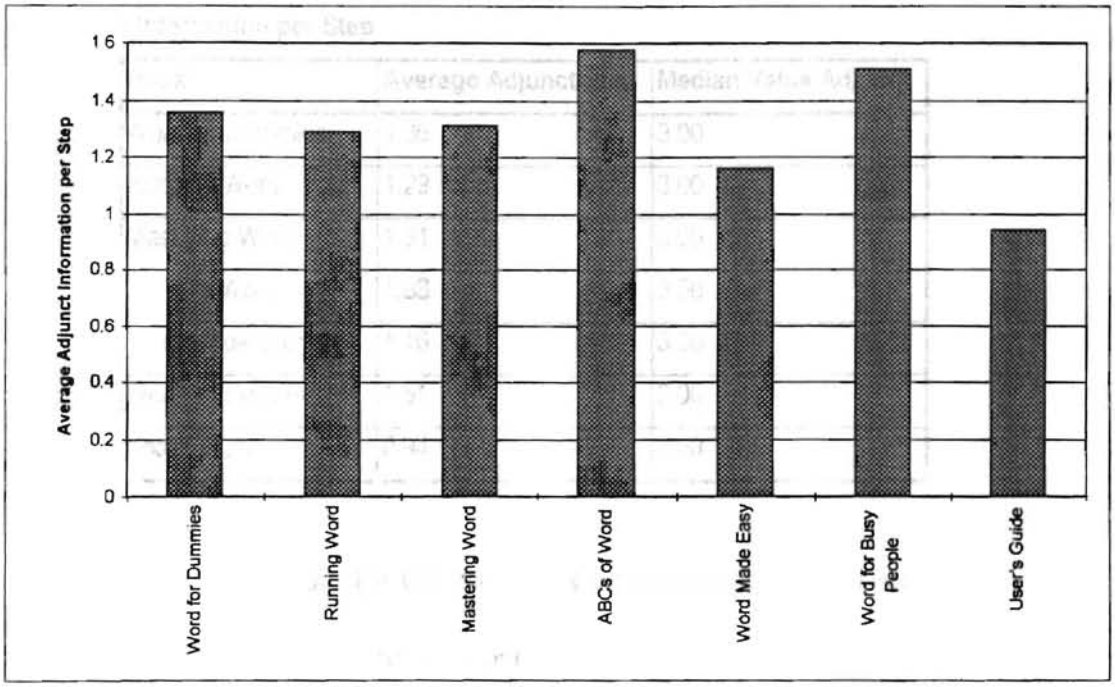


Figure 5: Average Adjunct Information per Actual Step of Procedure.

As shown in Figure 5, the amounts of adjunct information per step in all the commercial books lie in a fairly narrow range (1.16 to 1.58), with the *User's Guide* slightly lower at 0.94. On the whole, more than one piece of additional information is provided for every step in the procedure in all books except the *User's Guide.*

Table 4: Adjunct Information per Step

Book	Average Adjunct/Step	Median Value Adjunct
Word for Dummies	1.36	3.00
Running Word	1.29	3.00
Mastering Word	1.31	3.00
ABCs of Word	1.58	3.50
Word Made Easy	1.16	3.00
Word for Busy People	1.51	2.00
User's Guide	0.94	2.00

Table 4 shows the actual values for the average Adjunct information per step as well as the median values for adjunct information per step.

The relatively high median values (2.0-3.5) reflect a few sets of procedures — although not necessarily the same sets for each book — in which plentiful adjunct information was provided. These cases usually reflect a case of itemizing each field in a dialog box with either an explanation or a description.

Elaboration

As with adjunct information, all books showed consistent and regular use of elaboration in the studied procedural sets.

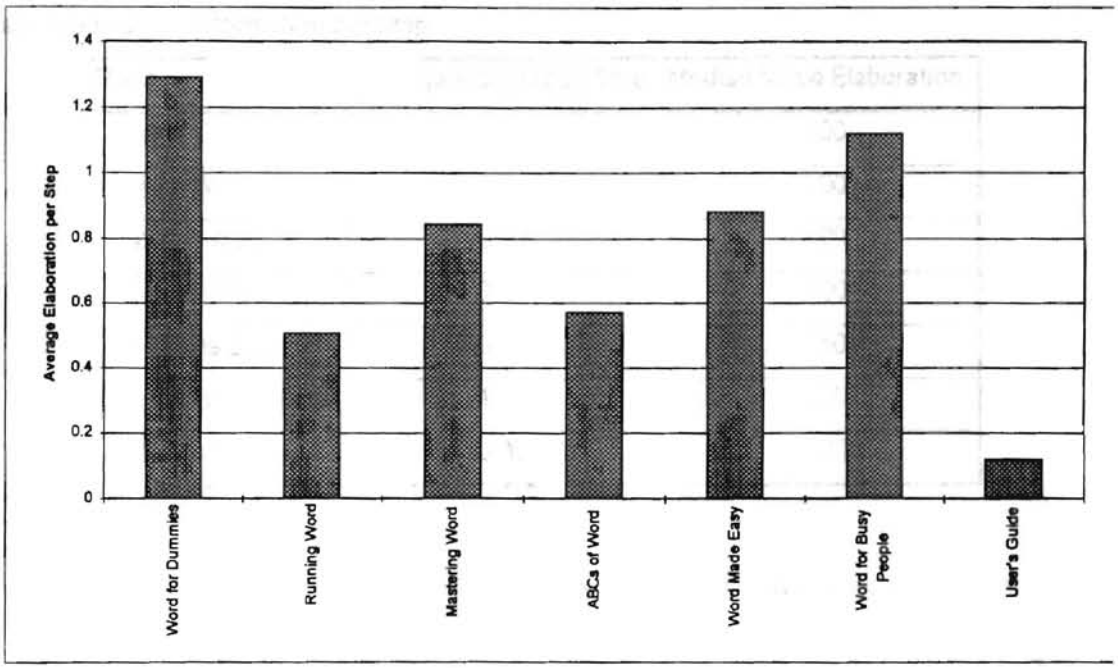


Figure 6: Average Elaboration per Actual Step of Procedure.

Because of the highly qualitative nature of the evaluation method, no significant results can be inferred from the differences in Elaboration as seen in Figure 6. Although W*ord for Dummies* and *Word for Busy People* show notably higher levels of elaboration than most of the rest — almost one additional instance of elaboration per actual step — there appears to be no systematic difference as well as no pattern to indicate that any particular books have more or less elaboration. *Word for Dummies* uses elaboration more than the other books, but all books except the Word *User's Guide* have substantial amounts of elaboration.

Table 5: Elaboration Information per Step

Book	Average Elaboration/Step	Median Value Elaboration
Word for Dummies	1.29	3.00
Running Word	0.51	1.00
Mastering Word	0.84	2.00
ABCs of Word	0.57	1.00
Word Made Easy	0.88	1.50
Word for Busy People	1.12	1.00
User's Guide	0.12	0.00

Table 5 shows that the median values for occurrences of elaboration are one or greater for all books except the *User's Guide*, and that the mean values exceed 0.5 for all books except the *User's Guide*, indicating a regular use of elaboration within the texts.

Total Supplemental Information

Examining all seven books in combination, few trends are immediately apparent in the data.

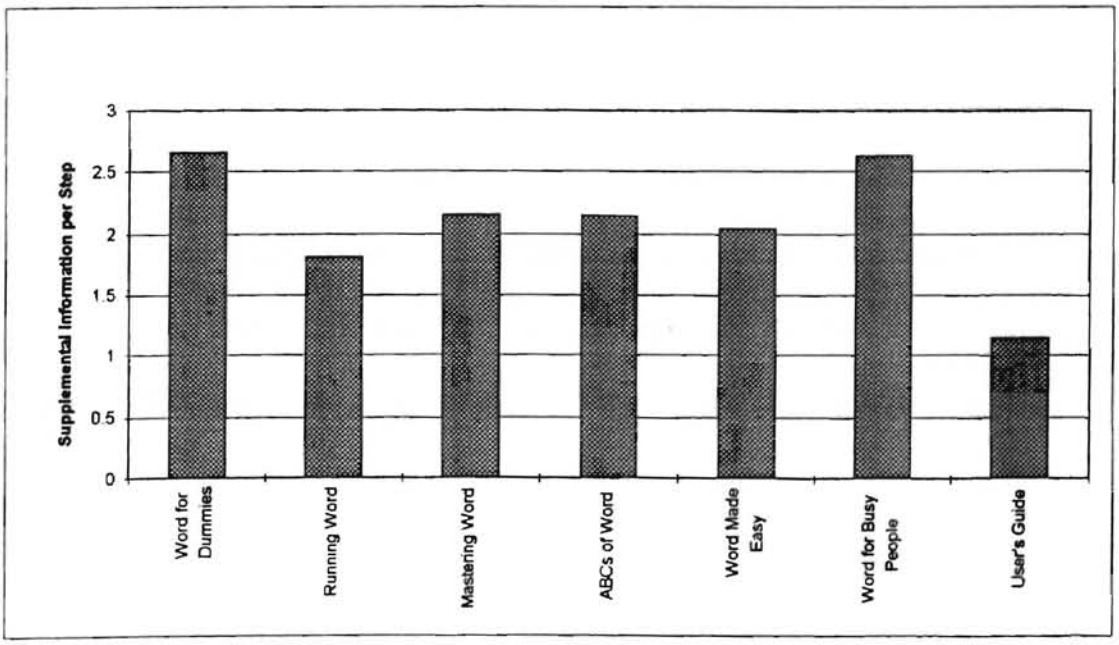


Figure 7: Total Supplemental Information per Step of Procedure.

As shown in Figure 7, the amounts of total supplemental information per procedure are between 1.0 and 2.5 — containing considerable information that is not strictly necessary for the procedure to be carried out — but show no pattern as far as which books have more supplemental information.

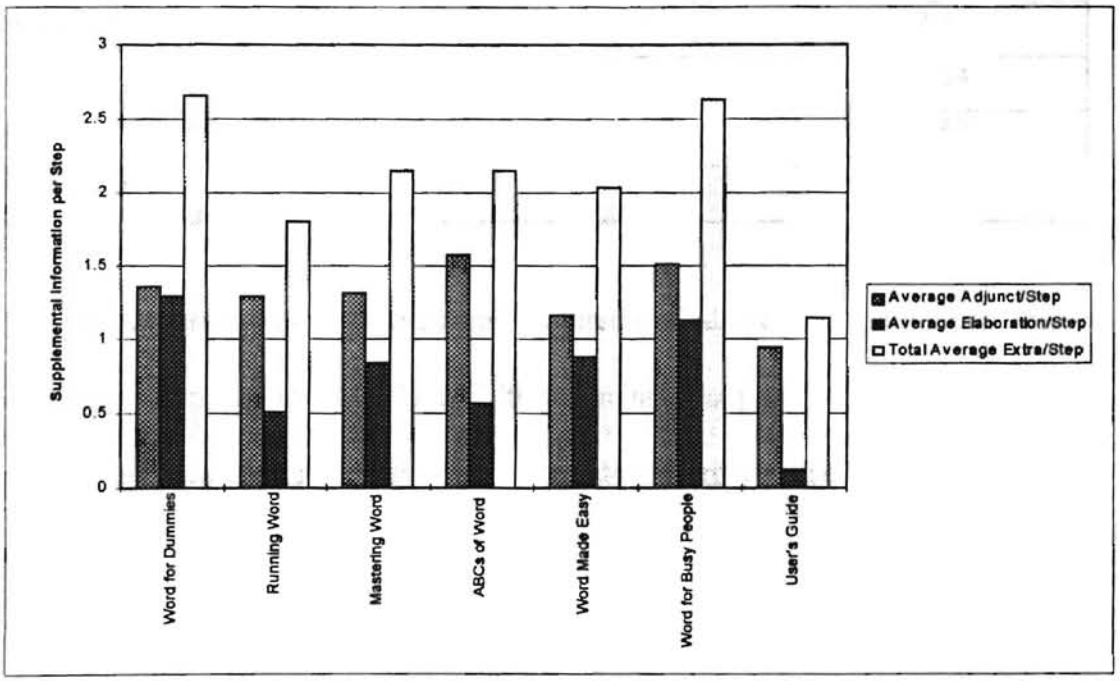


Figure 8: Supplemental Information.

As shown in Figure 8, adjunct information appears somewhat more frequently and consistently than elaboration, but does not appear to follow a specific pattern.

Some of the user's guides contain fairly high total levels of adjunct information and elaboration per step. Particularly *Word for Dummies* and *Word for Busy People* contains overall substantial amounts of extra information — 2.66 and 2.63 items per actual step.

Table 6: Summary Information Ratios

Book	Average Adjunct/Step	Average Elaboration/Step	Total Average Extra/Step
Word for Dummies	1.36	1.29	2.66
Running Word	1.29	0.51	1.80
Mastering Word	1.31	0.84	2.15
ABCs of Word	1.58	0.57	2.15
Word Made Easy	1.16	0.88	2.04
Word for Busy People	1.51	1.12	2.63
User's Guide	0.94	0.12	1.14

Table 6 shows average values for the number of items of adjunct information per step, elaboration per step, and the total "extra" (from a minimalist perspective) information per step in each procedural sets. However, no systematic differences in elaboration or adjunct information were apparent in the selected books.

Discussion

The process of extracting and analyzing the data in this study revealed few noteworthy problems or challenges, although a few interesting aspects were uncovered. I will review the unexpected constraints and surprises in my analysis procedure, then discuss anomalies in the data collected.

Procedural Discussion

With few exceptions, the data collection went smoothly. I encountered only minor difficulties with collecting the data, and those were fairly easily overcome. The problems included primarily

- issues with information access / missing procedures
- difficulty with identifying procedures

Poor information access design in certain books, most notably *Running Word*, made collecting the information difficult. Problems in the index, coupled with procedures categorized into highly arbitrary appearing sections, may have caused errors of omission, in which I claim that information is not included. In one case, *Running Word* did not include information about inserting a header or footer — as least as far as the index and table of contents showed — although that process was required for other procedures that were covered in the book.

As mentioned in the methodology section, four of the sets of procedures were eliminated from consideration because they were inconsistently handled, incompletely handled or omitted from several of the books. Those four procedures were probably a result of poor topic selection on my part — choosing relatively technical or obscure procedures — rather than indicative of substantive problems at a deeper level.

Some of the problems with information access may well be related to the previously discussed problems in which some books did not have all the procedures being analyzed. Specifically, only the *Microsoft User's Guide* covered every procedural set selected, while the remaining six books failed to cover from one to five of the procedural sets.

Table 7: Selected Procedures Not Covered

Book	Number of Procedures Not Covered
Word for Dummies	2
Running Word	2
Mastering Word	2
ABCs of Word	5

Word Made Easy	1
Word for Busy People	5
User's Guide	0

Table 7 shows the books studied and the number of sets they failed to cover. Because the comparisons of adjunct information and elaboration are based on the occurrences per actual step, rather than on an absolute value, these missing values do not pose problems for the accuracy or reliability of the data. As at least 16 procedural sets were analyzed from each book and the results were fairly consistent within books, it is highly unlikely that the missing data would change the results. The calculations of mean and median values excluded the procedural sets with no information.

The other problem mentioned above — difficulty identifying procedures — occurred primarily in cases in which a book or books presented a procedure within a paragraph or discussion, rather than separating the procedure as numbered list of actions. Some of the books — particularly the *ABCs of Microsoft Word* — used few or no numbered steps in procedures, choosing instead to bury all steps of the procedure within paragraphs of text. This format slightly slowed and hindered the data collection, but not insurmountably so.

I am confident that I accurately identified the procedural sets and the information within them, even when the book style or design precluded the use of a numbered list for the steps.

Data Discussion

The data collected proved remarkably consistent throughout the seven books. Although some books used more (*Word for Dummies* and *Word for Busy People*) or less

(*Word User's Guide*) adjunct information and elaboration, all books used both adjunct information and elaboration to some degree.

Some of this additional material can be described as passages in which the primary function is to support, encourage, and help the reader with more global issues, rather than passages specifically oriented to using a specific feature of the software. Instances of this material can often be found between steps within a numbered procedure and are not numbered. This elaboration is not essential — the instructions are complete even without this information. However, the addition of elaboration does in many cases have a purpose — it completes the overall instructional text by addressing, in terms of Hart's definition (1996, 144), not only the *what* but also the *why* readers are expected to do something.

Some issues within the books and procedural sets attracted my attention as I was collecting the data. I noticed that two of the commercial books were distinctly different from the remaining books — *Running Word* and *Mastering Word* are, as are the rest, labeled as user's guides for a beginning audience. However, rather than taking the approach of presenting only selected material clearly and effectively (as in . . . *for Dummies* or . . . *for Busy People*), these two volumes attempt to provide complete coverage of everything Word can do. As a result, these books are much longer than the other books and attempt to cover significantly more information. Specifically, *Running Word* and *Mastering Word* have 1009 and 995 pages respectively, the *Word User's Guide* has 823, while the remaining books range from 291 to 493 pages.

These noteworthy differences in approach did not appear to have an effect on the relative amounts of adjunct information or elaboration in procedural sets. I did observe, in

passing, that the size of the book does not necessarily indicate more comprehensive or effective coverage of most standard material.

The remaining three books — *ABCs of Word*, *Word Made Easy*, *Word for Busy People* — more closely resemble *Word for Dummies* in their relatively selective approach to the information and marketing push toward less experienced and less confident users. These three books, plus *Word for Dummies*, appear to address a slightly different audience from the two more comprehensive documents.

The levels of adjunct information and elaboration are somewhat surprising in their magnitude, although not the presence. As described in the Methodology section, adjunct information and elaboration are only assessed within procedural sets. Any information provided in introductions to the procedural set, in other areas of the section or part, or after the last step of the set, are not included in these numbers. Particularly in terms of the fairly conservative recommendations in the technical communication literature, providing this volume of additional information within procedural sets was somewhat surprising.

Of all the books, the *Microsoft User's Guide for Word* adheres fairly closely to standard technical communication recommendations, particularly insofar as the number of numbered steps matches the actual quantity of actions the reader must undertake and that few of the procedural sets include elaboration. Even with the *User's Guide*, however, the amount of adjunct information is somewhat surprising.

Grossly stated, a manual adhering to the conventional assumption in the literature that each numbered step should include one and only one action for the reader to perform would have the same number in both the # Steps and Actual Steps column. A manual written strictly to minimalist conventions would have no information recorded in the

Adjunct information or Elaboration columns. And, a manual designed to provide only necessary information (albeit at greater detail than a minimal manual) would contain no information in the Elaboration columns.

Additionally, the idea of having one numbered step for each action a reader must take seems fairly tightly ingrained as a given within the technical communication literature. However, I observed a great deal of disparity among these seven books in terms of use of numbered lists for procedural sets. For example, *Word Made Easy* uses no numbered steps — all procedural steps in the analyzed procedures in *Word Made Easy* are buried within paragraphs. *Mastering Word*, on the other hand, frequently numbers steps that require no action on the part of the reader. These "steps" are often descriptions of system actions or what the readers should expect to happen or see at a given point.

The next chapter concludes this thesis with a discussion of the possible reasons for these findings and recommendations for future research.

Chapter 5

CONCLUSION

In this study of computer user documentation, I assessed samples from commercial documentation to determine if they follow the guidelines presented in the technical communication literature. I found that commercial computer user documentation consistently deviates from the recommendations in the literature. In this chapter, I will explore possible explanations for this finding and outline possible future directions for research to investigate this discrepancy further.

Findings and Explanation

Although the technical communication literature falls into two schools — minimalism or the more liberal school focused on providing appropriate information — most recommendations from both areas of the literature specify that procedural sets should not be interrupted with any additional information. The literature indicates, as a whole, that information can and should reasonably be provided before the set of procedures, or after the set of procedures, but that the actual sequence of steps should not be interrupted.

On the other hand, my analysis of the actual composition of procedural sets from Microsoft Word documentation shows that procedural sets seem to be frequently and regularly interspersed with both adjunct information and elaboration.

This apparent disconnect between the literature recommendations on one hand and the practice of technical communication on the other hand could have several explanations, including:

- The technical communication literature currently lags behind the practice — that is, the practice is responding more quickly and possibly more effectively to the needs and demands of readers than the literature does, therefore the literature does not currently reflect the practice.
- The sample covers a broader scope in terms of multiple information goals than can be considered under the category of technical communication.
- The samples studied exhibit poor information structure and should — but do not — follow the recommendations in the literature.
- Commercial books are inherently different from non-commercial documentation samples and, while the study holds true for the given sample, it is a relatively minor part and not representative of technical communication as a whole.

My findings show that commercial software documentation does not reflect the recommendations from the literature, and it could be that this type of technical communication is distinctly different from those discussed in the literature.

As I conclude this thesis, I will address these possibilities in turn and close with a recommendation for continued inquiry and further development of this possibility.

Catching Up?

This study of some of the latest commercial software documentation shows a discrepancy between the recommendations in the literature and the practice. If the technical communication literature accurately reflects what happens in the practice of technical communication — although this is both unclear and beyond the scope of this

thesis to ascertain — a time lag between observing characteristics of published software documentation and publications about those characteristics would be understandable.

One possible reason for this discrepancy could be the very short development time of commercial software documentation as compared to the time to develop and publish scholarly papers or books. For example, many commercial software manuals, including those from IDG Books Worldwide and Osborne McGraw-Hill, are developed in two months, indexed and edited in another month, then printed and distributed in only a few more weeks — totaling approximately four months from conception to bookstore shelves. However, the proposal, submission, peer-review, scheduling for publication, and actual publication process for anthologies or for scholarly journals such as *Technical Communication*, takes far longer — often a year, quite often much longer.* Additionally, the time required to set up and conduct an empirical study greatly exceeds the time requires to learn and write about a software program. In more immediate terms, the entire development process for a commercial software manual could take place between the submission of a final manuscript and the publication in *Technical Communication* or similar journals.

This delay in publication could explain, at least in part, my finding that the literature does not accurately represent the technical communication practice. While the literature will certainly not always reflect the practice, I believe that significant discrepancies, such as those I found, call for further inquiry and research, preferably empirical research, to assess the usability and effectiveness of the differences in the practice.

* Times based on personal experience and personal e-mail with George Hayhoe, editor of *Technical Communication*.

On the other hand, we cannot tell from the information available if the practice leads the literature and the literature will soon catch up, or if this aspect of the practice is completely disconnected from the literature. Given the fact that commercial software documentation has existed for many years, and the series from which my samples were drawn range up to 5 years old, we could speculate that the characteristics observed in this thesis did not suddenly appear in this collection of books. In that case, perhaps the technical communication literature does not, in fact, reflect the practice, indicating that further study is called for.

In any case, more study would be required to ascertain definitively that the technical communication literature is lagging behind and will within a couple of publication cycles reflect the practice of a couple of years previously. However, I think that other explanations for the discrepancy I observed will prove more fruitful.

Typical Technical Communication?

Some might argue that this study does not accurately represent typical situations in technical communication. Some issues that raise questions about this study include the very fact that the documentation samples do not apparently fit standard technical communication practices, the multiple purposes and audiences for these samples, and the fairly limited scope of inquiry.

To those who question if this study can reflect a substantial aspect of technical communication because the samples do not reflect the technical communication literature, I would suggest that progress in the field may come from many fronts and that innovations in practice may well presage codification in the literature. Certainly a circular argument

that what is not in the literature cannot be in the practice because it is not in the literature should not be a concern for most readers.

I do not believe that the appearance of multiple and possibly conflicting audiences in this sample could possibly pose problems for further study or generalization. It could well be that these documents reflect an unusual combination of tutorial material, reference material, and more purely instructional material, but that cannot be peculiar to these documents. Many discussions of and publications in technical communication address the difficult and necessity of multiple audience considerations, and the combination possibly found in my sample is quite likely typical, or at least not unique.

Although the sample size is fairly limited, the results show an abundance of both adjunct information and elaboration within procedural sets. I suspect that similar studies of commercial documentation on other topics would produce similar results for the very reason that nothing about documentation for Microsoft Word appears, in any way, to be a special case or unusually likely to require significantly more elaboration or adjunct information than would be necessary for other software programs.

Overall, in response to any issues questioning the applicability of this study to technical communication, I would cite, again, David Dobrin: "Technical writing is writing that accommodates technology to the user." (Dobrin 1983, 242). By this among many definitions of technical communication, the purpose of technical communication is to provide usable information to readers. By any measure, the samples I selected strive to do so.

Poorly Structured Commercial Documentation?

Although, as stated above, the samples studied attempt to provide useful and effective information to readers, there exists little available evidence that the samples succeed in this aim. The possibility that the literature is correct and the commercially available documentation errs in deviating from the standards prescribed in much of the literature cannot be completely discounted and should probably be investigated through empirical research.

Is Commercial Documentation Different?

Although I have argued that my sample is, almost by definition, representative of technical communication, a closely-related possible explanation for my results also merits discussion. Is it possible that commercial software documentation as a whole cannot be assessed or evaluated by the same standards and measures used for most other technical communication materials? Are specific characteristics of commercial computer user documentation different enough from other computer documentation that they must be evaluated separately, and have not been assessed in the literature?

Before this question could be definitively answered, further studies would be required, but I will present an initial argument against this possibility.

These commercial software manuals attempt to provide information about how to use the product to a wide audience, just as traditional vendor-produced documentation does. The books do not appear to have strikingly different goals, rather, they have different stylistic approaches to providing information.

These commercial software manuals are, in and of themselves, a product, rather than being a part of a product as vendor-produced documentation is. As a separate product, authors and producers must address concerns of marketability and perceived usefulness that vendor-produced documentation does not have to address — at least not directly. However, this focus on audience preferences and tastes would, in my opinion, tend to improve the documentation more quickly and effectively than vendor-produced documentation would be affected. Any developments or changes in commercial documentation are unlikely to be substantively different from those in vendor-produced documentation, although they might very well occur faster.

The commercial documentation almost certainly responds to marketing and audience preferences, but I am not aware of any usability studies that have been performed on commercial software documentation. That said, it could be that commercial software documentation reflects what readers and customers want, while the technical communication literature reflects what readers find more usable and effective. This possibility raises the question of whether technical communicators should or do provide what their readers what or what studies indicate that they need.

In summary, the conditions and audience of both commercial and vendor-produced documentation are very similar, although the impetus for change and development is probably different. Commercial documentation, as it is driven by marketing concerns, will be quite responsive to indicated customer desires, while vendor-produced documentation might develop based on factors including usability testing, consumer preference, the old "that is how we have always done it" factor, cost considerations, internal or external

politics, or other factors. Vendor-produced documentation has a much murkier relationship to profit than commercial documentation does.

One related consideration that cannot account for the complete discrepancy between my findings and the literature, but is worthy of mention, is the relatively greater freedom that writers of commercial documentation have. It appears that much of the literature in computer documentation addresses technical writers on a software development team or at least working for the same company that produces the product.

Another possible — but unlikely — explanation for my findings is that that commercial documentation has more adjunct information and elaboration than vendor-produced documentation because the writers of commercial documentation are more free to state problems and issues clearly than the writers of vendor-produced documentation. Elaboration in product manuals is at least potentially problematic because many likely applications of elaboration are the very aspects of the software that are unintuitive, unclear, or difficult — not exactly what a company generally wants to include in their documentation. I do not think that issue could possibly account for the entire discrepancy in findings for a number of reasons — among them the fact that the writers of commercial documentation also have a vested interest in the success of the product — but this issue could merit further study at a later time.

Computer User Documentation is a Special Case

At this point, the other likely explanations for the observed discrepancy between the technical communication literature's recommendations for procedural sets and the use of procedural sets as seen in a selected sample of documentation have been rejected. This

leaves a significant discrepancy between academic recommendations and practice, in a field that appears to be experiencing significant growth and development.

What does this mean to us as technical communicators? First, a brief review of my findings: I found that the technical communication literature very consistently recommends providing no elaboration and little adjunct information within procedural steps. A significant sample of the literature advocates minimal documentation in which only the most essential information is provided, and certainly not anything on the order of one or two pieces of adjunct information or elaboration for each and every procedural step.

It appears that computer documentation poses a unique problem for technical communication. The sheer number of publications, articles, books focusing on the topic shows the frustration and confusion that we as technical communicators experience when we try to make computer documentation fit into traditional communication paradigms. The fact is, computer documentation does not fit the paradigms to which we have become accustomed.

Computer user documentation does not now fit traditional technical communication paradigms in several ways. Modern computer user instructions are not generally completely sequential — as writers, we cannot assume the process that readers will follow through the instructions, nor can or should we guide our readers on one path through a software program when the readers will actually pick and choose and therefore read only the parts that apply to them.

Computer user documentation is not modular — as writers, we cannot write disconnected sections of documentation and expect a significant proportion of our users to be able to make the necessary connections between different sections. Although readers

can certainly make some connections, due to the complexity of the material and the need to make accurate and useful connections, we must help establish a schema and build a framework in which our readers can fit each additional piece of information.

Computer user documentation is iterative — as writers, we cannot write about computer software without assuming prior actions and effects, although we do not know what the prior actions might be. Computer user documentation interprets exceedingly complex systems and processes for novice users — as writers, we have neither the luxury of simplifying the description nor of assuming any significant level of experience or knowledge on the part of our readers.

In other words, computer user documentation is used for reference purposes, but requires more connections and tutorial-type information than traditional reference materials written for advanced users. Computer user documentation is used for tutorial purposes, but require reading-to-do access and quick paths to just the essential information to complete a job. Computer user documentation attempts to be all things to all users, and must be so.

It could be argued that the problem computer user documentation seems designed to solve is not really a documentation problem — it is a training problem. Provide adequate training on software, then provide reference documentation, thereby solving the problem. Alternatively, provide tutorial documentation and a separate set of reference documentation, thereby solving the problem. However, that solution, while logical, is not practical or financially viable.

Computers and software are increasingly considered tools to be used to do other jobs — just as a voltmeter, set of wrenches, or pressure gauge are tools. That implies that the

documentation should be straightforward — do this, do that, then do something else. However, for a number of reasons ranging from experience to a more thorough user understanding to a greatly simpler process, these more traditional tools do not adequately compare with computers as tools.

Let us take as an example a computer user who has and uses the computer operating system, a word processing program, a spreadsheet or money management program, an email program, and a Web browser. An operating system and four programs would be a bare minimum configuration, but adequate for this discussion. Assuming conservatively that a new version of each of the programs is released every 20 months, this computer user would need to attend one training class, buy at least one book, or engage in intensive self-study every four months just to keep up with the existing software, let alone to use other programs.

However, the sociological complication is that it appears that many people also expect that computers are tools — virtually commodities, as a VCR or television. Many people who use computers are not generally paid to use the computer — they are paid to be accountants, technicians, customer support representatives, or to do any of a number of other things, and they need to use computer software as tools. The readers have little motivation to learn all this computer and technical information just to be a professor or secretary or accountant or whatever. For the computer to be a tool, the reader must understand and be able to use it effectively.

However, as we can see from browsing through bookstores and training catalogs, many people who must use computers consider themselves Dummies (or Idiots, but not yet Morons) in the use of computers. Traditional documentation — as recommended by

the technical communication literature — often requires that readers follow steps by rote, with no particular understanding of the process. These steps do not include additional information, and address only the task based information (Romiszowski 1981).

Unfortunately, as long as readers continue using manuals with a read-to-do approach, they will consider themselves Dummies — they are reading to do exclusively and when something goes wrong, they are stuck. Likewise, with the number of new programs appearing on the market, they do not have time to read to learn — they need to learn, but they more desperately need to do.

Perhaps a more productive long term approach would be to train in the use of the tools and not consider this to be a problem with computer documentation in the first place. Perhaps a thorough training program would be beneficial, but the costs in time and effort would be enormous. For a computer user who uses Excel, Word, an e-mail program and a Web browser, all on Windows 95, comprehensive training in each software package as well as the operating system would be extremely time consuming, extremely expensive, and would have to be at least partially redone at every software revision. Logistically, that approach just cannot work. Thorough training additionally implies deeper knowledge, which takes significantly more effort to instill than task-based procedures.

At this point, of course, we do not know enough about a number of these variables to do anything more than speculate about possible explanations. I believe, however, that sufficient evidence exists to call for further study into the use of procedural sets in documentation, designed to help users read to do and learn.

Although Charney, Reder, and Wells already conducted initial investigations into elaborated texts, ongoing rapid developments in computer technology, including the

graphic user interface and common look-and-feel applications, call for replication of their study. As an additional step, I would propose a usability test comparing elaborated and unelaborated texts for reference or reading-to-do purposes. Each of two groups would follow specific procedures, with one group using elaborated texts and one using un-elaborated texts. Second, each group would be asked to do a similar, but undocumented, procedure, to ascertain if elaboration in the texts helps transfer from one task to another. Finally, a follow-up study after three months or six months would help establish the retention for each of the groups.

I would call for these investigations as quickly as possible, because the rapid encroachment of computer technology into the everyday lives of all of us makes it more difficult by the day to find groups with appropriately limited experience with computers to allow for effective testing. As computer user interfaces tend to be fairly similar in many respects, any prior knowledge on the part of the participants must be accounted for in the study because any prior knowledge could lead to unwanted transfer into the tasks in the study.

In a similar vein, concurrent usability tests of a variety of commercial software documentation should attempt to find out if the elaboration and adjunct information found in the practice benefits readers or not.

Based on my initial findings, I would strongly suggest that further empirical research is required in the area of procedural sets to learn more about the usability of the style used in much commercial documentation, as well as the usability of different forms of procedural steps. Additionally, as ever, technical communicators should attempt carefully to analyze and respond to the needs of their readers. Further inquiry, particularly empirical

studies, into this area could certainly prove enlightening and useful for technical communicators and their readers.

BIBLIOGRAPHY

1. Alred, Gerald J., Walter E. Oliu, and Charles T. Brusaw. 1992. *The Professional Writer: A Guide for Advanced Technical Writing*. New York: St. Martin's Press.

2. Anderson, Paul. 1985. "What Survey Research Tells Us About Writing At Work." *In Writing in Nonacademic Settings*, ed. Lee Odell and Dixie Goswami, 1-84. New York: The Guilford Press.

3. Bereiter, Carl, and Marlene Scardamalia. 1983. "Levels of Inquiry in Writing Research." In *Research on Writing: Principles and Methods*, ed. Peter Mosenthal, Lynne Tamor, and Sean A. Walmsley, 3-25. New York: Longman.

4. Bloom, Benjamin S. 1956. *Taxonomy of Educational Objectives*. Ann Arbor, Michigan: David McKay Company, Inc.

5. Brockmann, R. John. 1990. *Writing Better Computer User Documentation: From Paper to Hypertext*. New York: John Wiley and Sons, Inc.

6. Caernarven-Smith, Patricia. 1983. *Audience Analysis and Response*. Pembroke, Massachusetts: Firman Technical Publications, Inc.

7. Carliner, Saul. 1993. "A Way with Words — Presenting Information Verbally." In *Techniques for Technical Communicators*, ed. Carol M. Barnum and Saul Carliner, 138-186. New York: Macmillan Publishing Company.

8. Carroll, John M. 1990. *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill*. Cambridge, Massachusetts: The MIT Press.

9. Charney, D.H., Reder, L.M., and Wells. G. 1988. "Studies of elaboration in instructional texts." In *Effective Documentation*, ed. S. Doheny-Farina, 47-72. Cambridge: MIT Press.

10. Chisholm, Richard M. 1988. "Improving the Management of Technical Writers: Creating a Context for Usable Documentation." In *Effective Documentation*, ed. Steve Doheny-Farina, 299-327. Cambridge: MIT Press.

11. Chomsky, Noam. 1971. *Chomsky: Selected Readings*. Ed Allen, J. P. B., and Paul Van Buren. London: Oxford University Press.

12. Cohen, Nancy E. 1991. "Problems of Form in Software Documentation." In *Perspectives on Software Documentation: Inquiries and Innovations*, ed. Thomas T. Barker, 123-136. From Baywood's Technical Communication Series, ed. Jay R. Gould. Amityville, New York: Baywood Publishing Company, Inc.

13. De Beaugrande, Robert. 1988. "How Can Rhetorical Theory Help Writers Create Readable Documents?" In *Solving Problems in Technical Writing*, ed. Lynn Beene and Peter White, 79-90. New York: Oxford University Press.

14. Dobrin, David N. 1983. "What's Technical About Technical Writing?" In *New Essays in Technical and Scientific Communication: Research, Theory, and Practice*, ed. Paul V. Anderson, R. John Brockmann, and Carolyn R. Miller, 227-250. Farmingdale, New York: Baywood Publishing Company.

15. Fawcett, Heather, Samuel Ferdinand, and Ann Rockley. 1993. "The Design Draft — Organizing Information." In *Techniques for Technical Communicators*, ed. Carol M. Barnum and Saul Carliner, 44-78. New York: Macmillan Publishing Company.

16. Hart, Geoff. 1996. "The Five W's: An Old Tool for the New Task of Audience Analysis." *Technical Communication* 43 (2nd Quarter): 139-145.

17. Hartley, James. 1985. *Designing Instructional Text*. New York: Nichols Publishing Company.

18. Haydon, Leslie M. 1995. *The Complete Guide to Writing and Producing Technical Manuals*. New York: John Wiley and Sons, Inc.

19. Holland, V. Melissa, Veda R. Charrow, and William W. Wright. 1988. "How Can Technical Writers Write Effectively for Several Audiences at Once?" In *Solving Problems in Technical Writing*, ed. Lynn Beene and Peter White, 27-54. New York: Oxford University Press.

20. Holtz, Herman. 1988. *The Complete Guide to Writing Readable User Manuals*. Homewood, Illinois: Dow Jones-Irwin.

21. Horn, Robert E. 1980. *Developing Instructional Materials and Procedures*. Lexington, Massachusetts: Information Resources, Inc.

22. Houp, Kenneth W., Thomas E. Pearsall, and Elizabeth Tebeaux. 1995. *Reporting Technical Information*, 8th edition. Needham Heights, Massachusetts: Allyn and Bacon.

23. IngramBook. 1996. *Computers-Word Processing Top 50 Requested Titles*. 1996. From http://www.ingrambook.com/TITLEWAVE/PRODUCT_INFO/CATEGORY_INFO/category_files/xp.html, September 1996, Ingram, Inc. Visited September 25, 1996.

24. IngramMicro. 1996. *Personal Computer Bestselling Software*. 1996. From http://www.ingram.com/whatsnew/pbestsl.html, September 1996, Ingram, Inc. Visited September 25, 1996.

25. Jonassen, David H. 1985. "Generative Learning vs. Mathemagenic Control of Text Processing." In *The Technology of Text Volume Two: Principles for Structuring, Designing, and Displaying Text*, ed. David H. Jonassen, 9-45. Englewood Cliffs, New Jersey: Educational Technology Publications.

26. Jonassen, David H. 1982. "Advance Organizers in Text." In *The Technology of Text*, ed. David H. Jonassen, 253-276. Englewood Cliffs, New Jersey: Educational Technology Publications.

27. Jonassen, David H. 1982. "Introduction to Section One: Implicit Structures in Text." In *The Technology of Text*, ed. David H. Jonassen, 5-13. Englewood Cliffs, New Jersey: Educational Technology Publications.

28. Lazonder, Ard W. 1994. "Minimalist Documentation and the Effective Control of Errors." In *Quality of Technical Documentation*, ed. Michael Steehouder, Carel Jansen, Pieter van der Poort, and Ron Verheijen, 85-98. From Utrecht Studies in Language and Communication Series, ed. Paul van den Hoven and Wolfgang Herrlitz. Amsterdam: Rodopi.

29. LeFevre, Jo-Anne, and Peter Dixon. 1986. "Do Written Instructions Need Examples?" *Cognition and Instruction*. 3(1): 1-30.

30. Littlejohn, Stephen W. 1992. *Theories of Human Communication*, 4th edition. Belmont, California: Wadsworth Publishing Company.

31. Mackh, Georgia E., and Lois Johnson Rew. 1991. "Using Information Aids to Boost Information Retrieval." *Technical Communication.* 38 (2nd Quarter): 210-213.

32. Mårdsjö, Karin. 1994. "Man — Text — Technology: Technical Manuals as Means of Communication." In *Quality of Technical Documentation*, ed. Michael Steehouder, Carel Jansen, Pieter van der Poort, and Ron Verheijen, 185-200. From Utrecht Studies in Language and Communication Series, ed. Paul van den Hoven and Wolfgang Herrlitz. Amsterdam: Rodopi.

33. Marshall, Catherine, and Gretchen B. Rossman. 1989. *Designing Qualitative Research*. Newbury Park, California: Sage Publications.

34. Miller, George A., Eugene Galanter, and Karl H. Pribram. 1960. *Plans and the Structure of Behavior*. New York: Holt, Rinehart, and Winston, Inc.

35. Mirel, Barbara, Susan Feinberg, and Leif Allmendinger. 1991. "Designing Manuals for Active Learning Styles." *Technical Communication.* 38 (1st Quarter): 75-87.

36. Morgan, Meg. 1988. "Empirical Research Designs: Choices for Technical Communicators." In *Effective Documentation*, ed. Steve Doheny-Farina, 25-44. Cambridge: MIT Press.

37. Mulcahy, P. I. 1989. "A problem solving schema for task instructions." In *Proceedings of the 36th International Technical Communication Conference*, RT119-121. Arlington, Virginia: Society for Technical Communication.

38. Odell, Lee. 1985. "Relations between Writing and Social Context." *In Writing in Nonacademic Settings*, ed. Lee Odell and Dixie Goswami, 129-153. New York: The Guilford Press.

39. Oram, Andrew. 1992. "Do We Dare to Free Our Computer Users?" *Technical Communication.* 39 (1st Quarter): 60-68.

40. Pearsall, Thomas E. 1969. *Audience Analysis for Technical Writing*. Beverly Hills, California: Glencoe Press.

41. Price, Jonathan. 1984. *How to Write a Computer Manual: A Handbook of Software Documentation*. Menlo Park, California: Benjamin/Cummings Publishing Company.

42. Ramey, Judith. 1988. "How People Use Computer Documentation: Implications for Book Design." In *Effective Documentation*, ed. Steve Doheny-Farina, 143-158. Cambridge: MIT Press.

43. Redish, Janice C. 1993. "Understanding Readers." In *Techniques for Technical Communicators*, ed. Carol M. Barnum and Saul Carliner, 14-42. New York: Macmillan Publishing Company.

44. Redish, Janice C., Robbin M. Battison, and Edward S. Gold. 1985. "Making Information Accessible to Readers." In *Writing in Nonacademic Settings*, ed. Lee Odell and Dixie Goswami, 129-153. New York: The Guilford Press.

45. Romiszowski, A. J. 1981. *Designing Instructional Systems: Decision Making in Course Planning and Curriculum Design*. New York: Nichols Publishing.

46. Sanders, Scott P. 1988. "How Can Technical Writing Be Persuasive" In *Solving Problems in Technical Writing*, ed. Lynn Beene and Peter White, 55-78. New York: Oxford University Press.

47. Sari, Fulya I, and Charles M. Reigeluth. 1982. "Writing and Evaluating Textbooks: Contributions from Instructional Theory." In *The Technology of Text*, ed. David H. Jonassen, 53-90. Englewood Cliffs, New Jersey: Educational Technology Publications.

48. Sebrechts, Marc M., Joseph G. Desk, and John B. Black. 1983. "A Diagrammatic Approach to Computer Instruction for the Naïve User." *Behavior Research Methods and Instrumentation.* 15(2): 200-207.

49. Seels, Barbara, and Zita Glasgow. 1990. *Exercises in Instructional Design*. Columbus, Ohio: Merrill Publishing Company.

50. Sides, Charles H. 1991. *How to Write and Present Technical Information*, 2nd edition. Phoenix, Arizona: Oryx Press.

51. Smith, Edward E., and Lorraine Goodman. 1984. "Understanding Written Instructions: The Role of an Explanatory Schema." *Cognition and Instruction*. 1(4): 359-396.

52. Spyridakis, Jan H. 1992. "Conducting Research in Technical Communication: The Application of True Experimental Designs." *Technical Communication* 39 (4th Quarter): 607-624.

53. Steehouder, Michael. 1994. "The Quality of Access: Helping Users Find Information In Documents." In *Quality of Technical Documentation*, ed. Michael Steehouder, Carel Jansen, Pieter van der Poort, and Ron Verheijen, 131-143. From Utrecht Studies in Language and Communication Series, ed. Paul van den Hoven and Wolfgang Herrlitz. Amsterdam: Rodopi.

54. Ummelen, Nicole. 1994. "Procedural and Declarative Information: A Closer Examination of the Distinction." In *Quality of Technical Documentation*, ed. Michael Steehouder, Carel Jansen, Pieter van der Poort, and Ron Verheijen, 115-130. From Utrecht Studies in Language and Communication Series, ed. Paul van den Hoven and Wolfgang Herrlitz. Amsterdam: Rodopi.

55. van der Meij, Hans. 1994. "Catching the User in the Act." In *Quality of Technical Documentation*, ed. Michael Steehouder, Carel Jansen, Pieter van der Poort, and Ron Verheijen, 201-210. From Utrecht Studies in Language and Communication Series, ed. Paul van den Hoven and Wolfgang Herrlitz. Amsterdam: Rodopi.

56. Warren, Thomas L. 1993. "Three Approaches to Reader Analysis." *Technical Communication* 40 (1st Quarter): 81-88.

57. Weiss, Edmond H. 1991. *How to Write Usable User Documentation*. Phoenix, Arizona: Oryx Press.

58. West, Charles K., James A. Farmer, Phillip M. Wolff. 1991. *Instructional Design: Implications from Cognitive Science*. Englewood Cliffs, New Jersey: Prentice Hall.

59. Wieringa, Douglas, Christopher Moore, and Valerie Barnes. 1993. *Procedure Writing: Principles and Practices*. Columbus, Ohio: Battelle Press.

60. Wollever, Kristin R. and Helen M. Loeb. 1994. *Writing for the Computer Industry*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc.

61. Wright, Patricia. 1994. "Quality or Usability? Quality Writing Provokes Quality Reading." In *Quality of Technical Documentation*, ed. Michael Steehouder, Carel Jansen, Pieter van der Poort, and Ron Verheijen, 7-38. From Utrecht Studies in Language and Communication Series, ed. Paul van den Hoven and Wolfgang Herrlitz. Amsterdam: Rodopi.

62. Zimmerman, Carolyn M. and John J. Campbell. 1988. *Fundamentals of Procedure Writing*, 2nd edition. Columbia, Maryland: GP Publishing, Inc.

APPENDIX A — COMMERCIAL DOCUMENTATION SAMPLES

Borland, Russell. 1995. *Running Microsoft Word for Window 95*. Redmond, Washington: Microsoft Press. (40th on bestseller list)

Crumlish, Christian. 1996. *Word for Windows 95 for Busy People*. Berkeley, California: Osborne McGraw-Hill.

Gookin, Dan. 1995. *Word for Windows 95 for Dummies*. Foster City, California: IDG Books Worldwide, Inc. (3rd on bestseller list)

Hart-Davis, Guy. 1996. *ABCs of Word for Windows 95*. Alameda, California: Sybex, Inc.

Mansfield, Ron. 1995. *Mastering Word for Windows 95*. Alameda, California: Sybex, Inc. (24th on bestseller list)

Microsoft Word User's Guide. 1994. Redmond, Washington: Microsoft. (Program documentation, previous version.)Neibauer, Alan R. 1995. *Microsoft Word for Windows 95 Made Easy*. Berkley, California: Osborne McGraw-Hill.

APPENDIX B — TABULAR RESULTS

Table 8: Summary Results for *Microsoft User's Guide for Word*

Topic	# Steps	Actual Steps	Adjunct Information	Elaboration
Adding Borders to Paragraphs with Toolbar	4	4	11	0
Adding Header	4	4	7	0
Changing Alignment with Toolbar	2	2	4	1
Changing Line Spacing in Dialog Box	5	5	2	0
Checking Grammar	3	3	10	1
Checking Spelling	3	3	8	0
Creating Indent with Toolbar	2	2	1	0
Creating or Changing Styles in Dialog Box	6	6	1	2
Customizing Margins	5	5	2	0
Insert Merge Fields	3	3	0	2
Inserting Newspaper Columns in Dialog Box	0	1	0	0
Inserting Symbols or Special Characters	5	5	1	1
Inserting Table of Contents	6	6	3	0
Inserting Table with Toolbar	3	3	3	1
Inserting Word Art	7	8	3	0
Rearranging Topics in Outline Mode	3	3	1	0
Selecting a Printer	0	1	1	0
Sorting a Table	7	7	1	1
Turning On Revision Marks	5	5	3	0
Using Print Preview	3	3	10	0
Wrapping Text Around a Frame	3	3	0	0

Table 9: Supplemental Information Ratios for *Microsoft User's Guide for Word*

Topic	Adjunct/Step	Elaboration/ Step	Total Extra/Step
Adding Borders to Paragraphs with Toolbar	2.75	0.00	2.75
Adding Header	1.75	0.00	1.75
Changing Alignment with Toolbar	2.00	0.50	2.50
Changing Line Spacing in Dialog Box	0.40	0.00	0.40
Checking Grammar	3.33	0.33	3.67
Checking Spelling	2.67	0.00	2.67
Creating Indent with Toolbar	0.50	0.00	0.50
Creating or Changing Styles in Dialog Box	0.17	0.33	0.50
Customizing Margins	0.40	0.00	0.40
Insert Merge Fields	0.00	0.67	0.67
Inserting Newspaper Columns in Dialog Box	0.00	0.00	0.00
Inserting Symbols or Special Characters	0.20	0.20	0.40
Inserting Table of Contents	0.50	0.00	0.50
Inserting Table with Toolbar	1.00	0.33	1.33
Inserting Word Art	0.38	0.00	0.38
Rearranging Topics in Outline Mode	0.33	0.00	0.33
Selecting a Printer	1.00	0.00	1.00
Sorting a Table	0.14	0.14	0.29
Turning On Revision Marks	0.60	0.00	0.60
Using Print Preview	3.33	0.00	3.33
Wrapping Text Around a Frame	0.00	0.00	0.00

Table 10: Summary Results for *Word for Dummies*

Topic	# Steps	Actual Steps	Adjunct Information	Elaboration
Adding Borders to Paragraphs with Toolbar	5	4	2	4
Adding Header	5	5	6	2
Changing Alignment with Toolbar	5	4	2	0
Changing Line Spacing in Dialog Box	0	2	3	4
Checking Grammar	0	1	5	3
Checking Spelling	3	2	6	1
Creating Indent with Toolbar	3	3	1	5
Creating or Changing Styles in Dialog Box	9	9	3	4
Customizing Margins	6	6	2	6
Insert Merge Fields	4	4	1	7
Inserting Newspaper Columns in Dialog Box	7	7	3	3
Inserting Symbols or Special Characters	4	3	3	3
Inserting Table of Contents	0	1	2	2
Inserting Table with Toolbar	7	7	5	4
Inserting Word Art	6	6	5	5
Rearranging Topics in Outline Mode	0	1	5	2
Selecting a Printer	3	3	3	3
Sorting a Table	0	1	0	4
Turning On Revision Marks	N/A	N/A	N/A	N/A
Using Print Preview	0	1	2	1
Wrapping Text Around a Frame	N/A	N/A	N/A	N/A

Table 11: Supplemental Information Ratios for *Word for Dummies*

Topic	Adjunct/Step	Elaboration/ Step	Total Extra/Step
Adding Borders to Paragraphs with Toolbar	0.50	1.00	1.50
Adding Header	1.20	0.40	1.60
Changing Alignment with Toolbar	0.50	0.00	0.50
Changing Line Spacing in Dialog Box	1.50	2.00	3.50
Checking Grammar	5.00	3.00	8.00
Checking Spelling	3.00	0.50	3.50
Creating Indent with Toolbar	0.33	1.67	2.00
Creating or Changing Styles in Dialog Box	0.33	0.44	0.78
Customizing Margins	0.33	1.00	1.33
Insert Merge Fields	0.25	1.75	2.00
Inserting Newspaper Columns in Dialog Box	0.43	0.43	0.86
Inserting Symbols or Special Characters	1.00	1.00	2.00
Inserting Table of Contents	2.00	2.00	4.00
Inserting Table with Toolbar	0.71	0.57	1.29
Inserting Word Art	0.83	0.83	1.67
Rearranging Topics in Outline Mode	5.00	2.00	7.00
Selecting a Printer	1.00	1.00	2.00
Sorting a Table	0.00	4.00	4.00
Turning On Revision Marks	N/A	N/A	N/A
Using Print Preview	2.00	1.00	3.00
Wrapping Text Around a Frame	N/A	N/A	N/A

Table 12: Summary Results for *Mastering Word*

Topic	# Steps	Actual Steps	Adjunct Information	Elaboration
Adding Borders to Paragraphs with Toolbar	7	7	1	0
Adding Header	5	4	4	2
Changing Alignment with Toolbar	2	2	2	0
Changing Line Spacing in Dialog Box	5	5	3	0
Checking Grammar	4	2	2	5
Checking Spelling	0	1	6	3
Creating Indent with Toolbar	0	1	1	3
Creating or Changing Styles in Dialog Box	N/A	N/A	N/A	N/A
Customizing Margins	7	6	2	0
Insert Merge Fields	13	12	1	4
Inserting Newspaper Columns in Dialog Box	0	1	2	1
Inserting Symbols or Special Characters	0	2	3	2
Inserting Table of Contents	7	7	3	2
Inserting Table with Toolbar	4	3	2	1
Inserting Word Art	11	12	4	4
Rearranging Topics in Outline Mode	0	2	4	0
Selecting a Printer	4	3	6	2
Sorting a Table	11	8	3	0
Turning On Revision Marks	3	2	3	0
Using Print Preview	0	1	3	3
Wrapping Text Around a Frame	N/A	N/A	N/A	N/A

Table 13: Supplemental Information Ratios for *Mastering Word*

Topic	Adjunct/Step	Elaboration/ Step	Total Extra/Step
Adding Borders to Paragraphs with Toolbar	0.14	0.00	0.14
Adding Header	1.00	0.50	1.50
Changing Alignment with Toolbar	1.00	0.00	1.00
Changing Line Spacing in Dialog Box	0.60	0.00	0.60
Checking Grammar	1.00	2.50	3.50
Checking Spelling	6.00	3.00	9.00
Creating Indent with Toolbar	1.00	3.00	4.00
Creating or Changing Styles in Dialog Box	N/A	N/A	N/A
Customizing Margins	0.33	0.00	0.33
Insert Merge Fields	0.08	0.33	0.42
Inserting Newspaper Columns in Dialog Box	2.00	1.00	3.00
Inserting Symbols or Special Characters	1.50	1.00	2.50
Inserting Table of Contents	0.43	0.29	0.71
Inserting Table with Toolbar	0.67	0.33	1.00
Inserting Word Art	0.33	0.33	0.67
Rearranging Topics in Outline Mode	2.00	0.00	2.00
Selecting a Printer	2.00	0.67	2.67
Sorting a Table	0.38	0.00	0.38
Turning On Revision Marks	1.50	0.00	1.50
Using Print Preview	3.00	3.00	6.00
Wrapping Text Around a Frame	N/A	N/A	N/A

Table 14: Summary Results for *Running Word*

Topic	# Steps	Actual Steps	Adjunct Information	Elaboration
Adding Borders to Paragraphs with Toolbar	2	2	2	0
Adding Header	N/A	N/A	N/A	N/A
Changing Alignment with Toolbar	2	2	3	0
Changing Line Spacing in Dialog Box	0	1	0	0
Checking Grammar	0	2	9	2
Checking Spelling	0	6	10	6
Creating Indent with Toolbar	1	1	4	0
Creating or Changing Styles in Dialog Box	4	5	4	2
Customizing Margins	4	4	1	2
Insert Merge Fields	8	8	3	0
Inserting Newspaper Columns in Dialog Box	5	5	4	0
Inserting Symbols or Special Characters	N/A	N/A	N/A	N/A
Inserting Table of Contents	4	4	1	0
Inserting Table with Toolbar	2	2	5	5
Inserting Word Art	5	6	7	2
Rearranging Topics in Outline Mode	0	2	3	4
Selecting a Printer	4	4	5	1
Sorting a Table	3	3	2	2
Turning On Revision Marks	2	2	2	1
Using Print Preview	0	0	1	0
Wrapping Text Around a Frame	2	2	0	0

Table 15: Supplemental Information Ratios for *Running Word*

Topic	Adjunct/Step	Elaboration/ Step	Total Extra/Step
Adding Borders to Paragraphs with Toolbar	1.00	0.00	1.00
Adding Header	N/A	N/A	N/A
Changing Alignment with Toolbar	1.50	0.00	1.50
Changing Line Spacing in Dialog Box	0.00	0.00	0.00
Checking Grammar	4.50	1.00	5.50
Checking Spelling	1.67	1.00	2.67
Creating Indent with Toolbar	4.00	0.00	4.00
Creating or Changing Styles in Dialog Box	0.80	0.40	1.20
Customizing Margins	0.25	0.50	0.75
Insert Merge Fields	0.38	0.00	0.38
Inserting Newspaper Columns in Dialog Box	0.80	0.00	0.80
Inserting Symbols or Special Characters	N/A	N/A	N/A
Inserting Table of Contents	0.25	0.00	0.25
Inserting Table with Toolbar	2.50	2.50	5.00
Inserting Word Art	1.17	0.33	1.50
Rearranging Topics in Outline Mode	1.50	2.00	3.50
Selecting a Printer	1.25	0.25	1.50
Sorting a Table	0.67	0.67	1.33
Turning On Revision Marks	1.00	0.50	1.50
Using Print Preview	N/A	N/A	N/A
Wrapping Text Around a Frame	0.00	0.00	0.00

Table 16: Summary Results for *ABCs of Word*

Topic	# Steps	Actual Steps	Adjunct Information	Elaboration
Adding Borders to Paragraphs with Toolbar	N/A	N/A	N/A	N/A
Adding Header	3	3	3	0
Changing Alignment with Toolbar	2	2	2	0
Changing Line Spacing in Dialog Box	6	6	6	0
Checking Grammar	3	3	13	2
Checking Spelling	3	2	15	5
Creating Indent with Toolbar	N/A	N/A	0	N/A
Creating or Changing Styles in Dialog Box	10	10	8	10
Customizing Margins	0	2	2	0
Insert Merge Fields	0	2	0	5
Inserting Newspaper Columns in Dialog Box	6	6	6	1
Inserting Symbols or Special Characters	6	5	13	1
Inserting Table of Contents	8	8	2	6
Inserting Table with Toolbar	0	3	1	1
Inserting Word Art	N/A	N/A	0	N/A
Rearranging Topics in Outline Mode	0	2	4	0
Selecting a Printer	N/A	N/A	0	N/A
Sorting a Table	9	9	4	5
Turning On Revision Marks	0	2	2	0
Using Print Preview	0	2	2	1
Wrapping Text Around a Frame	N/A	N/A	0	N/A

Table 17: Supplemental Information Ratios for *ABCs of Word*

Topic	Adjunct/ Step	Elaboration/ Step	Total Extra/Step
Adding Borders to Paragraphs with Toolbar	N/A	N/A	N/A
Adding Header	1.00	0.00	1.00
Changing Alignment with Toolbar	1.00	0.00	1.00
Changing Line Spacing in Dialog Box	1.00	0.00	1.00
Checking Grammar	4.33	0.67	5.00
Checking Spelling	7.50	2.50	10.00
Creating Indent with Toolbar	N/A	N/A	N/A
Creating or Changing Styles in Dialog Box	0.80	1.00	1.80
Customizing Margins	1.00	0.00	1.00
Insert Merge Fields	0.00	2.50	2.50
Inserting Newspaper Columns in Dialog Box	1.00	0.17	1.17
Inserting Symbols or Special Characters	2.60	0.20	2.80
Inserting Table of Contents	0.25	0.75	1.00
Inserting Table with Toolbar	0.33	0.33	0.67
Inserting Word Art	N/A	N/A	N/A
Rearranging Topics in Outline Mode	2.00	0.00	2.00
Selecting a Printer	N/A	N/A	N/A
Sorting a Table	0.44	0.56	1.00
Turning On Revision Marks	1.00	0.00	1.00
Using Print Preview	1.00	0.50	1.50
Wrapping Text Around a Frame	N/A	N/A	N/A

Table 18: Summary Results for *Word for Busy People*

Topic	# Steps	Actual Steps	Adjunct Information	Elaboration
Adding Borders to Paragraphs with Toolbar	0	3	1	3
Adding Header	0	2	3	8
Changing Alignment with Toolbar	0	2	1	4
Changing Line Spacing in Dialog Box	0	4	2	0
Checking Grammar	0	0	0	1
Checking Spelling	0	2	13	1
Creating Indent with Toolbar	0	1	0	4
Creating or Changing Styles in Dialog Box	4	5	7	5
Customizing Margins	0	4	1	1
Insert Merge Fields	0	2	2	4
Inserting Newspaper Columns in Dialog Box	0	3	9	1
Inserting Symbols or Special Characters	N/A	N/A	N/A	N/A
Inserting Table of Contents	0	5	2	0
Inserting Table with Toolbar	0	1	2	1
Inserting Word Art	N/A	N/A	N/A	N/A
Rearranging Topics in Outline Mode	0	3	6	1
Selecting a Printer	N/A	N/A	N/A	N/A
Sorting a Table	N/A	N/A	0	N/A
Turning On Revision Marks	0	3	2	1
Using Print Preview	N/A	N/A	N/A	N/A
Wrapping Text Around a Frame	0	3	8	0

Table 19: Supplemental Information Ratios for *Word for Busy People*

Topic	Adjunct/ Step	Elaboration/ Step	Total Extra/Step
Adding Borders to Paragraphs with Toolbar	0.33	1.00	1.33
Adding Header	1.50	4.00	5.50
Changing Alignment with Toolbar	0.50	2.00	2.50
Changing Line Spacing in Dialog Box	0.50	0.00	0.50
Checking Grammar	N/A	N/A	N/A
Checking Spelling	6.50	0.50	7.00
Creating Indent with Toolbar	0.00	4.00	4.00
Creating or Changing Styles in Dialog Box	1.40	1.00	2.40
Customizing Margins	0.25	0.25	0.50
Insert Merge Fields	1.00	2.00	3.00
Inserting Newspaper Columns in Dialog Box	3.00	0.33	3.33
Inserting Symbols or Special Characters	N/A	N/A	N/A
Inserting Table of Contents	0.40	0.00	0.40
Inserting Table with Toolbar	2.00	1.00	3.00
Inserting Word Art	N/A	N/A	N/A
Rearranging Topics in Outline Mode	2.00	0.33	2.33
Selecting a Printer	N/A	N/A	N/A
Sorting a Table	N/A	N/A	N/A
Turning On Revision Marks	0.67	0.33	1.00
Using Print Preview	N/A	N/A	N/A
Wrapping Text Around a Frame	2.67	0.00	2.67

Table 20: Summary Results for *Microsoft Word Made Easy*

Topic	# Steps	Actual Steps	Adjunct Information	Elaboration
Adding Borders to Paragraphs with Toolbar	0	2	2	8
Adding Header	0	4	4	0
Changing Alignment with Toolbar	0	1	2	3
Changing Line Spacing in Dialog Box	0	3	3	3
Checking Grammar	0	2	7	5
Checking Spelling	0	3	5	1
Creating Indent with Toolbar	0	2	3	1
Creating or Changing Styles in Dialog Box	0	7	2	7
Customizing Margins	0	4	2	0
Insert Merge Fields	N/A	N/A	N/A	N/A
Inserting Newspaper Columns in Dialog Box	0	3	1	2
Inserting Symbols or Special Characters	0	3	4	3
Inserting Table of Contents	0	5	5	2
Inserting Table with Toolbar	0	2	6	1
Inserting Word Art	0	10	4	2
Rearranging Topics in Outline Mode	0	2	0	0
Selecting a Printer	0	3	1	1
Sorting a Table	0	4	4	5
Turning On Revision Marks	0	3	2	0
Using Print Preview	0	3	5	0
Wrapping Text Around a Frame	0	1	1	1

Table 21: Supplemental Information Ratios for *Microsoft Word Made Easy*

Topic	Adjunct/ Step	Elaboration/ Step	Total Extra/Step
Adding Borders to Paragraphs with Toolbar	1.00	4.00	5.00
Adding Header	1.00	0.00	1.00
Changing Alignment with Toolbar	2.00	3.00	5.00
Changing Line Spacing in Dialog Box	1.00	1.00	2.00
Checking Grammar	3.50	2.50	6.00
Checking Spelling	1.67	0.33	2.00
Creating Indent with Toolbar	1.50	0.50	2.00
Creating or Changing Styles in Dialog Box	0.29	1.00	1.29
Customizing Margins	0.50	0.00	0.50
Insert Merge Fields	N/A	N/A	N/A
Inserting Newspaper Columns in Dialog Box	0.33	0.67	1.00
Inserting Symbols or Special Characters	1.33	1.00	2.33
Inserting Table of Contents	1.00	0.40	1.40
Inserting Table with Toolbar	3.00	0.50	3.50
Inserting Word Art	0.40	0.20	0.60
Rearranging Topics in Outline Mode	0.00	0.00	0.00
Selecting a Printer	0.33	0.33	0.67
Sorting a Table	1.00	1.25	2.25
Turning On Revision Marks	0.67	0.00	0.67
Using Print Preview	1.67	0.00	1.67
Wrapping Text Around a Frame	1.00	1.00	2.00

VITA

Eric J. Ray

Candidate for the Degree of

Master of Arts

Thesis: PROVIDING APPROPRIATE INFORMATION: AN ANALYSIS OF PROCEDURES IN COMPUTER USER DOCUMENTATION

Major Field: English

Biographical:

Education: Graduated from Stillwater High School, Stillwater, Oklahoma, in May, 1986; received Artium Baccalaeuri degree in German and Secondary Education from Washington University, St. Louis, Missouri, in May, 1990. Attended Universität Regensburg, Germany, on Deutsche Akademische Austauschdienst fellowship from 1990 to 1991. Completed the requirements for the Master of Arts degree with a major in English (Technical Communication) at Oklahoma State University in December 1996.

Experience: Employed as a technical communicator by Atlatl Communications, Perry, Oklahoma, 1991. Independent consultant for German translation and technical communication and technical communicator for Oklahoma State University, Computing and Information Services, 1992 to 1995. Telecommunications Development Analyst for LDDS WorldCom, Inc., 1995-1996. Independent technical communications consultant, author of computer books, and co-owner, RayComm, Inc., 1996 through present.

Professional Membership: Society for Technical Communication.