

1991

The Rhetoric of Documentation: Two Approaches

Walt Wisnewski

Eastern Illinois University

This research is a product of the graduate program in [English](#) at Eastern Illinois University. [Find out more](#) about the program.

Recommended Citation

Wisnewski, Walt, "The Rhetoric of Documentation: Two Approaches" (1991). *Masters Theses*. 2234.
<https://thekeep.eiu.edu/theses/2234>

This is brought to you for free and open access by the Student Theses & Publications at The Keep. It has been accepted for inclusion in Masters Theses by an authorized administrator of The Keep. For more information, please contact tabruns@eiu.edu.

THESIS REPRODUCTION CERTIFICATE

TO: Graduate Degree Candidates who have written formal theses.

SUBJECT: Permission to reproduce theses.

The University Library is receiving a number of requests from other institutions asking permission to reproduce dissertations for inclusion in their library holdings. Although no copyright laws are involved, we feel that professional courtesy demands that permission be obtained from the author before we allow theses to be copied.

Please sign one of the following statements:

Booth Library of Eastern Illinois University has my permission to lend my thesis to a reputable college or university for the purpose of copying it for inclusion in that institution's library or research holdings.

23 Aug - 91

Date

I respectfully request Booth Library of Eastern Illinois University not allow my thesis be reproduced because _____

Date

Author

The Rhetoric of Documentation: Two Approaches

(TITLE)

BY
Walt Wisnewski

THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

Master of Arts in English

IN THE GRADUATE SCHOOL, EASTERN ILLINOIS UNIVERSITY
CHARLESTON, ILLINOIS

1991
YEAR

I HEREBY RECOMMEND THIS THESIS BE ACCEPTED AS FULFILLING
THIS PART OF THE GRADUATE DEGREE CITED ABOVE

8-23-91
DATE

7/23/91
DATE

Abstract

In an attempt to clarify how a beginner who has not yet mastered a software program is to learn to use word processing programs, this thesis examines two examples, WordPerfect and W.W. Norton's TEXTRA from the standpoint of their documentation. Software developers have adopted conflicting rhetorical strategies in their documentation: some have sought clear instruction via a rule-based rhetoric with a reliance on jargon-free standard English, and others have pursued an inferential rhetoric. These two strategies parallel two models in modern communications theory: a decoding model and an inferential model. WordPerfect seems to follow a decoding (rule-based) approach and TEXTRA seems to follow an inferential approach. An examination of the rhetorical environment of the documentation is a useful strategy for determining the complexity of the program and the complexity of learning how to use it.

Acknowledgements

Frank C. Stokes, who wrote the editorial comment, "Spit it out" . It was good advice delivered just in time.

Donald Dolton. and Robert W. Funk who responded rapidly to requests for comments.

The Ladner family of Lawrenceville, Illinois. Frank and Julia Ladner provided a place to work. Rita Ladner provided study materials and advice.

Anne Zahlan, graduate advisor, who gave guidance through the length of the graduate program.

TABLE OF CONTENTS

INTRODUCTION	2
BACKGROUND THEORY	8
TWO APPROACHES TO THE AUDIENCE	15
COMPARISONS	18
Program Structure	18
Documentation structure	21
Templates	25
Comparison by function	26
Initial screens	26
Editing screens	28
Help Screens	29
Default Settings	31
Entering Text	32
Moving the Cursor	34
Saving the Text	35
Printing the Text	37
CONCLUSIONS	39
Works Cited	42
Appendix A	43
Appendix B	44
Appendix C	45
Appendix D	46
Appendix E	47

THE RHETORIC OF DOCUMENTATION: TWO APPROACHES

INTRODUCTION

One has only to walk into a college library to see the degree to which computer technology has entered the learning process. The card catalog has been replaced by computer terminals. The Modern Language Association (MLA) index now comes on a computer disk, as do a dozen other indexes of specialized publications. All of these forms of bibliographic information fall into the category of data bases, and are the most visible form of computer usage. Another highly visible and even more common use of computers on college campuses is as word processors. For a student who must learn to express ideas on paper, mastering the use of a word processor is as important to academic success as mastering the use of the electronic card catalog. This need to master the new technology leads to the focus of this thesis, which is to compare the documentation in two word processors, to examine how a student might acquire a facility with this form of writing.

Where data bases tend to rely upon large scale computers (frequently referred to as "main frames") which can service many users at remote terminals simultaneously, word processors usually use desk-top, or personal, computers. Writing laboratories have sprung up on campuses, usually

equipped with personal computers using a variety of the many forms of commercial word processors now on the market. Some universities now equip dorm rooms and faculty offices with personal computers, primarily because they are flexible enough to run a variety of commercially available programs and also able to communicate with the larger main frames.

Obviously, success in an environment which uses this technology depends on finding a convenient way to learn how to apply it. In the case of, for example, a library data base (the technological equivalent of an on-line card file), a student need learn only a few commands to direct a search of the data base, making it relatively easy to use. The student also has access to the librarians, many of whom are trained experts in the system and who can act as guides or instructors.

In contrast, mastering a word processor on a personal computer forces the student to rely on the software producer's documentation, and it is here that a problem begins. Even a relatively basic word processing program such as W.W. Norton's TEXTRA has 53 functions for the student to master, and WordPerfect, a popular full-featured program, has 137. The user must also have mastered the concept of "software" and the operating system (such as DOS) through which the

word processor operates. But program complexity alone is not the major problem, nor is the need to learn new concepts. The problem is the documentation itself. As R. John Brockman points out, "Inadequate software documentation for users is a widely acknowledged problem in the computer industry" (1). He seconds the opinion of Peter McWilliams, whom he quotes from The Personal Computer in Business Book: "The weakest link in the world of small computers is the documentation" (14).

Software documentation takes a variety of forms, including instructional manuals, tutorials or practice exercises, reference cards which offer mnemonic clues to the available functions, and templates to map certain keyboard keys to specific functions. Tutorials and abbreviated instructions may also be incorporated into the program for display on the screen, as "on-line help," ostensibly reducing the need to turn away from the screen and the writing task to read the manual. Recent trends indicate an increased use of this form of on-line help.

Examples of tangled syntax, faulty organization or confusing layout in software documentation abound. As a sample of poor organization, the authors of the Handbook for Developing Computer User Manuals cite the example of "a

seventy-nine page manual that on page 69 announces 'the first thing you will have to do before performing anything...'" (Adams *et al.* 9).

The most common complaint about software documentation, however, is the overuse of jargon (Brockman 153; Houghton-Alico 49; Price 6). Brockman provides a definition and illustration of jargon:

Jargon can be defined as words unfamiliar to the intended audience. There is rarely a jargon word that is jargon to all audiences. For instance, an 'objective correlative' is perfectly meaningful to a literature major who specialized in T.S. Eliot, but is incomprehensible to anyone else. By the same token a 'MUX' would be totally meaningless to a literature major (153).

Doann Houghton-Alico offers as one example of flawed rhetoric this definition of the computer command "XXX" found in a user manual:

XXX: The screen on which the application should define its windows (48).

The language of the example is nonsense on its surface and is doubly abstruse because it uses common words (window, screen, application) as jargon with private definitions which are unique, not just to the computer industry, but even more narrowly, to the company which wrote the software and the accompanying documentation from which the example was taken. Houghton-Alico uses this example as a reason for her rule to "Avoid Jargon" (50).

Charles Sides, however, defends the use of jargon. "Good documentation uses jargon," he says (5). He would argue that the definition provided by Houghton-Alico is not understandable because the writer failed to identify the audience as being outside the knowledgeable group. For the knowledgeable (the "in") group, jargon serves as a form of shorthand for communicating¹⁴. To Sides, it makes no sense to alter terminology if the readers share the knowledge of the writer. There is an advantage in using jargon within a knowledgeable group simply because the jargon provides context as well as fact. There is disadvantage, however, when the jargon disguises information to those not members of the "in group."

We can see, therefore, that texts on writing software vary widely in their separate approaches; some taking a

rule-based approach, others taking a procedural or inferential approach, with the two approaches in strenuous disagreement. When Houghton-Alico advocates the rule of avoiding jargon, she is in effect advising the writer to remain in the environment of general English and to use words with precision; Sides, however, accepts the use of jargon if the writer can establish a common "cognitive environment" or can identify what the writer and the audience share. These two approaches parallel a current theory in communication of two models of understanding, one called a code (or decoding) model, the other an inferential model.

A heavy reliance on the decoding model leads to the use of rule-based software manuals, and reliance on the inferential model leads to a more holistic treatment. As an example of the rule-based approach, we can examine the documentation for the word processing program built around a reference manual, WordPerfect; for the holistic approach, we can look at W.W. Norton's TEXTRA.

BACKGROUND THEORY

A long-standing dispute continues over the location and construction of meaning in communications. The conflict in rhetorical style for documentation is a part of this overall disagreement over meaning that has gone on in the field of literature and communication. "From Aristotle through modern semiotics, all theories of communication were based on a single model, which we call the code model" (Sperber and Wilson 697). In 1926, however, the literary critic I. A. Richards proposed that language has two uses: the Scientific (*sic*) and the emotive. Within the "emotive" language there was ambiguity, but within the "Scientific" language, he saw the possibility for clarity. "... so far as any body of references is undistorted it belongs to Science" (Lodge 111). But where Richards sought to look at a piece of literature as an organic "whole" isolated from the author, a modern communications theory adopts a different perspective, shifting the focus from the medium to the participants in communication: treating communication as a cooperative act between the speaker (or writer) and the audience.

In a 1967 Harvard University lecture, H. P. Grice provided a long list of conversational exchanges to illus-

trate the difference between the words used and the intended implication of the message. In a later article, "Logic and Conversation," based on the lectures, Grice offered the idea of conversation as a cooperative act; a "quasi-contractual matter" (48). This contract is based on a Cooperative Principle supported by nine maxims -- grouped under the headings of Quantity (do not say what you believe is false), Quality (be relevant), Relation (say it understandably within that context) and Manner (be perspicuous) -- by which meanings beyond the words are available.

Dan Sperber and Deirdre Wilson, in a precis of their book, Relevance: Communication and Cognition, describe the method by which people communicate, once a common environment is established. In a linguistic approach to communication, Sperber and Wilson describe the process by which people who "share a cognitive environment" can make assumptions about, for instance, the characterization of others who have access to that environment. They define a cognitive environment for an individual as "a set of facts that are manifest," that is, perceptible or inferable to the individual (699). This would seem to be the basis for jargon being understandable to the "in group." The documentation writer who can identify with the reader's experience or who can establish that environment is better able to

communicate new information to the reader (Sides 7).

Sperber and Wilson also describe two internal information processing devices by which "linguistic utterances" are rendered usable: a decoding model and an inferential model (698). The decoding model is "a system which pairs internal messages with external signals, thus enabling two information processing systems (organisms or machines) to communicate" (697). The inferential process "takes a set of premises as input and yields as output a set of conclusions which follow logically from, or are at least warranted by, the premises" (698). Inference is a process based on the accumulated set of assumptions already in the mind.

In Relevance: Communication and Cognition, Sperber and Wilson offer an anecdotal example of the power of communication:

Peter and Mary are sitting on a park bench. He points in a direction where she had not so far noticed anything in particular. This time she takes a closer look and sees their acquaintance Julius in the distance, sitting on the grass. In other words, as a result of Peter's behavior, the presence of Julius, which was weakly manifest in Mary's cognitive environment, has become more

manifest, to the point of being actually noticed. Moreover, it has become manifest that Peter himself noticed Julius and intended her to notice him too. (700)

Although the example contains not a word of spoken or written dialogue between Peter and Mary, it illustrates a number of principles about human communication which the authors propose. First, the example illustrates that Peter and Mary are acquaintances, not necessarily intimate, and therefore will invest some measure of cooperation in the task of communications. This cooperation seems a necessary ingredient in successful communications and has been addressed as a separate subject by H. P. Grice, upon whose work Sperber and Wilson build. Second, the example illustrates the notion that Julius, because he is known to both, represents a mutual "assumption" which each can accept. The assumption is not based on their equal knowledge of Julius, but that he is known in some way to each. Third, the example illustrates that their joint presence in the park provides a context, a subset of Mary's assumptions about the world of which Peter shares some portion. Fourth, the example illustrates that Peter, by making the gesture, implies an attempt to communicate the message about Julius's presence as well as the message that he intends to communicate some informa-

tion to Mary. This act of drawing attention is called ostention.

This example from Sperber and Wilson illustrates that communication is possible even without words. Spoken language has some advantage over written language, of course. A speaker can add information to the words themselves by the ostensive devices of gesture, expression and inflection. Within this spare example of Peter and Mary in the park are principles that can be extended to spoken and written words, and therefore to writing software documentation. In particular, it draws attention to the ways by which meaning is conveyed without words, and emphasizes that communication conveys meaning in two ways, "by means of a code shared by the communicator and his audience or by means of ostensive stimuli providing the audience with evidence from which the communicator's informative intention can be inferred" (Sperber and Wilson 701).

In writing, the emphasis in the communication of an idea is on written words, their connotation and denotation and on their arrangement on the page (or on a text screen) and the rhythms they create in the mind. But even in writing, methods of communication beyond the decoding of words are also possible. Sperber and Wilson cite as examples the

literary use of the "poetic effects" of irony, metaphor, meiosis, and hyperbole which, they say, "create common impressions rather than common knowledge" (707). To Grice these poetic effects are explainable because they are blatant attempts to flout the maxims of Quantity or Quality, opening the path to other interpretations (53). The converse of this, and the warning to documentation writers, is that the other participant in communications, the reader, may apply other interpretations to the words because of the context, or rhetorical environment.

A software documentation writer has some advantage over a fiction writer in that the reader has come, cooperatively, to the park bench looking for information and is willing to invest some effort in the communication. The reader will, according to Sperber and Wilson make economical assumptions leading to "maximizing the relevance of the information processed" (700).

As can be seen from the emphasis on the cognitive environment, it is to the advantage of the software documentation writer to understand this environment in which the audience has pre-formed assumptions and to know what part of the environment he shares with that audience. Many writers place emphasis on the need to know the audience (Brockman

Wisnewski--Rhetoric of Documentation

58; Price 32). Brockman goes so far as to say that the audience must be "the absolute dictator of documentation style" (110).

TWO APPROACHES TO THE AUDIENCE

It may seem impossible to address secretaries, students, writers, and teachers as sharing a common cognitive environment: it may seem too broad an audience, but the software producer and the documentation writer control the immediate rhetorical environment of the software user. They do this by designing the page and screen layout, by selecting the language they use, by arranging the reference materials and mnemonic aids in an accessible fashion (Brockman, 48-59). The primary importance in communicating to a beginner how to use a software program is the whole rhetorical environment of computer screen, instructional manual, and the tutorials together with the ostensive effects of the layout of the material, the accessibility of the commands, and the ease with which instructional material may be found. These constitute the "park" in which the documentation writer meets the user.

The disagreement over the use of jargon may be simply a distraction: those who eschew it identify themselves as favoring a rule-based approach to documentation writing. This rule-based approach may include the strict adherence to

general English, reliance on procedural steps beginning with imperative verbs, fixed format for each command, fixed conventions for numbering steps in a procedure, and limits to the number of steps in any single procedure. Other rules are dependent on the documentation designer. The rule-based approach is equivalent to the decoding model of Sperber and Wilson, and relies on close control of the ostensive effects in order that meaning would be conveyed by the words themselves. The equivalent of the inferential model in documentation is the approach that, among other criteria, uses ostensive effects to provide additional information, uses a conversational language, groups common ideas into a paragraph structure, and "recognizes that readers bring generic expectations of text organization to the task of reading (our) manuals" (Brockman 65). In examining the functions and the documentation of WordPerfect and Norton's TEXTRA we can see how two major producers of software documentation approach the task in two different ways.

The other participant in this act of communication is the writer using the word processor. For the purpose of comparing the two approaches to documentation, we will consider how a college-level composition student must learn to use each program, given the integrated information package of manual, tutorial, on-line help, etc., provided by the

documentation writer. The student writer must first learn some of the documentation writer's jargon, including new terms for traditional ideas. What the student thinks of as a "piece," an "essay," or a "story," for instance, will be a "file" or "document." Textual "phrases" and "paragraphs" become "blocks" of text. Writing itself becomes "entering text" or "editing a file."

In examining the two manuals, it seems that the student approaching either program is expected to have acquired a basic vocabulary of computer jargon. He should be able to apply such nouns as: disk, hard disk, drive, file, cursor, arrow keys, function keys, home, and end, and such verbs as: format, retrieve, enter, and delete. These words are used, without explanation, in both manuals, as though they had already been absorbed into general English. They would have been introduced in the computer manufacturer's instructions on setting up the machine. WordPerfect introduces many of these phrases in the tutorial workbook, implying that the workbook is the proper beginning point for a novice.

COMPARISONS

Because WordPerfect differs so greatly from TEXTRA in complexity, the two cannot be compared point-by-point. WordPerfect includes not only word processing functions but also many more printing and graphics features. Norton's TEXTRA contains only word processing functions. The program structure is described below since it too is a part of the rhetorical environment, and places limits on what the documentation must say and do. The documentation which describes the use of the program will then be compared only at the level where they overlap. The comparisons will be made of the documentation only in terms of the basic functions of word processing: creating, editing, storing, and printing (in a basic form) the text.

Program Structure

The structure of any program, its commands, screen presentations, the on-line help offered, the printed documentation and tutorials constitute a rhetorical environment in which the student carries on a dialogue with the computer while also trying to construct a piece of text.

Both TEXTRA and WordPerfect use keystroke sequences (one key at a time; [F1], for example) and combinations (two keys pressed at the same time; [ALT-H], for example) as commands to the program as to how to handle the text. A list of the function keys with their names appears in Appendix A. The repertoire of available commands constitute a language by which the program and the user communicate. At best it is a difficult dialogue. While the user is capable of both inferential and decoding modes of understanding, the computer is capable of only decoding the commands given it.

TEXTRA

A repertoire of 53 commands.

Commands controlled by function keys, alone and in combination with Shift key [SHIFT] and Alternative key [ALT]; 23 commands derived from Control key [CTRL] combined with letter keys.

WordPerfect

A repertoire of 137 commands.

Commands controlled by function keys used alone and in combination with Shift key [SHIFT], the Alternative key [ALT], and Control key [CTRL] followed by subsequent numerical choices.

Discussion. In general, a keystroke or keystroke combination selected from the repertoire takes the user to a set of choices which make the initial choice more specific. In TEXTRA, for instance, selecting function key [F3] (defined as "New Format") while creating or editing the main document evokes a menu of seven possible page formats or the

chance to alter (edit) any of them. Choosing one of the formats by selecting a second function key will cause the text following the cursor to use that format. In effect, the key stroke sequence [F3], [F2] will switch from main document line spacing, paragraph indentation and line width to an indented quote format which will then continue in effect until the keystroke combination [F3], [F1] restores the main document format. The line format in use is displayed graphically on the screen, as a "ruler" with margins, paragraph indentation and tabs indicated.

To create the same effect in WordPerfect, the three-keystroke sequence [Shift-F8], [1], [7] will bring the student to the point at which the margin entries can be entered. In general, keystroke sequences in the more complex program, WordPerfect, will be longer than those in TEXTRA. In the above sequence, [Shift-F8] (defined as Format), leads to the four numbered choices of Line, Page, Document, and Other. Selecting Line [1] leads to another screen list of nine choices of line format including Hyphenation, Justification, and Margins, among others. Selecting Margins [7], allows entry of the column numbers which will be the left and right margins. No ruler is displayed.

As can be seen from the above, the keystroke sequence

in WordPerfect is not just one stroke longer; at each selection screen, the student finds considerable information to sort through to find the desired function. The student must also shift attention from composition to searching through a larger mass of instruction on what to do, whether that information is on the screen or in the manual. The WordPerfect repertoire of 137 commands was derived by counting the entries in the Table of Contents of the Reference section of the manual (49-52). By taking other permutations of keystrokes, other totals are possible.

As is frequently the case in WordPerfect, the sequential keystroke procedure to accomplish a task is supplemented with a single command. The process of indenting the margins described above can be accomplished by the command [SHIFT-F4] to indent both margins by one tab stop, or with the command [F4] to indent the left margin by one tab stop.

Documentation structure

Documentation becomes the medium through which the student learns to use the program. It constitutes the major part of the rhetorical environment in which the program is used.

TEXTRA

Documentation consists of a 142-page printed manual plus the on-line help screens and an on-line handbook of English.

WordPerfect

Documentation consists of a 563-page printed manual, plus a 398-page tutorial workbook, a reference card which list commands and a template which fits around the function keys as a memory aid (see below).

The on-line Help, invoked by entering [F3], is described below.

An on-line tutorial supplements the printed workbook.

Discussion. The contents of the TEXTRA manual are arranged by the functions of getting started, creating, editing, formatting, printing, and saving the text. Three chapters are devoted to editing procedures. It proved useful to read these chapters over before beginning to enter or edit text. The tone of the manual is nearly conversational, following the inferential model. The student is told what to expect with certain actions, such as:

When you press [F1] the Edit menu will be replaced by the Shift menu.

And

If you press [ESC] at this point, the prompt will disappear (32).

To locate the information for format selection given

above, the student has two basic choices for finding the information, as well as the option of experimentation. An adept (experienced) program user could rely on personal memory for the keystroke sequence [F3], [F1], but the student writer must look for clues in the immediate environment, which consists of the on-line help or the manual. To experiment, the student could see at the foot of his editing screen a list of ten instructions including [F3] New Format, and simply invoke the command to see what result might occur. The student could also look in the manual where the Table of Contents or the index entry for "Formatting" would lead to Chapter Seven.

In WordPerfect documentation is voluminous, and follows the rule-based model. The manual includes chapters on Installation, Getting Started and a Reference section. The 374-page Reference section is arranged as an encyclopedia of features: Cancel, Center Page, Concordance, etc. Each feature description uses a prefatory sentence or paragraph to describe the purpose, a numbered list of procedural steps, and a final sentence or paragraph of explanation or warning. Procedural steps all begin with an imperative verb (Move, Select, Enter, etc.). The Reference section becomes the most used source of information about the commands. With 137 commands to name, however, some of the names are

less than meaningful to a beginning user. The command Mark Text [ALT-F5], for instance, leads to a feature which allows the merging of several documents into one, a feature not apparent in the name. Nor does the feature named Screen [CTRL F3] imply that it can be used to draw lines around text or enclose illustrations. The Reference section has its own Table of Contents arranged as an alphabetical list of the features. The tone of the manual is more imperative than TEXTRA:

Press -> Search to begin the search.

And

Type the characters and/or codes you want to find
(311).

An on-line tutorial is available as an alternative to the Fundamentals section of the workbook. Completing the on-line tutorial or, at minimum, the first 12 to 15 lessons in the workbook is an important first step. Either lead the student through the commands keystroke-by-keystroke, with extensive explanation.

To locate information for the format selection described above, the student using WordPerfect has the same three options described for TEXTRA, except that experimentation is limited because there is no on-screen listing of

basic commands, as In TEXTRA. The reference section Table of contents yields an entry for Format, and the index yields a variety of entries in which format is a key word.

Templates. Each program provides a listing of the features or functions associated with the function keys. In TEXTRA the listing is given at the foot of the screen as a two line menu. In WordPerfect the listing is given on a cardboard template which fits around the function keys.

TEXTRA

In lieu of a template to list the 53 functions, two lines at the bottom of the edit screen list the ten functions directly accessible.

WordPerfect

WordPerfect provides a cardboard template which fits over the function keys of the two IBM style keyboards. The template lists the forty initial choices offered to accomplish the desired task.

Pressing the Help key [F3] twice will evoke a screen display of the template.

Discussion. In TEXTRA, pressing function key [F1], labeled "more...", in the on-screen menu will change this two line menu to the ten functions accessible with the key combination [SHIFT-Fx]. Selecting [F1] again will change the display to the [ALT-Fx] menu. In effect, a function can be accessed in two ways. The student can cycle the menu until the function is on the screen and press the function key. The alternative is to use the combination [SHIFT-Fx]

or [ALT-Fx]. This alternative requires some familiarity with the program. The arrangement of the functions, however, follows a logical pattern, as can be seen from the listing in Appendix A. The functions accessible with the function key alone are all related to text manipulation: [F2] to insert a line, [F3] to choose a new line format, [F4] to highlight the text, etc. Functions accessible with the combination [SHIFT-Fx] relate to manipulation of larger blocks of text: [SHIFT-F5] to copy a block, [SHIFT-F8] to modify the page layout, etc. The [ALT-Fx] functions are related to ancillary functions: bibliography, endnotes, word count, and access to DOS commands.

In WordPerfect, the template lists the forty functions directly accessible. Hunting through a list of forty features can be slow until the student gains familiarity. As seen in the Appendix A listing, some patterns can be seen in the arrangement. The search and replace commands use [F2] alone or as modified by the [SHIFT], [ALT], or [CTRL] keys. An analogy can be made that the programming function defined by the function key is modified by the adjectives inherent in the [SHIFT], [ALT], or [CTRL] keys.

Comparison by function

Initial screens. Both programs begin by flashing briefly on

the screen the company logo (TEXTRA) or copyright notice (WordPerfect). After a few seconds this fades.

TEXTRA

After the logo screen fades, a choice screen appears on which the student faces four choices:

1. Retrieve a document
2. Create a new document
3. Information on getting started
4. Exit from the program.

WordPerfect

After the copyright screen fades, a nearly blank editing screen appears. It is described below.

A line at the bottom of the screen advises "Press [Alt-H] anytime you need Help."

No Help is offered; prior knowledge of how to access on-line Help is required.

Discussion. That the TEXTRA opening screen offers "information" as an option is less threatening to a beginner. A line at the bottom of the screen advises "Press [ALT H] anytime you need some help." Selecting Information leads to another Help screen. This screen and the choices are explained both on page 16 of the manual and by a Help screen accessed with [ALT H]. In effect, there area variety of ostensive effects which offer clues to the student as to what to do.

Frequent complaints have been voiced about the introduction to WordPerfect. The blank screen can be disconcerting the first time the user initiates the program, even after has reading the "Getting Started" section of the manu-

al, because there are no visible cues as to what to do next. The manual advises, "Upon entering WordPerfect, you can start typing" (34), but if the student types text onto the screen, he or she must then learn what to do with it. This requires prior knowledge of the documentation writer's assumptions by reference to the workbook or the Reference section of the manual.

Editing screens. On the editing screens a student would enter, manipulate, and correct text.

TEXTRA

At the top of the editing screen, and information line displays the current file name and the page, line and column position of the cursor. Below this is a "ruler" line, a graphic representation of the line format, with margins, tabs, and paragraph indentation. At the bottom of the screen, an information bar gives the name of the current screen, and the reminders of the keys for exiting [ESC] and for on-line Help [ALT-H]. Below this is a two line menu of the function key features.

WordPerfect

Immediately after the initial screen fades, the program presents the nearly blank editing screen. The cursor blinks in the upper left corner and a line of letters in the lower right indicate: "Doc 1 Pg 1 ln 1" Pos 1."

Discussion. These screens represent the strongest contrast between the two systems. Following Sperber's and

Wilson's analogy, the WordPerfect "park" is barren. TEXTRA, by contrast, provides reference cues in the form of the ruler and the three lines of information. The TEXTRA editing screen holds four fewer lines of text than WordPerfect; 20 compared to 24, or a decrease of 17%. To what extent this affects the student's overall view of the theme is not apparent, but it would seem that the bordering information would more important to a student than an additional four lines of text. WordPerfect's default cursor position gives page and line position of the cursor in inches. To change them to columns and line numbers requires changing the Setup conditions.

Help Screens. A key part of the cognitive environment of the two programs is the on-line documentation offered to the student.

TEXTRA	WordPerfect
The Help option is evoked by pressing [ALT-H]	The Help option is evoked by pressing [F3].

Discussion. The TEXTRA key combination [ALT-H] leads to a menu of choices of the five types of help available. The reminder of this available Help is always displayed on the screen. The menu choices include an index of help topics, a keystroke summary of commands, the cursor movement

commands, access to film-strip type tutorials, and a choice called "How do I...?" which also accesses film strips by the action desired. The most obvious form of Help, however, is the on-screen listing of commands and the function keys which invoke them. The [F1] key is labelled "more..." and causes the next group of ten commands to be displayed.

On the WordPerfect Help screen can be seen the one sly bit of humor in the documentation; it contains the instruction, "Press ENTER to exit Help." The first screen of on-line Help gives two Help choices. One tells the student to press a letter key to get a listing of features which begin with that letter; in effect, an index of information available, similar to the Table of Contents in the Reference section of the manual, or the Index in the back of the manual. The on-screen index entries overlap, but differ from, either the Table of Contents or the Index. The screen index also serves as a mnemonic similar to the reference card since the listing gives the keystroke sequence to call up that feature. The other Help screen choice is to press a function key to get a screen of information about that key. In each program a minimum of two keystrokes is required to call up a screen of information about a key or a function.

When the student accesses the on-line Help, the text is removed from the screen to be replaced with the sought-after information, and is therefore as much an interruption to the student's thought process as turning to a manual. The TEXTRA opening screen choice of Information on Getting Started makes it possible to use the program without relying on the printed manual.

Default Settings. The character of the display, the type and amount of information shown on the screen, the printer and its fonts, editing characteristics and other controls on the system are pre-set by the software designer, but may be altered by the user. The degree of ease in finding the switches for the settings varies between the two systems as follows:

TEXTRA
Default settings for the margins and page sizes are listed in the section on "page layout" (93).

Defaults can be altered at the cursor position or permanently.

Wordperfect
Default settings for the page size are illustrated in the Getting Started section of the manual (34) and in the on-line tutorial.

Defaults can be altered at the cursor position or permanently.

Discussion. In both programs, the default settings can be altered temporarily (at the cursor position, or for the current document only) or permanently. Temporary changes

are accessible through format control. In TEXTRA this is covered in a manual chapter. The information bar at the bottom of the screen lists New Format [F3]. An experimenter could make that choice, which leads to the command sequence for changing page size, line spacing, indentation, and type styles in the current document. In WordPerfect, the Format [SHIFT-F8] entry in the Reference section of the manual lists the format options and sends the reader to an entry for each option. Permanent changes in defaults in TEXTRA are contained in the Customize entry under the [ALT] menu. In WordPerfect, permanent changes are made under the Setup feature [SHIFT-F1]. The Setup entry also sends the reader to entries for the particular options to be changed.

Each of the programs provides the user with a screen display which represents, in a general way, how the text will lay on the page. In the case of WordPerfect, the screen display is only an approximation, and valid only if the default print font matches the screen font.

Entering Text. Using the MLA approved format for a college theme, the student would enter a heading in the upper left corner of the page in a stacked array, followed by the text of the theme. A novice user would discover the information for entering this initial text for each program as follows:

TEXTRA

Chapter three of the manual (31-42) describes the basics of creating and editing text, cursor movement and moving among the menus.

WordPerfect

The Getting Started section of the manual (21-46) describes the keyboard, the function keys, access to on-line Help and the default setting for margins and spacing. According to the manual, "Starting WordPerfect is like rolling a clean sheet of paper into a typewriter with margins and spacing already set" (33).

Discussion. Intuitively, a novice on either program would take the approach of treating the keyboard as a typewriter, entering text and possibly entering the equivalent of a carriage return ([Enter]) at the end of each line. Interestingly, neither manual introduces the phrase or idea of "wrap around text," nor instructs the student to use [ENTER] only at the end of a paragraph. This emphasizes the need for some prior assumptions required of the student. The information, however, can be found in the WordPerfect workbook tutorials. For a more complete introduction to entering text, the student should refer to the WordPerfect workbook. The first two chapters introduce the basic jargon terms and give extensive explanation of basic procedures.

Editing existing text (deleting, inserting, or formatting) in TEXTRA is relatively straightforward. In WordPer-

fect it is unsafe to edit without turning on the feature named Reveal Codes [ALT-F3]. This feature uses the lower half of the screen to repeat the text in the upper half, but with the hidden codes shown. These codes control the appearance of the text on the screen or printed page, and are inserted into the text, sometimes in pairs, when any of the features are invoked by pressing a function key. Because they are normally hidden, it is easy to erase them or one of a pair of them, leading to considerable confusion in the editing process.

Moving the Cursor. In word processing programs, moving the cursor is basic to operation of the program. The cursor is a pointer which marks the position where text is entered and changes are to be made, and is used to mark the beginning and end points of blocks of text. The speed and accuracy with which a student can move the cursor through text influences the overall rate of speed of composition.

TEXTRA

Cursor movement is controlled by the Arrow keys, Page Up, Page Down, Home, and End keys and modified by the Control [CTRL] and Shift [SHIFT] keys.

Cursor movement is described in a section of Chapter 3 of the manual (34-37).

WordPerfect

Cursor movement is controlled by the arrow keys, Page Up, Page Down, Home and END keys, and modified by the Escape [ESC] and Control [CTRL] keys.

"Cursor movement" is an entry in the Reference section Table of Contents and "Cursor move" is an Index entry; both lead to page 99.

Discussion. TEXTRA offers more complexity in cursor movement, since various key combinations will allow moving the cursor a word, sentence, paragraph, or page at a time. The sequence [Shift-Left Arrow], for instance, will move the cursor to the beginning of previous sentences. To a writer accustomed to thinking of text in these terms, this would seem to act as reference for memorizing the keys involved. The two programs use the same key sequence ([CTRL-Left Arrow] or [CTRL-Right arrow]) to move the cursor to the previous or following word.

Saving the Text. Text entered on the screen is ephemeral.

It is held in the short-term memory of the computer where it will disappear if the power is turned off. It must be transcribed to a permanent medium, such as disk or tape, if

it is to be re-edited or re-used. The text must also be given a unique name so that it can be retrieved.

TEXTRA	WordPerfect
Two commands for saving text are provided.	Two commands for saving text are provided.
A brief chapter is devoted to the options available with the Save commands (111-115).	The information on saving text is dispersed.
Name verification is required.	Name verification is required.

Discussion. Both programs offer optional methods of saving the text. The student can save the entire document, portions of it, save it under a new name, save it automatically at timed intervals while editing, save it in a formatted (with page layout codes in place) or unformatted mode. A recent version of TEXTRA allows saving the file in WordPerfect format. This would permit separating the basic function of writing from the confusion of editing. That is, a student could write in TEXTRA, and save the file on disk in WordPerfect format for editing in WordPerfect, allowing insertion of special features, fonts, and graphics.

In TEXTRA the student may save the text with the [SHIFT-F10] entry or by selecting [S] at the main menu. Each one invokes the same sequence of name verification.

The TEXTRA manual gathers this information in the brief chapter nine (111-115).

In WordPerfect, the basic Save command, [F10], is listed on the template. Saving the text is also a part of the Exit sequence invoked by [F7]. Information from the on-line Help defines the action of [F10], and refers the student to other entries for Long Names, Exit, and the option of Fast Save. Nine index entries in the WordPerfect manual lead to the optional methods of saving the text. The reference manual entry for Save does not contain any mention of Long Names, but covers the other options. Under the Index entry Filing, the manual provides basic information about the options available and leads the student to cross references.

Printing the Text. With the text saved on disk, printing is not necessary any longer, even for assignments to be handed in to a teacher. Both programs offer the teacher or editor the option of inserting comments into text files on disk. The comments can be used for guidance just as proof-reader marks are. Paper is still the preferred medium, however. In both word processors, the default printer settings will produce a single spaced 8.5" x 11" format.

TEXTRA

WordPerfect

Two methods of access to printing are available.

One access to printing is available via the [SHIFT-F7] command.

Printing from the main menu follows the default settings. Printing from the print menu allows for options in printer, text to be printed, etc.

Printing follows the default values, unless the options on the print menu are exercised.

Information on printing is explained in a separate chapter on printing.

Information on printing is gathered in a four-page Reference section entry on printing.

Discussion. It is in the printing functions that WordPerfect features can become most complex. It contains a larger variety of print options, type fonts, kerning, spacing options, and also can insert graphics into the text. To master all of this would take a considerable amount of time and practice.

The Wordperfect on-line Help for printing consists of the same list of options as seen if the print command [SHIFT-F7] were invoked. The Reference section of the manual provides a four-page description, with examples, of the options shown on the print menu. For information on each of the options, such as Printer Selection or View Document, the student must look up that option in the Reference section.

CONCLUSIONS

In contrasting the two manuals, we see that the TEXTRA manual is arranged in a holistic fashion, with word processing functions such as printing or editing described as topics, which include within the space of brief chapters all the commands and techniques related to the function. In the Wordperfect manual we find that it can't be approached as a single manual; the template, reference card, tutorial, manual and on-line Help have to be absorbed into the user's assumption before any level of proficiency can be achieved. That is, to gain mastery of the program, the package must be absorbed as a whole.

According to Brockman, the reference manual is the best approach to take with complex programs:

The more the software is open-ended, the more it can be customized by the user and used and viewed from different directions ... the more the material ought to be packaged in a reference manual (12).

This description fits WordPerfect, which makes a Reference manual and tutorial an apt strategy for documentation. As can be seen from the earlier descriptions, it would be difficult to learn to use such a program by relying on the Reference section of the manual. It is advisable to work through the tutorial lessons in the workbook or on-line tutorial. Brockman and Price agree that along with the reference manual must go a tutorial, either as a part of the manual or as part of the software, or both (Brockman 12; Price 99).

For the less complex TEXTRA, the strategy of an inferential manual with a varied approach to on-line Help also seems appropriate. Because of the reduced command repertoire the information in either form allows a compact presentation of the procedures for use.

In contrasting the two programs, we see that TEXTRA has a limited repertoire of commands but is sufficiently flexible to provide word processing functions of creating, editing, formatting, printing, and saving text as does the more complex WordPerfect. The appearance of the finished text can be varied sufficiently to give texture to the printed material, making it useful for more than just typing themes.

WordPerfect has a larger repertoire of commands, implying that the user has more options for manipulating and controlling the appearance of the output. For a student, this facility has limited value; to a writer involved in a book project, or a legal office producing documents to meet court standards, this degree of control is highly desirable, especially if it is repeatable. This is not to say that a novice user could not produce a theme paper using WordPerfect. But for the task of producing the theme paper, the student would use only a small fraction of the features in the program.

In selecting a word processor, a user should not be guided by just a long list of features and capabilities, but by style of the documentation, which determines the accessibility of the features, the ease of learning, and the time available for that learning. To a teacher making such a choice of word processor for students to use in a writing class, this is especially apt, since the more time the student spend on learning to use the word processor the less there is available for the task of constructing readable text.

Works Cited

- Adams, Kay A., Ida M. Halasz, and R. Jerry Adams. Handbook for Developing Computer User Manuals. Lexington: Lexington-D. C. Heath and Co., 1986
- Brockman, R. John. Writing Better Computer User Documentation: From Paper to Online. New York: Wiley Intersciences, 1986.
- Grice, H.P. "Logic and Conversation." Syntax and Semantics. Vol. 3. Ed. Peter Cole and Jerry L. Morgan. New York: Academic Press, 1975.
- Houghton-Alicio, Doann. Creating Computer Software User Guides: From Manuals to Menus. New York: McGraw-Hill, 1985.
- Norton Textra Writer 2.0. Computer Software. Ann Arbor Software, 1989.
- Price, Jonathan. How To Write a Computer Manual. Menlo Park: Benjamin/Cummings Publishing Company, 1984.
- Richard, I. A. 20th Century Literary Criticism: A Reader. Ed. David Lodge. London: Longman, 1972.
- Sides, Charles H. How to Write Papers and Reports about Computer Technology. Philadelphia: ISI Press, 1984.
- Sperber, Dan and Deirdre Wilson. "Precis of Relevance: Communication and Cognition." Behavioral and Brain Science 10 (1987): 697-754.
- W. W. Norton. Norton Textra Writer 2.0. New York: W. W. Norton, 1989.
- WordPerfect Corp. WordPerfect V5.0. Orem, UT: WordPerfect Corp. 1989.
- WordPerfect V5.0. Computer Software. WordPerfect Corp., 1989.

Wisnewski--Rhetoric of Documentation

Appendix A.
The WordPerfect Function Keys.

Key	Alone	[SHIFT]	[ALT]	[CTRL]
[F1]	Cancel	Setup	Thesaurus	Shell
[F2]	->Search	<-Search	Replace	Spell
[F3]	Help	Switch	Reveal Codes	Screen
[F4]	->Indent	->Indent<-	Block	Move
[F5]	List	Date/Outline	Mark Text	Text In/Out
[F6]	Bold	Center	Flush Right	Tab Align
[F7]	Exit	Print	Columns/Table	Footnote
[F8]	Underline	Format	Style	Font
[F9]	End Field	Merge Codes	Graphics	Merge/Sort
[F10]	Save	Retrieve	Macro	Macro Define

TEXTRA Function Keys

Key	EDIT (Alone)	[SHIFT]	[ALT]	[CTRL]
[F1]	more...	more...	more...	
[F2]	insert line	split screen	high ASCII	
[F3]	new format	search	edit formats	
[F4]	highlight	replace	works cited	
[F5]	handbook	copy block	DOS access	
[F6]	spell check	move block	customize	
[F7]	delete line	print	comments	
[F8]	delete block	page layout	statistics	
[F9]	delete word	merge/browse	endnotes	
[F10]	undelete	save options	options	

Appendix C WordPerfect Help Screen Example

Save

Saves the current document on disk. The DOS filename can have up to eight characters plus an optional period and three-letter extension. (_____.____). If a file with that name already exists on disk, WordPerfect will ask if you wish to replace it.

Files may also be saved using the Long Document Names feature. A long document name can contain up to 68 characters and is saved with your document along with the regular DOS filename (see Long Document Names on the Document Management/Summary screen under Setup: Environment).

Exit may also be used to save the document that you are working on. The only difference is that the Save key returns you to your document.

WordPerfect Fast Saves documents without formatting them first, cutting down on save time. You may change this option in the Setup: Environment menu.

Appendix D
Questionnaire.
W. W. Norton & Co.

1. I describe the tone of the TEXTRA manual as "narrative." Do you have a favored descriptor?

2. Was the initial manual written by technical personnel (programmers) or by a writer with English or Communications training? Has this changed since the initial version?

Manual is derived from commercial version by Ann Arbor Software written by technical person. Revised here at Norton by editor.

3. How much research went into selecting a manual design and integrating on-line with printed information?

Research — no. But a great deal of care in coordinating the two.

4. Did the simplicity or complexity of the program design influence the choice of manual design?

Absolutely. Norton Textra is easy to use, so we wanted ~~an~~ non-threatening manual.

5. Did the manual style of the IBM Corp. influence your manual design?

No. Macintosh manuals were more suggestive.

6. The first chapter emphasizes using the on-line help and "film strips" These "helps" seem to replicate the information in the book closely. Have you had complaints that dropping out of writing to read information on the screen is a disruption?

No. Information is repetitive because people learn in different ways. Best way with Norton Textra is to learn onscreen.

7. Professor Tuman, who is credited with developing the on-line handbook has also written a book on writing with TEXTRA. Was he also an author of the printed TEXTRA manual?

No

8. Feel free to add any other comments.

The program is easy enough to use that the manual is not really necessary, so we kept it brief and non-threatening but, we hope, always clear.

Appendix E

Questionnaire.
WordPerfect Corp.

1. Was the initial manual written by technical personnel (programmers) or by a writer with English or Communications training? Has this changed since the initial version?

Writer, English (communications) - Has not changed other than a team of writers now write our doc

2. WordPerfect comes with a reference manual, a tutorial (hard copy and on-line) and a Workbook. Was this design intended from the beginning, or were some of these added after version 1.0 to supplement the others? Design developed over time; and may change again in the future.

3. Was the on-line help material prepared by programmers or by communications personnel?

Programmers.

4. Did the simplicity or complexity of the program design influence the choice of manual design?

No

5. What economic considerations went into selecting manual design?

Cost of goods sold is a very important consideration. We want to low

6. Did the manual style of the IBM Corp. influence your manual design. Yes

7. WordPerfect seems to have spawned a secondary industry among writers and publishers who produce supplementary texts on how to use the program. To what extent was this a selected company strategy, or did they perceive a need? To what extent were they encouraged or aided by the company?

It was not a strategy. The authors perceived a need. They were not encouraged (for the most part) by the company - they were just acknowledged or tolerated.

8. Feel free to add any other comments.

We have no other comments.

Please send us a copy of your findings to:

Barney Oster
Editor, PC Products
WordPerfect Corporation
1111 17th Street, N.W.
Bellevue, WA 98005