To the Graduate Council:

I am submitting herewith a thesis written by Thomas J. Garritano entitled "Approaches to Assessing the Usability of Technical Documents." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Arts, with a major in English.

Michael L. Keene

Michael L. Keene, Major Professor

We have read this thesis
and recommend its acceptance:

Smith

Kathryn Riley

Accepted for the Council:

C W Minkel

Vice Provost
and Dean of the Graduate School

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Master's degree at The University of Tennessee, Knoxville, I agree that the Library shall make it available to borrowers under rules of the Library. Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of the source is made.

Permission for extensive quotation from or reproduction of this thesis may be granted by my major professor, or in his absence, by the Head of Interlibrary Services when, in the opinion of either, the proposed use of the material is for scholarly purposes. Any copying or use of the material in this thesis for financial gain shall not be allowed without my written permission.

Signature Thomas J. Garritano

Date July 1988

Approaches to Assessing the Usability of Technical Documents

A Thesis
Presented for the
Master of Arts Degree

The University of Tennessee, Knoxville

Thomas J. Garritano

August 1988

TABLE OF CONTENTS

Part I:
Literature Review 1

Overview of Usability 2

1. Testing Methodology/Technical Content 5

2. Style 18

Readability 18

Job Aids 21

Modularity 22

3. Graphic Design 24

Typestyles 24

Layout and Visuals 27

Summary of Part I 34

Part II:
Reader Response to
Intergraph Manuals DIXD6510,
DIXD6100, and DIXD5700 36

Background 36

Overview of Part II 37

1. Motivation 39

2. Task Orientation 42

3. Reference and Instruction 46

4. Style 50

Layout and Graphics 50

Readability 54

Conclusion 60

Bibliography 62

Vita 68

LIST OF FIGURES

Part I: Literature Review

FIGURE	PAGE
1. Weiss' Functions of User Documentation	4
2. The Levels of Edit, with Soderston's "Usability Edit"	8
3. Arnold's Two-Page Module	25
4. The Two Main Categories of Typefaces	26
5. Functions of Illustrations	29
6. Brockmann's Four Dimensions of Graphics	32
7. Example of Bleeding Tabs	33

Part II: Reader Response to Intergraph Manuals DIXD6510, DIXD6100, and DIXD5700

1. Mock-up of Two-Page Module for *WMS Section 6*	51
2. Introduction to *WMS Section 6.2*	52
3. *WMS* Page 1-12	55

Part I: Literature Review

This paper will argue that technical documents can best be usability tested by breaking them down into separate components that, once isolated, may be made to more directly reflect the needs of users. Various usability testing methodologies will be examined in the literature review, and their relative merits weighed. Also within the scope of the literature review are studies on such variables as readability, graphic design and layout, typestyles, and psychological cues -- all of which can drastically affect the usability of a document. In ***Part II: Reader Response to Intergraph Manuals DIXD6510, DIXD6100, and DIXD5700****, the usability variables of three selected documents will be examined in-depth, leading to suggestions for improving the documents and for avoiding problems in future documents.*

In the relatively brief period since its inception, usability testing has become fairly standardized; many of the basic techniques suggested by Candace Soderston of IBM in 1985 are still widely used and have been refined gradually by that corporation and others. Along with the laboratory-based method that currently predominates in the literature, the technique of field testing will also be discussed in **Part I** of this report and will play a significant role in **Part II**, which is itself the result of an actual usability assessment.

Most of the essays dealing with testing methodology are concerned primarily either with technical content (which varies from document to document) or with information-gathering techniques and, consequently, they fail to adequately describe the many complex variables that profoundly influence the usability of all documents. Although the methodology essays seldom mention them explicitly, the variables discussed in the Style and **Graphic Design** sections of this paper are at the root of every document's usability and simply must be considered during any usability test. One of the goals of this paper is to illuminate the specific variables that affect usability and to offer suggestions as to how they can be manipulated to create better documents.

Overview of Usability

The factors affecting usability can be divided into the following basic categories, as suggested by Mark Zirinsky in "Usability Testing of Documentation" (p. WE-12).

- Technical content
- Style (including organization and language)
- Graphic design

Whether a document succeeds or fails inevitably depends upon its performance in each of these categories, which can be further subdivided as the following review of usability-related literature will show. All three categories will be addressed in Part I, but Technical Content will be discussed mainly with regard to the role it plays in the testing methodologies; Part II will deal with Technical Content at greater length.

According to Edmond Weiss in *How to Write a Usable User Manual* (pp. 12-13), technical documentation has four functions: Reference, Tutorial, Demonstration, and Motivation (see Figure 1). Reference provides "key definitions, facts, and codes that [operators] could not be expected to memorize." Tutorials are instructions aimed at inexperienced users; Demonstrations "teach by showing" and are aimed at users with prior experience on the system at hand.

Perhaps most significant is Weiss' theory of Motivation, which is the psychological key to usability. Weiss asserts that users, in addition to needing to know *how* to perform a task, need to know *why* a task should be performed; without proper motivation, a user is liable to neglect even the most crucial tasks. However obvious a task's importance might appear to a systems expert, an inexperienced user cannot be expected to recognize that importance on his or her own. Weiss holds that documentation should be "designed to get people to do what they are reluctant to do . . . Put simply, many system problems can be blamed on reluctance, not ignorance" (p. 13).

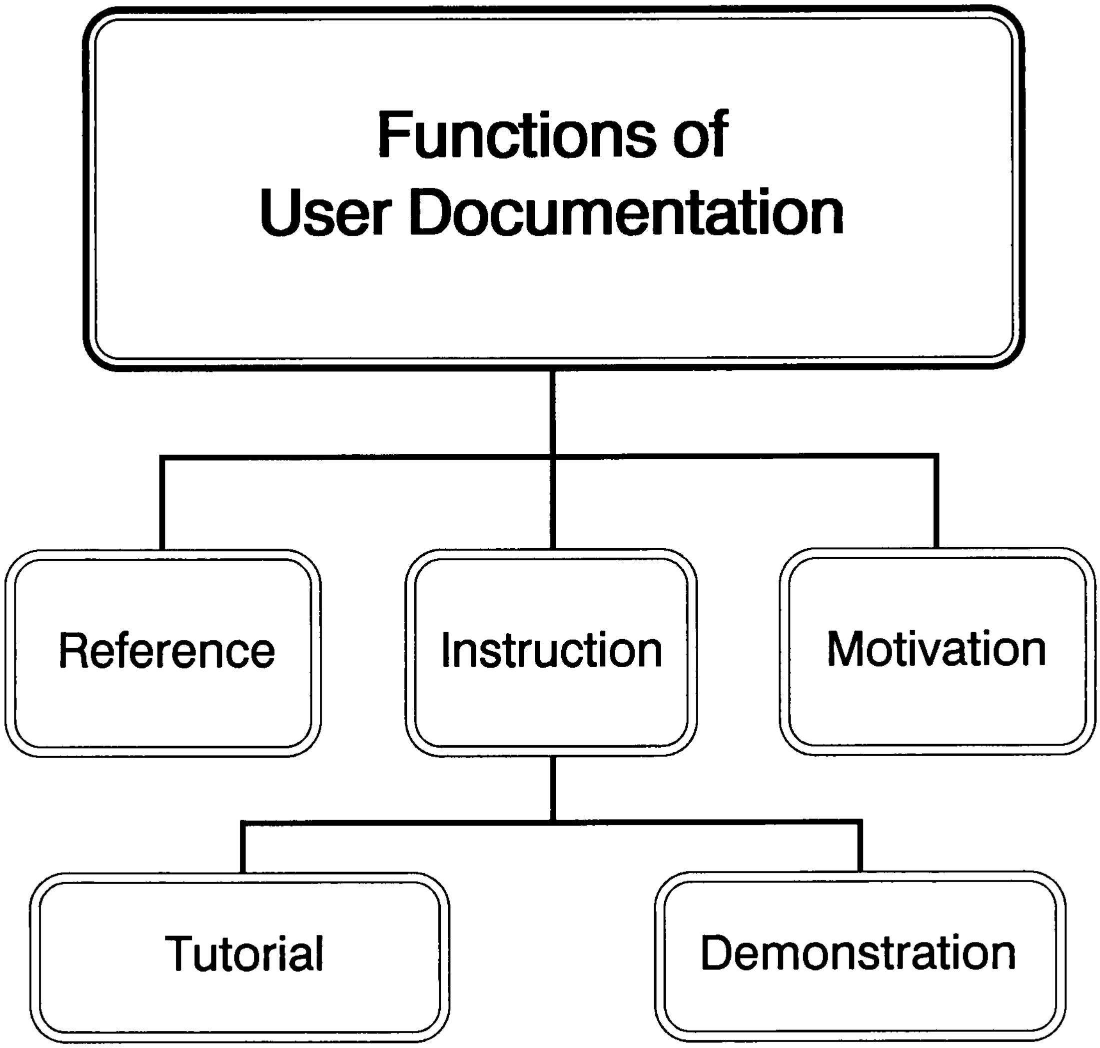


Figure 1. Functions of User Documentation.

1. Testing Methodology/Technical Content

Usability testing should evaluate the success of documentation in fulfilling its four functions. Both systems experts and writers/editors have blind spots that prevent them from noticing flaws in content that, paradoxically, might be readily apparent to less-experienced users. The goal of such assessments, therefore, is essentially to detect problems that would otherwise go undetected during the ordinary document preparation process. Most of the writings discussed in this section of the paper center on matters of content, to the exclusion of other important variables affecting usability that will be addressed in the ***Style*** and ***Graphic Design*** sections of this literature review. These variables can be equally important, however, and they represent areas of study that should play much more significant roles in future essays on usability testing methodology.

The usability concept dates to M. A. Atlas' article, "The User Edit: Making Manuals Easier to Use," in the March 1981 issue of *IEEE Transactions on Professional Communication*. Atlas suggested the technique of observing test subjects in the process of using products and documents as a means of measuring and improving usability. Candace Soderston coined the term "usability" in her January 1985 article in *Technical Communication*, "The Usability Edit," which described IBM's implementation of the technique. This new approach essentially took the old saw about "audience awareness" and pushed it out of the realm of theory and into direct practice. Usability testing has since become a prime issue in the field of technical communication, with many refinements on Atlas' and Soderston's initial formulas being offered. Among such articles are Esther Kando Odescalchi's "Productivity Gain Attained by Task Oriented Information," Judy Iler's "The CAMS Usability Laboratory," Moore, Eyre, and Rideout's "Testing Software Installation Procedures: An Integrated Approach," and Stephen Doheny-Farina's "Usability Testing in the Field." Each of these methodologies uses, to varying degrees, the technique known as *protocol analysis* (Mulcahy; Witman), in which the test subject describes aloud each action he or she takes while working with a product or document. By gleaning the most important information from among these essays, the first step will be taken toward constructing a more complete philosophy of usability assessment.

Soderston's seminal work has since been expanded upon by others, but it continues to be the blueprint for usability testing. In it, she stresses the shortcomings of the outmoded method of having only "subject matter experts" review documentation before publication; however well-intended, she says, such reviews are inadequate because "technology does not exist in a vacuum or rest in an ivory tower" (p. 16). In other words, documentation that is system-oriented rather than user-oriented presents an often-impregnable barrier between the user and his or her needs and expectations.

In their highly influential book, *The Levels of Edit*, Robert Van Buren and Mary Fran Buehler established a systematic approach aimed at helping editors to prioritize their various assignments; Van Buren and Buehler recognized that, while each document must at least be made consistent and correct throughout, some documents simply require more thorough editing than others. They discerned nine types of editing procedures, which they ranked according to thoroughness and frequency of use. To augment Van Buren and Buehler's edit levels, Soderston proposes that a "usability edit" be applied to the most crucial documents (see Figure 2).

Soderston draws an important distinction between usability edits and usability tests. While the word *testing* implies an action that is taken when a document is complete or nearly so, she says, *editing* represents an ongoing process that begins during the earliest stages of document preparation and doesn't end (if it ends at all) until the document reaches print. The term usability edit also strongly reinforces the notion that those involved in preparing documentation must constantly focus on creating user- and not system-oriented manuals; the problems of system-oriented documentation will prove too deeply rooted to be eradicated by even the most elaborate testing if that testing occurs only in the latter stages of production.

The assessment procedures that Soderston and others at IBM's Human Factors Laboratory apparently began implementing in the mid-1980's entailed the use of test subjects who were observed while attempting to use various products and documents in a controlled laboratory environment. Many of the techniques she describes are now in wide use across the U.S., such as

Type of Edit	*Level of Edit*					
	0	1	2	3	4	5
Coordination	X	X	X	X	X	X
Policy	X	X	X	X	X	X
Integrity	X	X	X	X	X	
Screening	X	X	X	X	X	
Copy Clarification	X	X	X	X		
Format	X	X	X	X		
Mechanical Style	X	X	X			
Language	X	X	X			
Substantive	X	X				
Usability	X					

Figure 2. The Levels of Edit, with Soderston's "Usability Edit" (Soderston, p. 17).

using one-way glass partitions, recording the sessions with video equipment, and placing a "help phone" near the subject to relieve anxiety. In a statement that is echoed in several later articles by other authors, Soderston emphasizes the importance of convincing the test subject that the documentation and the system are being evaluated, not his or her aptitude and performance. Particularly in cases involving protocol analysis, the subjects may be severely inhibited and, therefore, the accuracy of the results obtained might be compromised.

An article written subsequent to Soderston's by another member of IBM's Human Factors Laboratory, Odescalchi's "Productivity Gain Attained by Task Oriented Documentation," offers statistical proof that user-oriented (synonymous with task-oriented) documentation raises user productivity levels far above those associated with system-oriented documents. Odescalchi defines user-oriented documentation as "step-by-step instructions on how to perform tasks," with a personal tone and written in the active voice. System-oriented documentation "describes how the product works," has an impersonal tone, and is written in the passive voice, she says. User-oriented manuals must identify:

- the action of the user,
- the conditions (i.e. tools and supplies needed),
- the environment (locale), and
- the criteria (e.g., time needed to perform tasks).

IBM tested 26 volunteers possessing varying degrees of mechanical aptitude, who were given instructions on how to perform three separate tasks. Task 1 was a "familiar" project: to assemble a flashlight. Task 2 was "unfamiliar" for most of the test group: to wire an electrical circuit. Task 3 was also assumed to be unfamiliar: to assemble a lock block rescue truck. Each volunteer received a mixture of user-oriented and system-oriented documentation.

Odescalchi concluded, not surprisingly, that failure rates of the test subjects were dependent on their familiarity with the tasks. She states unequivocally that user-oriented documents are "essential" to inexperienced users and that system-oriented documents are useful only to experienced users. User orientation improved productivity by 41 percent over system orientation, and 79 percent of all test subjects preferred the user-oriented manuals. On the negative side, writers were found to require 42 percent more time to generate a user-oriented document, which on the average would have 33 percent more words and use 87 percent more paper than a system-oriented document (pp. 360-62).

Judy Iler's article explains the usability testing procedures used in General Motors' CAMS laboratory in Troy, Michigan. An interactive computer system designed to diagnose service and repair problems in cars and trucks, CAMS is used by repairmen at GM dealerships across the country. These usability tests were intended to determine the degree of difficulty associated with installing the system, and to determine the effectiveness of the CAMS user manuals and software. Tests were conducted in a controlled environment similar to that employed at IBM's Human Factors Laboratory, but the equipment for monitoring subjects was even more elaborate: multichannel audio to narrate the sessions, a complex video editing system, remote-control cameras, and even a special effects generator (the exact purpose of which Iler never divulges). A staff of ten employees administered the tests, which were divided into three stages.

1.) Unpacking and installation
2.) On-line tutorial and introduction
3.) Diagnostic program

The first stage raised the issue of whether the users should be required to install the CAMS system. Although Iler never addresses this explicitly, one might suspect that the users' problems with the instructions stem from system-oriented documentation. Because automobile repairmen obviously must have great mechanical aptitude, a system-oriented bias would seem the likely cause of the problems; with a well-focused manual they would in all likelihood have little difficulty mastering even complex tasks. Iler mentions that the second test stage led to "enhanced logic flow of software" (p. VC-83), which may be another means of saying that the user's needs were belatedly considered by those in charge of usability.

Hewlett-Packard, like IBM and GM, has a "Human Factors Department" that assesses usability under laboratory conditions. As described in Moore, Eyre, and Rideout's excellent "Testing Software Installation Procedures: An

Integrated Approach," the Hewlett-Packard (HP) testing system is a highly refined operation that features an accredited psychologist who observes and comments on the test procedures as they unfold. The HP lab is equipped with such standard equipment as video cameras and recorders, microphones, a help phone, and a partition to separate subjects from observers, along with a personal computer into which the psychologist types commentary at specific junctures in the process. The article revolves around the user- vs. system-oriented dichotomy; the authors stress that usability testing is necessary to

- Identify problems overlooked by employees who are overly familiar with the product.
- Solve problems *before* the product reaches customers and causes dissatisfaction.
- Improve the public's perception of the company and its products.

Moore, Eyre, and Rideout describe the HP tests in greater detail than do the authors of the other articles discussed here. Because both the motivation and the methods behind HP's testing are made extremely clear to the reader and because *specific* suggestions for improving usability are offered, the article is an indispensable portrait of the state of the art in usability assessment as of 1986.

HP EGS is a computer-assisted design (CAD) system for engineering applications. During its first years of release, HP Systems Engineers had gone on-site to install EGS, but increased sales eventually made this economically unfeasible. It then became apparent that the system-oriented documentation that had sufficed earlier would have to be re-oriented to enable inexperienced users to install the entire system themselves, without the assistance of HP personnel. The tests described by Moore, Eyre, and Rideout were aimed at determining the time required by users to complete various tasks, and at determining the types, severity, and frequency of problems that arose during installation. After gathering this data, the testers ranked the severity of the problems, proposed solutions, and helped implement those solutions.

Unlike, for example, the laboratory tests detailed in Iler's essay, the HP testers focused on the practical rather than the statistical significance of the test results. The most crucial practical concern was the amount of outside help needed to perform the tasks, because HP's primary goal was to reduce its client's costs by eliminating the considerable expense of having a Systems Engineer install EGS. Related goals were to decrease the calls to HP for technical support and to decrease overall startup time.

Three separate versions of EGS were tested over a six-month period. The earliest version was the most system-oriented of the three, and it was rated "poor" by the test subjects. With the third and final version (which was ultimately made available to the public), HP had achieved a greater degree of user orientation, and the subjects rated that version "good." Like Soderston, the HP authors discovered that "user interface should be an iterative process -- it should not be defined and frozen too early" (p. 379).

In addition to improving the usability of the EGS documentation, they were able to narrow the field of available system solutions to the following three choices that were ultimately presented to HP customers.

1.) Purchase EGS as a stand-alone, configured hard disk.

2.) Purchase EGS software on floppy diskettes and self-install it.

3.) Pay an HP Systems Engineer to install EGS.

Had usability not been improved via laboratory testing, only the third option would be available, and public perception of the EGS would be dramatically decreased.

In a manner of speaking, each of the articles discussed above (apart from Moore, Eyre, and Rideout's) is guilty of its own brand of "system bias" -- the system, in each case, being the particular brand of usability testing methodology discussed in the article. The usability testers, like the computer system experts, fall prey to the forest-for-the-trees syndrome; they frequently focus on general methods of assessment without detailing the specific areas in which usability can be improved. The "users" of essays on usability testing are thus denied access to practical information and may be overburdened with excessive theoretical information.

One of the most valuable aspects of the HP article, therefore, is its list of precise suggestions for improving the usability of documentation. Below is an abbreviated version of that list (p. 380).

- Include only the steps necessary to complete a given task.
- Tell users what to expect at the very outset of the document.
- Indicate the order in which manuals (or sections thereof) should be read.
- Include a troubleshooting section after each series of steps to help users recover from errors.
- Point users to subsequent steps with a "Where to Go Now" section after each significant task.
- Consider the relative location of different bits of information.
- Give users a way to verify the successful completion of tasks.
- Don't forget to provide users with (seemingly) obvious information.

Doheny-Farina also addresses practical concerns in attempting to establish detailed guidelines for usability testing in the field -- that is, testing not conducted in the controlled laboratory environment discussed in Soderston's, Iler's, Odescalchi's, and Moore et. al's articles. This distinction is important for several reasons: field testing involves actual customers (and thus requires great tact), it generally evaluates products that are already commercially available, and it tends to yield qualitative results as opposed to the quantitative data generated by laboratory testing. Doheny-Farina begins by describing the various types of information that can be collected in the field, which he divides into three categories.

1.) Descriptions of physical and social environments

2.) Copies of the users' re-worked or annotated documentation

3.) Accounts of actual use

The first category includes such factors as whether the documentation is stored in an accessible area, whether the users employ self-written "how-to" posters for reference guides, and whether the users' social status (relationships with colleagues and supervisors) affects their needs with regard to the documentation. The second category helps detect inaccuracies or omissions in the documents. For the third category, Doheny-Farina suggests the following four methods of data collection.

1.) Field notes (three kinds: Observational [objective], Theoretical [subjective], and Methodological [speculative])

2.) Discourse-based interviews (based on re-worked documentation)

3.) Post-hoc interviews (based on observations, but conducted *after* testing to avoid interrupting the test subject)

4.) Read-aloud protocols

Among the candidates for inclusion in a field usability report are the users' actual re-worked or annotated documentation; evidence of user problems (including documentation and product design); evidence of successful strategies or features; details about the work environment (including maps of the work area); details about the physical use of documents (where they're stored, whether they're read on desktops or in users' laps); user profiles; suggestions from users; analyses of interviews and protocols; and speculation on areas requiring further research.

The Intergraph manuals discussed in detail in Part II of this report were subjected to field usability tests by participants in English 5280, a UT graduate course titled *Reader Response to User Documentation*. Conducted at the Geographic Information Services office of the Tennessee Valley Authority (TVA) in Norris, Tennessee, these tests were subject to many of the constraints described in Doheny-Farina's article. Because the TVA employees were interviewed on company time, the sessions were necessarily brief; two users and one supervisor were made available for a joint session that lasted less than three hours. The documents discussed were not the prototypes tested in sterile usability labs, but manuals that had been in use at TVA for a year or more. The interviewees' focus, therefore, was on proposing solutions to longstanding problems arising in daily use rather than on simply acquainting themselves with a new product, as is the case in lab testing. A crucial difference between the Intergraph/TVA test and the field test described by Doheny-Farina, however, was that both the student interviewers and the interviewees could be more candid than if the test had reflected a true company/client relationship. This "third-party method" of usability testing has been overlooked in literature on the subject to date, but its beneficial impact is certainly worth consideration. Nuclear and chemical plant operators, for instance, have long recognized the desirability -- even the necessity -- of commissioning objective third parties to assess safety and functionality.

2. Style

Readability

The readability formulas that have long been used for rating the difficulty of textbooks may have partially inspired the development of usability testing. Though obviously not a panacea, readability testing can play a useful subordinate role in assessing overall usability, as will be demonstrated in Part II of this report. In use for more than 35 years, the Gunning Fog Index continues to be one of the most popular readability formulas. Robert Gunning designed the Fog Index for nearly universal use, and its focus on sentence length and word difficulty is certainly germane to the field of technical communication. The three-step formula is simple, yet effective: 1.) Count the words in successive sentences (e.g., several 100-word samples); 2.) Count the number of words of three syllables or more per 100 words; and 3.) Add the two totals and multiply by .4. The result will be a numerical Fog Index equivalent to an educational reading level. A score of 12, for example, indicates the reading level of a senior in high school (pp. 34-39).

One of the more interesting recent studies appears in "Authors' Plans Meet Readers' Plans" by Jan Spyridakis, which surveys various studies of the author/reader dynamic. Whereas the Gunning Fog Index measures readability via sentence length and complexity, Spyridakis summarizes the results of a number of studies that assessed the effects of what she calls "signals" on reader comprehension. The following are among the document features that fall under the heading of signals.

- Previews
- Headings
- Logical connectives
- Pointer words
- Summary statements
- Conjunctions

Because she neglects to adequately define these terms, her article is not as useful as it might have been. Still, Spyridakis provides ample evidence that the so-called signals -- many of which are similar to the devices described below under the heading of ***Job Aids*** -- greatly facilitate comprehension, and that the importance of word and sentence length must be balanced against other variables that often go undetected by readability formulas.

Philip Lillies stretches the boundaries of readability testing to include analysis of what he calls the four levels of a document:

1.) Overall organization;
2.) Page layout/format;
3.) Paragraph organization; and
4.) Sentence construction.

He defines readability as "a measure of the efficiency with which a text can be comprehended by an intended reader" (p. WE-13). Rather than proposing yet another formula aimed at quantifying readability, however, Lillies seeks to improve the quality of documentation at each of the four levels; his conclusions, although not strongly reliant upon statistical evidence, are based on expertise that he has accumulated during his editorial career. Lillies' article includes a nearly comprehensive "shopping list" of the components that must be present in a document to ensure readability, and hence usability.

The first level, Overall Organization, should include easily accessible introductory matter, indexes, a table of contents, separate sections (or even separate manuals) for reference and tutorial functions, and job aids such as tabbed chapter dividers marked with functional titles as opposed to merely numbered (see ***Job Aids*** also). At the page layout level, Lillies says, there should be visually prominent, truly descriptive titles. A hierarchical style based on tasks should be employed, rather than a narrative, system-based style. He claims that serif fonts are easier to read than sans-serif fonts, and that point sizes for blocks of text should range only between 9- and 12-point type (typefaces are discussed in greater detail in the Graphic **Design** section of this literature review). "Rag right" text is also more readable than justified text, and lower-case letters are easier to read than all upper-case text (which should be avoided whenever possible).

As for paragraph organization, Level 3, he advocates adherence to H. H. Clark and S. E. Haviland's "Given-New Contract Theory," which holds that a user assimilates new information by searching his or her memory for similar knowledge (called the "Antecedent") that can be used as a guide to understanding. The primary goal of technical documentation, Lillies says, is to minimize the number of false inferences that are drawn by the user. This must be achieved through two means: semantic coherence (i.e., clarity of meaning) and lexical coherence (i.e., clarity of vocabulary). Finally, readability at the sentence construction level relies on conciseness and repetition. He cites three studies that show how shorter sentences aid comprehension and recommends that this be achieved by converting nominalizations and gerunds to verb forms and by using the active voice when possible. In keeping with the Given-New Contract Theory, repetition of nouns is preferable to the use of pronouns and synonyms (pp. WE-13-16).

Job Aids

This is a relatively new term for some fairly old means of organizing information in ways that are more comprehensible than long, uninterrupted passages of text. As Lisa Griffith writes in "Job Aids: When a Manual is Too Much or Not Enough," these means include step-by-step instructions, decision tables, flow charts, and diagrams. She recommends listing all pieces of information that are candidates for inclusion in a document, then ranking them in three categories according to importance: 1. Essential; 2. Desirable; 3. Helpful. For instance, warnings and cautions are *always* essential. To determine which job aids are appropriate, Griffith suggests that the writer should always ask two questions: "What *must* the user know?" and "What does the user *already* know?"

Patricia Lawson's "Job Aids: Give the Readers What They Want" also advocates the use of job aids for improving usability. Job aids use only "the barest number of words to convey essential information," relying strongly on visual cues that are easily perceived and understood (p. 339). Lawson asserts that most people can remember five to seven steps, and that any task requiring more than seven should include a job aid. They are not so much instructional as they are procedural, she says, in that they are usually intended to be used repetitively throughout the lifetime of the document; they tend to fulfill both tutorial and reference functions. The user-written reference posters described by Doheny-Farina are simply homemade job aids that, like those Lawson mentions, are more portable than a hefty manual. Job aids are most often designed to "become part of the worker's daily environment," although they are also useful when used as reminders for infrequently performed tasks.

Modularity

This concept emphasizes the importance of organizing information into two-page "modules" that are both easily comprehended by the user and easily revised by the technical editor or writer. Weiss' *How to Write a Usable User Manual* stresses modularity by example -- the book itself conforms to the two-page standard. In terms of usability, readers tend to absorb information more quickly when it is presented on two facing pages; this format is eye-pleasing and, if executed correctly, it avoids the pitfall of placing related bits of information on inconveniently located pages that must be flipped repetitively. Moreover, because each two-page module is a separate entity, they can be edited and even rearranged without necessitating a major revamping of the document. Weiss traces the genesis of the modular format to Hughes Aircraft Company in 1965, and he neatly summarizes their importance as follows.

> *A modular publication is a series of small, cohesive chunks of technical communication of predictable size, content, and appearance. Once the design -- the exact sequence -- of the modules is frozen, it becomes possible to treat the one, large, complicated manual as a set of many small, nearly independent manuals. Each takes only an hour or two of effort to write; each can be developed independent of the others, in any sequence* (p. 82).

Weiss also notes that the two-page modules can help both to gauge and to limit the expense of producing a document. William Arnold, in "Learning Modules in Minimalist Documents," claims that using concise modules causes the size of an average document to decrease by 30 to 50 percent, further lowering production costs.

Arnold also makes a persuasive case for modularity as an aid to comprehension. "Everything a person needs to know should appear within a module," Arnold writes. "To document a major task [i.e., one absolutely requiring more than two pages], a writer breaks the task down into a series of subtasks until the task in point can be documented within the one or two page ecology" (p. WE-17). By creating such a "minimalist" document, according to Arnold, the writer can use brief, task-oriented sentences and job aids to guide users through even the most difficult operations. Figure 3 displays his suggested outline for a two-page module.

Task-oriented headings are also vital elements of usable modules, Arnold believes. Each module should have its own title, such as "Initializing a Disc." His sample headings are active rather than passive -- *Recovering from Errors*, not *Error Recovery*. The emphasis must remain on teaching the reader how to use a product, rather than on explaining how a product works (again, user vs. system orientation).

3. Graphic Design

Typestyles

The many typestyles made available by new technology have increased the importance of this already-vital component of technical documents. When used carefully, variety in type can greatly enhance usability; if overused, it can become an impediment. The two main categories of typefaces are *serif* and *sans-serif* (see Figure 4). Serif type, such as the Times Roman type in Figure 4, has cross-lines (also called "feet") at the ends of letter strokes to facilitate eye movement. As a result, serif type may be read easily even for long periods of time and is best for extended blocks of text (Brockmann, p. 73). Sans-serif type, such as the Helvetica type in Figure 4, is simpler and has cleaner lines, making it appropriate for display headings and captions. In addition, bold type can be used to lend

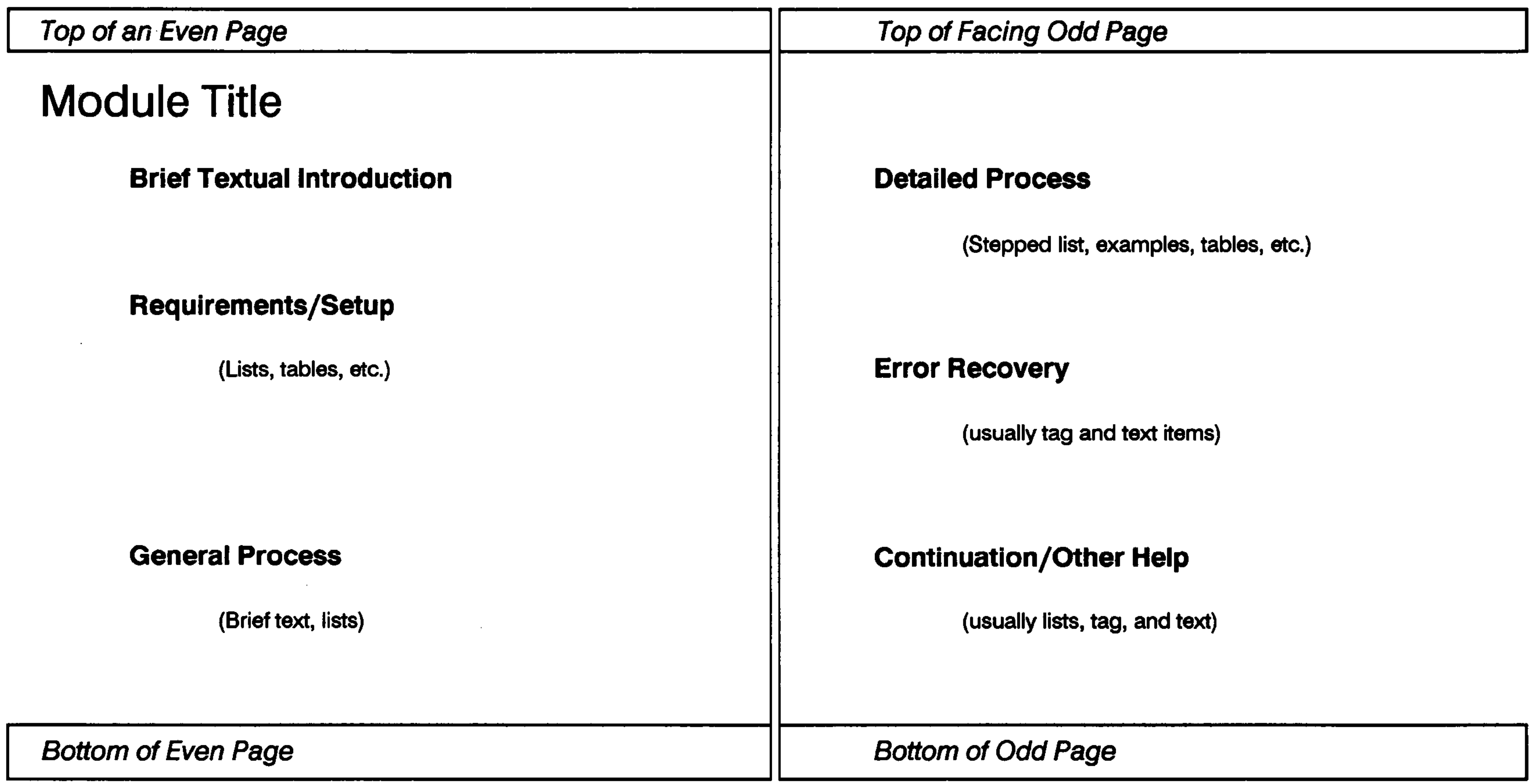


Figure 3. Arnold's Two-Page Module.

Figure 4. The Two Main Categories of Typefaces.

authority, italics can emphasize text somewhat gently, and the two can be combined to convey motion. Type size can also be varied to shift emphasis (Parker, pp. 25-26, 41-42).

Gayle Wyman and Lynn Rothberg have written two valuable articles on the subject, titled "Why Typestyles Convey Mood and Image" and "Selecting Type for Mood and Image." The former provides a brief history of typefaces, which serves as a good introduction to the various types and the responses they evoke. More to the point is the latter article, which draws on examples from newspapers and advertising to illustrate the range of messages that can be conveyed with typefaces. A serif type such as Times Roman, for example, can be used to convey credibility; sans-serifs like Helvetica can seem more modern and dynamic. In a technical manual, care should be taken to avoid intimidating the reader with an overly aggressive typeface. Rothberg and Wyman note that new technologies are constantly expanding the variety and availability of typefaces, so at least a rudimentary knowledge of the subject will be increasingly important to technical communicators.

Joyce Hutton's "The Character and Its Role in Message Conveyance" offers some additional insight into the precise effects of particular typestyles, along with some do's and don'ts. She echoes Lillies, who said that all-capital words ar much more difficult to read than the standard mixture of capitals and lower-case. As a rule, she says, no more than two typefaces should be used on any page; to use more can lead to confusion and eye fatigue on the part of the reader. Sans-serif types go well with almost any other, but different serif types should not be mixed. Hutton recommends that long blocks of text be broken up with subheadings, which should be a larger, bold version of the text's typeface (pp. VC-41-43).

Layout and Visuals

This very broad category encompasses such issues as accessibility of page design, the effective integration of art and text, and the importance of white space (page space left totally devoid of text or graphics) as an aid to comprehension. These and other visual elements are increasingly crucial to editors, who are often asked to fulfill graphic art functions via desktop publishing programs and equipment. Willoughby assesses the overall usability of many types of visual "access aids" in his article, "Information Design: Determining Access-Aid Tradeoffs." Retrievability aids such as tabs and dictionary headings also fall into this category (Ray et al.).

Cohen's "Graphic Design Do's and Don'ts" is a brief primer aimed at the many writers and editors whose duties have grown to include control of layout and visuals. Among the "Do's" are suggestions to use visual rhythm and repetition, to strive for visual balance, to use ruled lines for emphasis and variety, and to leave sufficient white space. The "Don't" list includes warnings against using all capital letters, using too many typefaces, indenting excessively, and cluttering pages.

Watzman emphasizes that few concrete design rules exist, though there are some important guidelines that should be followed. Simplicity and accessibility are paramount virtues, the latter of which is predicated on how well the document fits the environment in which it is to be used; large manuals, for example, are unsuitable for daily, laptop use. Horizontal and vertical rules should be used along with other graphic devices to create an instantly recognizable visual hierarchy, one in which information is divided into easily digested "chunks." Visual cues such as icons should be used to indicate warnings, procedures, and cross references (see Redish, below). She warns, however, that "less is more -- too many graphics, illustrations, lists, or tables are as bad as (or worse than) too few" (p. ATA-49).

In "Integrating Art and Text," Janice Redish offers persuasive evidence of the ability of visual communication to enhance usability (Figure 5). Graphics, she says, can act as:

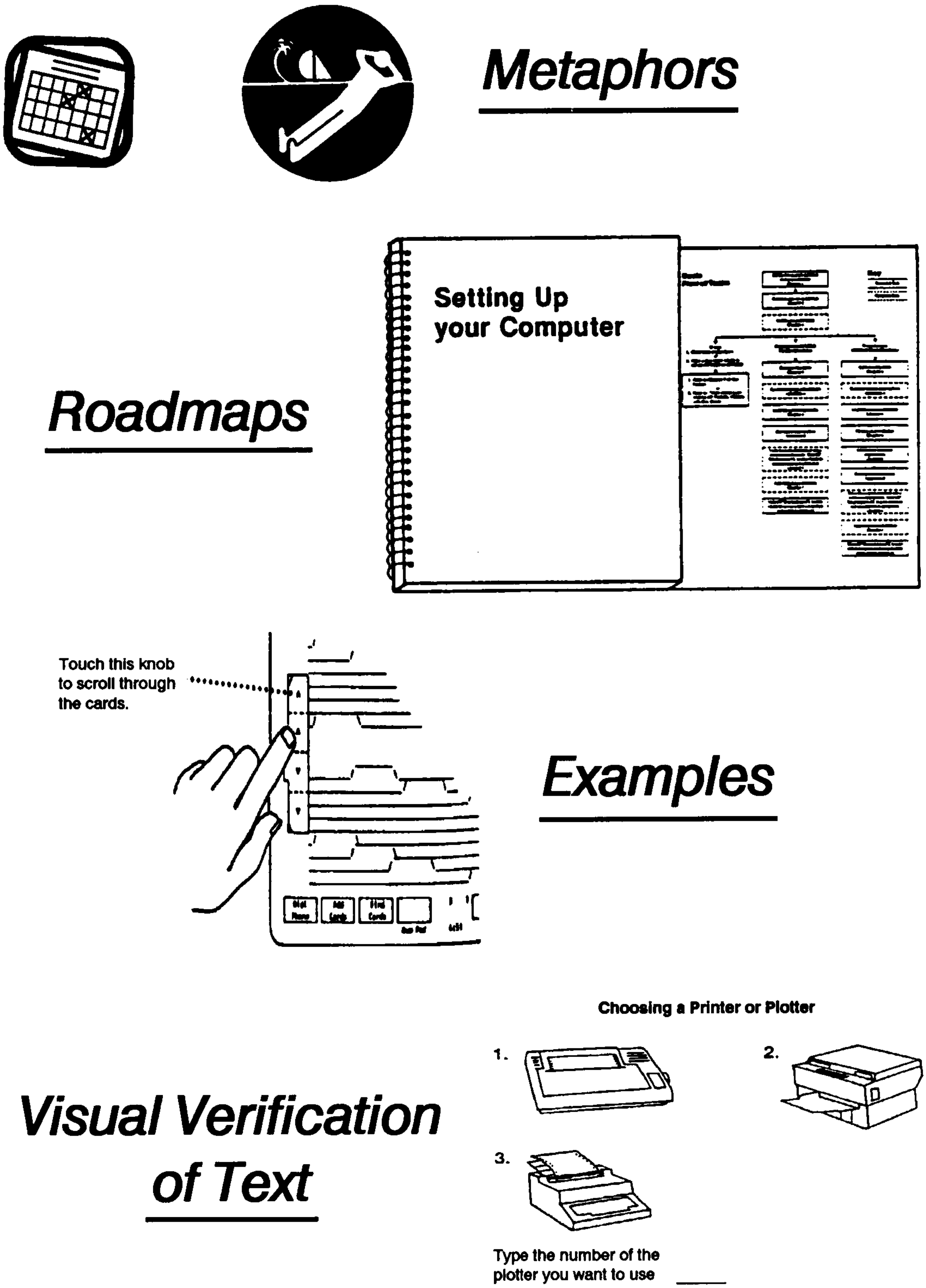


Figure 5. Functions of Illustrations (Redish, pp. VC-5-6)

- metaphors
- roadmaps
- examples
- visual verification of text

Metaphoric illustrations such as icons can quickly familiarize users with new terms and concepts by vividly connecting them with already-familiar information. Visual "roadmaps" such as the flow chart pictured in Figure 5 are useful in establishing the relationships between various elements of a system or its documentation. They can serve as introductory material, as quick reference, as checklists, or as trouble-shooting aids. The printed representation of a hand pointing to a display screen is just one instance of how visual matter can serve as an example assisting comprehension. And finally, illustrations such as the printer plotting diagram can reinforce textual information that otherwise might take much longer to absorb.

Brockmann offers some further principles for the effective use of graphics (pp. 184-88). An illustration should be balanced, he says, so that the reader is able to quickly recognize its visual center. Graphics should be separated from the surrounding text by contrasting lines, color, or shades. Devices such as arrows should be employed to guide the reader's eye in the desired direction. Finally, Brockmann cites William Horton's fine article on the use of graphics, "Towards the Four-Dimensional Page," to emphasize that "the more dimensions a graphic has, the more memorable it is" (p. 185). To illustrate, he presents the four self-explanatory examples shown in Figure 6.

Willoughby defines an "access aid" as "a graphical or typographical device which aids a reader to access desired information. Access aids provide location reference, guide the reader's eye, or classify information" (p. VC-89). As they are distinct from job aids, a narrower category whose variables focus more on content and theory, access aids can be divided into the following four major types.

1.) External (e.g., cover, tabs, index)
2.) Internal (e.g., table of contents, running headers)
3.) Paginal (e.g., headings, bullets, step numbers)
4.) Global (e.g., white space, color, reverse type)

To provide a guide with which editors may make informed choices, Willoughby uses three separate tables to rank various access aids by usefulness, cost efficiency, and relative advantages. Although dictionary-style diecut tabs, for instance, are highly useful in several ways, their cost efficiency is low. Artwork, on the other hand, is both effective and inexpensive. Among the most desirable aids overall are monochromatic divider tabs (external), margin icon tabs (internal), artwork (paginal), and reverse fonts (global).

A *one-dimensional* graphic is the least memorable. An example of this would be

"Press the RETURN key."

A simple *two-dimensional* graphic would be more memorable and would revise the example above:

"Press  **."**

A *three-dimensional* graphic would revise the example to:

"Press  **."**

A *fourth dimension* can be simulated by indicating action:

"Press  **."**

Figure 6. Brockmann's Four Dimensions of Graphics.

"Effectiveness of Tabs and Dictionary Heads as Retrievability Aids," by Ruth Ray, Anne Myers, and Kathleen Gomoll, reports the results of an IBM study of the usability of various heading and tab types. Six versions of a manual were created, each using a different combination of headings and tabs, and test subjects were asked to locate 23 messages in a random order. The amount of time required by the subjects to locate the messages was recorded, and comments were solicited. The IBM researchers concluded on the basis of the data collected that protruding tabs are preferable to bleeding tabs (in which ink is "bled" flush to the right margin; see Figure 7) and that dictionary-style headings have little effect on retrievability. The bleeding tabs, in fact, may lead to user discontentment because they are so difficult to use. A desirable ratio of tabs to pages is 1:34; more tabs than this can be counterproductive, leading to an annoying "porcupine effect" (p. RET-47).

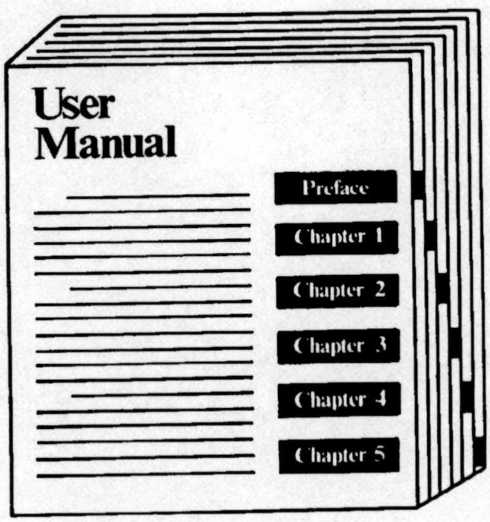


Figure 7. Example of Bleeding Tabs (Brockmann, p. 199).

The importance of white space in aiding comprehension is mentioned frequently in the literature. Brockmann and Reitman recommend that, on the average, pages should contain 40 percent text and 60 percent white space; to emphasize a visual hierarchy, the white space should be *active* (separating units of information) rather than *passive* (i.e., at the edges of the page) (Brockmann, p. 75; Reitman p. WE-94). In "A New Look at White Space," Henrietta Shirk calls its use "one of the undisputed commandments of page design" (p. WE-130). She agrees with the 40:60 ratio, but warns that the proper use of white space is an "aesthetic" exercise requiring great flexibility on the part of a technical communicator. Shirk also stresses that the passive white space on a page can be just as important as the active, because one seldom succeeds without the other.

Summary of Part I

As demonstrated in the preceding literature review, usability testing methodology has become somewhat standardized, particularly with regard to laboratory tests. The specific issues dealt with in usability tests, however, are seldom addressed directly in the literature, despite the fact that many issues are common to all technical documents. The variables discussed above in the **Style** and **Graphic Design** sections can, as indicated, be manipulated to enhance usability. Those same variables are also among the issues that should be scrutinized during thorough usability testing.

Part II of this report, **Reader Response to Intergraph Manuals DIXD6510, DIX6100, AND DIXD5700**, assesses the usability of three computer user manuals that were subjected to a field usability test. Many of the issues raised in Part I are also pertinent to Part II, and the suggestions offered for improving the Intergraph manuals could be applied with equal success to many other technical documents. In terms of technical content, the importance of task orientation and motivation -- mentioned prominently in much of the literature surveyed above -- remains a dominant theme in Part II. As for layout and graphic design, the value of two-page modules in enhancing usability (as advocated by Weiss and Arnold) is also demonstrated.

Part II:
Reader Response to Intergraph Manuals DIXD6510, DIXD6100, and DIXD5700

Background

Part II assesses the usability of three computer manuals that were selected as research models for the class titled *Reader Response to User Documentation*, English 5280, taught by Dr. Michael Keene at the University of Tennessee, Knoxville. Dr. Keene's course was aimed at establishing a method for the systematic study of the relationship between computer manuals and their various audiences, focusing on the tasks of identifying impediments and devising enhancements to readability and usability. Students were first exposed to a theoretical approach to the study of such documents, after which they were asked to apply that theory to manuals donated by Intergraph Corporation, a leader in the field of computer-assisted graphics and design.

Manual DIXD6510 supports Intergraph's architectural design system, DIXD6100 supports the World Mapping System (WMS), and DIXD5700 supports the Graphic Polygon Processing Utilities (GPPU). Both WMS and GPPU are used by the Geographic Information Services office of the Tennessee Valley Authority (TVA) at Norris, which cooperated in the effort to assess the manuals by allowing Dr. Keene and his students to interview several of its employees who had considerable experience using the Intergraph systems.

The task of evaluating the manuals was made immeasurably easier by the fact that the class participants agreed on virtually all of the significant areas in need of improvement: Indexing, Graphics, Layout, Prose, and overall Structure (especially with regard to Motivation and Task Orientation). Moreover, the early impressions of the classmembers were largely reinforced by interviews conducted with TVA employees Bob Bryant (Senior Systems Analyst), David Hankins (Operator), and Ebeth McMullen (Manager). Though each of these three individuals had his or her own specific perspective and level of expertise with regard to the manuals, they nonetheless confirmed many tentative conclusions reached previously during extensive class discussions of the three manuals, in addition to shedding new light on several aspects of usability that had not been considered to that point.

Overview of Part II

This part of the paper assesses the usability of the three Intergraph manuals and offers specific recommendations for improvements. According to Edmond H. Weiss in his book *How to Write a Usable User Manual*, the main functions of user documentation are Motivation, Reference, and Instruction (which includes the sub-groups Tutorial and Demonstration). These are also the primary concerns of Part II, along with somewhat more tangible aspects of documentation such as Layout, Graphics, and Prose.

Part II is divided as follows:

- **Section 1, Motivation,** will concentrate on this key function of documentation that dramatically affects the usefulness of Intergraph's systems and manuals.

- **Section 2, Task Orientation,** will deal with the lack of task orientation in the manuals, as discerned by actual users. Problematic passages will be cited, and possible solutions suggested.

- **Section 3, Reference and Instruction,** will detail the need to divide the information in the manuals according to these crucial functions. "What Is" and "How To" divisions will be discussed, as will separate manuals (or sections) for programmers and operators.

- **Section 4, Style,** will cover problem areas such as Layout, Graphics, and Readability. Included is a mock-up showing how the accessibility of graphics and layouts of the manuals might be improved through the use of a modular two-page format. The results of a readability test to which the manuals were subjected will also be discussed.

1. Motivation

For the user to gain the maximum benefit from the system, he or she must be properly motivated to do so by the documentation. No matter how useful a particular operator command might be in theory, it will be much less useful if not accompanied in the manual by an explanation of *why* it is essential that the user learn to use it. Without being given sufficient explanation, the user often will not bother to read -- much less implement -- the instructions in even the most important passages.

Each of the manuals was found to be deficient in the area of motivating the reader; though many commands and functions are described, they are seldom, if ever, accompanied by explanations of what benefit the user will derive from them. For example, the introduction of the WMS manual (Section 1) warns the user not to begin using the software without first reading Section 1. The section, however, describes none of the *benefits* to be derived from the use of the system (nor do any of the subsequent sections, in any significant detail); Section 1 immediately begins using as-yet-undefined terms and concepts such as "digitizing" and "ellipsoid," yet it gives the reader no means or reasons for learning either the terms or the system itself. The introduction to the GPPU system fares somewhat better in this regard (it defines "digitizing," among other terms, and hints at the system's usefulness), but the Architectural manual lacks an introduction of any kind.

Consider the following paragraph marked "Description" in Section 6.2 of the GPPU manual. The superscript letters have been added to indicate that further motivation and explanation is needed, as detailed immediately following the GPPU passage.

> **ZONE** accepts all IGDS elements, including cells, except for the new 8.8 IGDS elements such as B-spline curve.[a] A separate zone is generated for each element.[b] You can create outside zones (from the element outward), inside zones (from the element inward), or both inside and outside zones.[c] Linear elements and point elements can only have outside zones, since they have no internal area. Area elements can have inside and outside zones.[d] **ZONE** has an option to generate a new entity with a unique identifier and zone distance attribute. For loading the distance value in **ZONE**, the alphanumeric format (AN(20)) must be used.[e]

a.) What are the IGDS elements ZONE accepts, and of what use is this to the user? Why doesn't it accept the B-spline curve, and what are the ramifications?

b.) Is this a task the user must perform? (As worded, the implication is that the system itself is responsible.) *Why* would one want to create a zone for each element?

c.) Why would the user want to create an outside or inside zone? What are the characteristics of each?

d.) Is there any particular reason the user should select linear, point, or area elements? What are the relative benefits of each type of element?

e.) Why would the user want to take advantage of the option for generating a new entity with a unique identifier and zone distance attribute? Are these and the above terms defined anywhere in this manual? If so, the user should be told where to find the definitions. If definitions aren't available, they should be.

The lack of motivation that exists virtually throughout all three manuals was a recurrent theme of the TVA interviews and class discussions. It represents an all-encompassing conceptual flaw that could be remedied by making the manuals less system-based and more task-oriented, beginning with having separate sections and/or manuals for programmers and operators. While we will return to the question of task orientation, note for now that manager Ebeth McMullen and operator David Hankins both requested detailed introductory passages at the beginning of each section or group of commands to explain the significance and uses of the given subject matter. "Our operators need to know *why*," according to Ms. McMullen, "just as much as they need to know *how*." Without this crucial motivation, the user will either miss many of the system's benefits or waste valuable time trying to discover those benefits. In the words of Ms. McMullen, the manuals present "a buffet of things to choose from," without explaining the benefits to be derived from each or "when to use what."

2. Task Orientation

The overriding theme of class discussions regarding the Intergraph manuals was their lack of orientation toward the user's task. While it is quite easy to point a finger that accuses the manuals of being too system-based, the realization of that fact is only the first step toward making the manuals more task-oriented. Rather than follow a logical progression from one task to another, the manuals are arranged for the most part according to similar-command groups, a clear reflection of bias in favor of the system itself over the user.

Section 4, Polygon Creation, in the GPPU manual can serve as a good example of the need to emphasize tasks and how they are interrelated. As it stands, the section falls into five sub-sections that reflect the five polygon creation commands in the system. The section covers thirty-six pages that mix description of the commands with instruction on how to execute them. Unfortunately, the rather thorough detail included gives very little attention to what may well be the most important point -- when should each command be used?

Specifically, Sub-section 4.5 in Polygon Creation consists of a Polygon Check command overview, instructions for calling it up on the screen, sample screens, and instructions for running it in both graphics and nongraphics environments. The following crucial questions go unanswered:

- Why would the user choose one environment over another?
- Is this task (Polygon Checking) used along with others from the same section?
- Is this task used to the *exclusion* of others in the same section?
- What is the relation of this and other tasks in Section 4 to those described elsewhere in the manual?
- In what order should the tasks be performed?

Polygons are apparently used to draw boundaries between characteristics on maps that show the distribution of scientific or social traits over given geographical areas. Why not call polygon creation something like "How to Divide Maps by Area Traits"? The title itself is less daunting and more informative, denoting a shift in focus from the abstraction of the system to the concrete detail of *task*. The section overview could describe in-depth the job to be done, and the sub-sections could give specific examples of the function in specific applications such as forestry, geography, and economics (Swain, p. 6).

To take this task orientation a step further, sub-sections could be added throughout all three manuals to highlight error correction or troubleshooting. Carefully designed flow charts would serve to condense and clarify information that is frequently cluttered, as is the case with WMS Section 2.4, Preliminary Parameters Tutorial, whose lengthy command strings (one of which is 16 steps long) indicate a problem common to all of the manuals. Such command strings present an impenetrable labyrinth to anyone not already well-acquainted with the system, forcing new users to learn by the most inefficient method possible: trial and error.

The TVA employees concurred with the judgment of the class that the manuals must emphasize task rather than system. They agreed that the manuals should proceed from section to section (and from sub-section to sub-section) according to the hierarchy of task rather than according to similarity of commands. Systems analyst Bob Bryant himself proposed using flow charts to replace long command strings, in addition to asking for separate manuals for programmers and operators -- a major point that is examined below in Section 3.

The following are the main issues that the TVA employees raised with regard to task orientation:

- The manuals are cumbersome at best due to their size and flawed organization; in the case of novice operators, the manuals are nearly useless due to their system-based presumption of the reader's familiarity with tasks, terms, and concepts.

- Intergraph's assumption that all users will read the manuals from cover to cover is completely unfounded. At TVA, new operators begin working on the system immediately and consult the manual only if an error message appears on-screen.

- All three TVA employees indicated that new users tend to be "afraid" of a document due to its lack of task orientation and thus are forced to consult experienced users like Dave Hankins, thereby decreasing the productivity of not one, but two employees.

The severe doubts raised about the manuals' suitability for use by inexperienced operators are particularly significant in light of the fact that experienced users at TVA very seldom use the manuals, and then only for reference. This indicates two obstacles that stand in the way of the manuals becoming truly useful: 1) Intergraph's target audience should be novices, who are effectively excluded from using the manuals at present due to the mistaken assumption of familiarity with the system, and 2) the manuals must be made more useful to experienced users.

The solution to both of these problems is to divide the information in the manuals according to two categories: **Reference** (also called "What Is . . . ") and **Instruction** ("How To . . . "). Section 3 of this report, which follows immediately, outlines specific problems and recommended solutions to more clearly define these two crucial functions of user documentation.

3. Reference and Instruction

Weiss points out in *How to Write a Usable User Manual*:

> Today, most vendors and developers provide at least some user documents. Sadly, though, they tend to subscribe to the encyclopedic view of the user manual: Put everything in one big volume and let the users fend for themselves. What they ought to do is *analyze* what documents are needed: align particular publications with the tasks and interests of particular readers . . . That means the developer has analyzed what the users do, how they use the system and product, and what information they need (p. 85).

Bob Bryant, senior systems analyst at TVA, rates the overall usefulness of the WMS manual as "7 or 8" an a scale of 10, but suggested that the reference manual and user manual be separated -- ideally into two separate documents, but at least into two distinct sections, with material for the operator in front and reference material for the systems analyst or programmer in the back. The present combination of reference and instruction produced what Mr. Bryant called "a *very* thick manual," which he felt was "disjointed" due to the lack of distinction between its reference and instructional functions (he found Section 6 of WMS especially guilty of this). "Users need to know how to do something," he said, "while programmers need more detail on how something is done in the machine." Operators are frequently confused by system information that is useful only to programmers or system analysts.

At the same time, however, Mr. Bryant found that the amount of detail the manual offers on how the software interfaces with the machine is inadequate to facilitate the creation of new routines. When asked what he liked least about the manual, Bryant said it was the fact that it called itself a programmer's manual. While the system itself does provide "hooks," ways to gain access to the system to create new routines, the manual did not delve deeply enough into these hooks; it also failed to list files that were originally customized but had been fully integrated into the system since the last version of the manual, according to Mr. Bryant. In its present form, the manual provides *too much* detail for the operator and *not enough* for the programmer.

This applies to the GPPU manual as well. For example, Section 6.5 (Element Overlay Processor) gives information about its operations that is evidently aimed at the operator: how to determine the relationship between points on lines, points on points, points on polygons, etc. But in the very same section -- without being marked or set apart in any manner -- is information on how to "Enter New Information to Database Subtutorial" (p. 6-36). This is apparently similar to the WMS system hooks to which Bob Bryant referred, and it no doubt provides some very useful information for programmers despite the fact that they must wade through irrelevant operations information to find it.

But pity the poor operator who stumbles upon its list of programmer's steps (which appear in exactly the same typeface and layout as the operator's steps that immediately precede them), because if that operator tries to follow those steps he will soon be hopelessly and needlessly confused.

Each of the manuals attempts to fulfill the functions of reference ("What Is") and instruction ("How To") simultaneously, creating a tangle that actually prevents the fulfillment of either function. Both Bob Bryant and Dave Hankins agreed that the manuals themselves should describe "How To" thoroughly in order of tasks performed; furthermore, they said that the on-line help should be pared down to what would amount to a "What Is" reference guide, what they called "a brief refresher" to replace the "complete lessons" currently in the on-line help.

In the words of Mr. Hankins, the manuals simply "try to serve too many kinds of users." Ironically, however, they ignore the very user most likely to need extensive documented information: the novice operator. Because the TVA interviewees said that experienced users tend to use the manuals for reference only, the detailed information in the present manuals is serving more to confuse the uninitiated users, who then waste the time of those with more experience by asking questions.

Specific recommendations for improving the functions of reference and instruction are:

1. Create separate manuals for programmers and operators.

2. Barring this, create separate sections for programmers and operators at minimum.

3. Create separate reference and instruction sections; gear reference toward experienced users, and gear instruction toward novices.

4. Mark sections clearly as to whether they pertain to functions performed by programmers or operators.

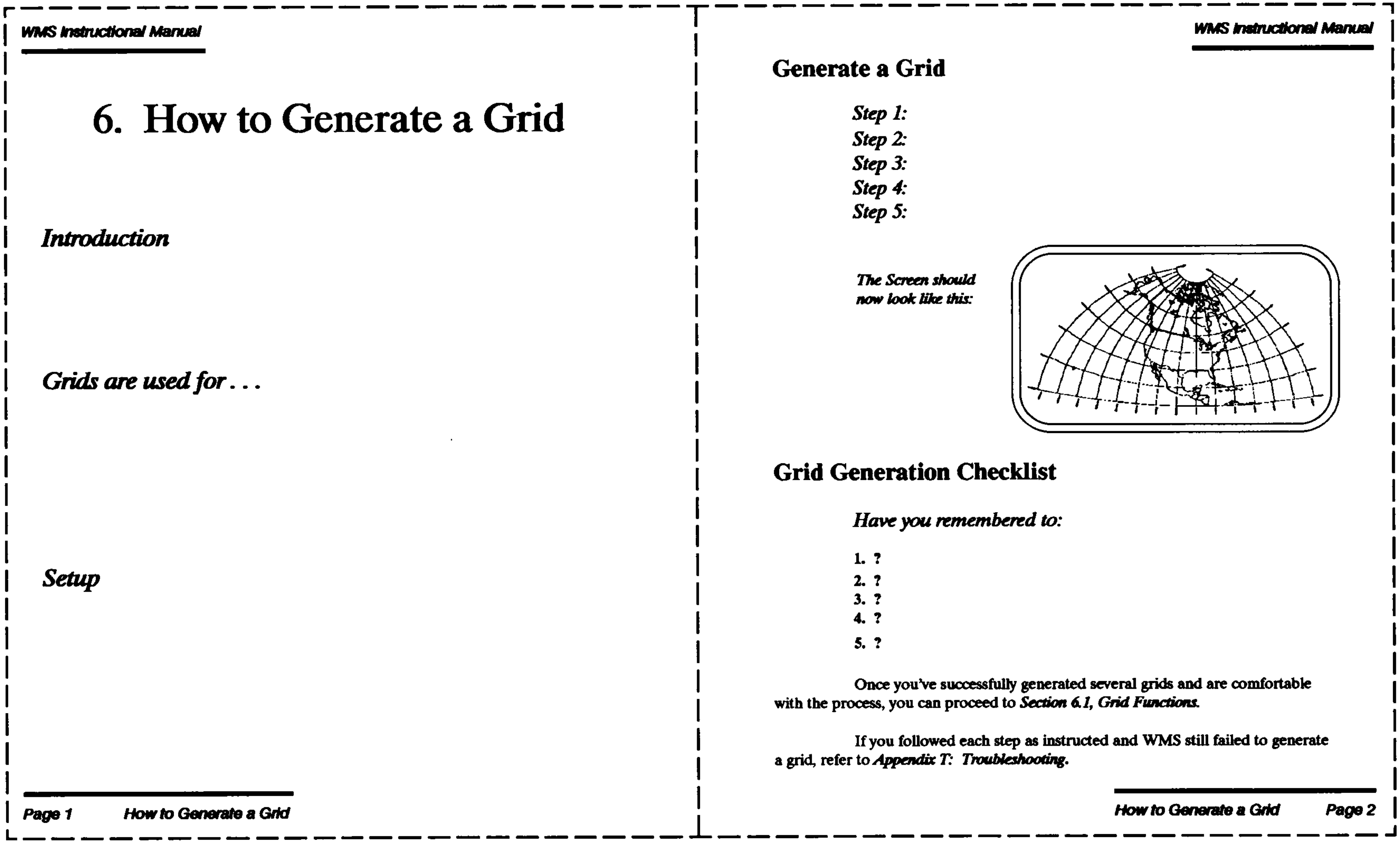
5. Describe in detail potential hooks within the system; list any formerly customized features that are now integrated into the system itself.

4. Style

Layout and Graphics

The manuals' poor design and layout currently reinforces their ill-suitedness to the user's needs. Though the variety of typefaces and graphics in the later manuals (WMS and GPPU) show improvement over those of the earlier architectural document, there remains plenty of room for improvement. Using readily available graphics, Intergraph could easily implement a more accessible format that improves both readability and usability without calling attention to itself. Figure 1 is a mock-up that demonstrates some simple solutions to the manuals' design flaws; one strong recommendation is a switch to a "modular" two-page format as prescribed by Weiss in *How to Write a Usable User Manual.* For purposes of comparison, Figure 2 shows the introductory page from *WMS Section 6.2.*

According to Weiss, the instructional module must present a single concept or task and then test the reader to see if that task has been learned properly. Section 6 of the WMS manual, however, consists of one "procedure" that is 23 steps long (more than tripling the maximum of seven, and nearly quintupling the recommended 5-step limit proposed by Lawson [p. 340]). Such a long procedure simply must be divided into smaller, separate tasks to avoid needlessly confusing the user. The discipline of adhering to the two-page format would help to enforce a clear, task-oriented hierarchy and to winnow out unnecessary passages.



WMS Instructional Manual

6. How to Generate a Grid

Introduction

Grids are used for . . .

Setup

Page 1 How to Generate a Grid

WMS Instructional Manual

Generate a Grid

Step 1:
Step 2:
Step 3:
Step 4:
Step 5:

The Screen should now look like this:

Grid Generation Checklist

Have you remembered to:

1. ?
2. ?
3. ?
4. ?
5. ?

Once you've successfully generated several grids and are comfortable with the process, you can proceed to *Section 6.1, Grid Functions.*

If you followed each step as instructed and WMS still failed to generate a grid, refer to *Appendix T: Troubleshooting.*

How to Generate a Grid Page 2

Figure 1. Mockup of Two-Page Module for *WMS Section 6.*

6.2 The Grid Generation Tutorial

Description

The Grid Generation tutorial (Figure 6-3) allows you to select several different options for grid generation. You can choose between several different types of coordinates for input, and two different ways of incrementing the coordinates for placing the grid elements. A secondary projection option is available.

You can place any combination of grid lines, labels, tick marks, crosses, a frame, and a border. Also, you can specify the level, weight, style, and color for each of these items. You do not have to place all of these elements. In fact, if you select crosses and a grid that have the same deltas, they are drawn on top of each other and you cannot view them both at the same time.

The Grid Generation tutorial allows you to calculate the number of crosses and the sheet size for the output grid before the grid is generated. This allows you to check the results and make any necessary corrections before you actually generate the grid

Setup

Take the following steps before running the Grid Generation tutorial.

- To display the labeling text correctly, the font library must contain FONT037. See Appendix B for the procedure to merge FONT037 into the font library.
- If you want create your own font for FONT037 you can do so by using the standard IGDS font generation procedures. The octal codes for the characters used in FONT037 are as follows:

Octal Code	*Character*
052 (octal)	G (grade)
053 (octal)	o (degree)
054 (octal)	' (minute)
055 (octal)	" (second)
057 (octal)	- (minus sign)

Figure 2. Introduction to *WMS Section 6.2.*

The manuals should be aimed at visually oriented users (who make up its natural audience, it would seem) by using interactive text and illustrations combined in units called modules. Frequently referred-to passages (including figures, definitions, and flow charts) should be printed on fold-out cards for quick and easy reference, and "visual grammar" such as boldfacing, white space, and color should be used to emphasize portions of text or convey a sense of information hierarchy. Bleeding tabs such as those used to distinguish sections of the WMS manual have been proven ineffective and even counterproductive (Ray et al.); Dave Hankins has added colored divider pages and protruding tabs to his copy of WMS, a clear indication that the addition of this sort of retrievability aid by Intergraph would be greatly appreciated. Presumably, less-experienced operators would benefit even more from the use of graphics that illustrate the capabilities of the system, such as sample maps and procedural flow charts that clarify operating sequences.

Two-page modules certainly lend themselves to visuals that reinforce the text that faces them, a form of redundancy that Weiss says is absolutely necessary to ensure effective communication. The module must be able to stand alone without cross-referencing and occupy two facing pages so the reader isn't forced to flip pages frequently (a common complaint of the interviewees was that figures are often on the back of the page that refers to them, necessitating a maddening flip-flop). Modularity would also reduce the size of the manuals, making them more appropriate for the space restrictions of the operating environment at TVA.

Readability

One of the main complaints of the TVA employees with regard to the WMS and GPPU manuals is their unwieldy size: they total 290 and 338 pages, respectively. The manuals are used largely by novices to learn the system, but they're simply too cumbersome for anyone to use for extended periods at the workstation. While addressing the problems of reference and instruction (see Section 3) will help solve this size problem, Intergraph needs also to dramatically alter the wordy prose style that at least partially accounts for the excessive size of its manuals.

Part of the wordiness problem is the level of detail included explaining the system. While added information about *task* would be welcome indeed, the manuals presently go into far too much detail about complex system functions about which the user generally has no need to know. For instance, page 1-12 of the WMS manual informs us what language the program is written in and what type of variable the numbers are stored in, information that is mainly superfluous (see Figure 3). By including so much system-based detail, the manuals are eventually dragged down by their own weight.

Readability is an important feature of any manual but, as alluded to in the literature review of this report, the reliability of techniques for assessing readability remains a controversial subject. Nonetheless, valuable insight was gained by subjecting representative sections of the Intergraph

- The PRO program gives precision coordinate readouts from nongraphic coordinates.
- The WMSCVT subroutine filters points entered in one coordinate system and/or converts them to values in another coordinate system.

WMS provides you with a database in which you can create and store data sets for projection, datum shift, and ellipsoid definitions. Once the projection, datum shift, or ellipsoid data set is stored in the database, you can recall it repeatedly for use in projection definitions without having to memorize or key in the values each time. If you want to use datum shifts, you must create, store, and recall them from the WMS database. This is the only way to enter datum shift definitions. The WMS database operations run from a separate set of tutorials, and are driven by the WMSCODES program.

Error messages issued by the various WMS procedures are described in Appendix C. Causes of the error and suggested recovery procedures are included.

WMS software is written in VAX-11 FORTRAN 77 and C. All WMS calculations are performed in VAX double-precision floating point, which provides 17 decimal digits of precision. X, Y projection values and longitude, latitude values are stored in standard IGDS 32-bit integer format when the design file format is used.

The accuracy of WMS functions in design file coordinates is dependent upon the working units of the file. For non-projected files, the accuracy is .001 of an arc second. For projected files, the map scale along with the working units (that is, the number of UORs per master unit) determine the precision. Increasing the precision by increasing the number of UORs per master unit normally results in reducing the area that can be stored in the design file.

To plot output from WMS procedures, you must determine the correct scale to use. Procedures for determining scale are included in Appendix D.

Figure 3. *WMS* Page 1-12.

manuals to analysis using the reading difficulty equation known as the Gunning Fog Index. In Table 1, the overall total of 13 for the manuals indicates that they were written at a college freshman reading level; the much higher Fog Index ratings of specific sections such as WMS 2.0, however, attest to the inaccessibility of much of the prose in the manuals. Robert Gunning recommends that the reading level of any technical document should not exceed 12 (high school senior level) (p. 36). Based on this criterion, the Intergraph manuals' rating indicates that they cannot support sustained reading.

The totals in the first group of Table 2 indicate what percentage of the sentences have active, linking, and passive verbs. The second group quantifies sentence difficulty, which is divided into simple, complex, and compound sentence structures. The third group indicates the percentage of sentences that employ imperative, second person, or third person verbs. Imperative verbs were defined as those employing the implicit "you" (i.e., commands lacking direct address of the user). Second person verbs, on the other hand, directly address the user. Finally, the subject of most third person verbs encountered in the manuals was found to be the computer system itself; the first sentence in Figure 3 ("The PRO program gives precision coordinate readouts . . .") is just one example of system bias resulting from the manuals' reliance on the third person. The mixture of active, linking, and passive verbs shown in Table 2 is acceptable, although the percentage of passive verbs is a bit high; more active and linking verbs would enhance readability (Gunning, pp. 99-109). The low percentages

Table 1. Review of Sentence Length, Hard Words, and the Gunning Fog Index (Parrott, unpublished, 1987).

Section	*Sentences*	*Words* per sentence	*Hard Words** per sentence	*Gunning Fog Index*
WMS 1.1 (Overview)	7	19.3	4.4	17
WMS 2.0 (Section Introduction)	4	19.5	7.5	23
WMS 2.3 (Tutorial)	51	13.8	3.1	15
WMS 3.3 (Batch Driver)	48	15.8	2.0	11
WMS 6.1 (Section Introduction)	16	17.4	2.1	12
WMS Average/Total	*126*	*15.5*	*2.8*	*13*
GPPU 1.1 (Overview)	8	22.3	5.6	18
GPPU 3.1 (Section Introduction)	19	13.1	3.4	12
GPPU 4.3 (Utility)	34	15.0	2.7	12
GPPU 5.5 (Tutorial)	34	14.2	2.3	12
GPPU 6.6 (Tutorial)	46	14.7	2.8	13
GPPU Average/Total	*141*	*14.9*	*2.8*	*13*
Overall Average/Total	*267*	*15.2*	*2.8*	*13*

*Defined as words with three syllables or more (Gunning, p. 34).

Table 2. Summary of Verb Types, Sentence Structure, and Person (Parrott, unpublished, 1987).

Section	*Verb Type (%)*			*Sentence Structure (%)*			*Person (%)*		
	Active	*Linking*	*Passive*	*Simple*	*Complex*	*Compound*	*Imperative*	*2nd*	*3rd*
WMS 1.1	29	42	29	29	71	0	0	0	100
WMS 2.0	75	0	25	50	50	0	0	0	100
WMS 2.3	90	5	5	41	54	3	70	11	17
WMS 3.3	37	41	21	41	58	0	16	0	83
WMS 6.1	75	12	12	25	68	6	6	25	68
WMS Average	*64*	*21*	*14*	*44*	*59*	*2*	*36*	*8*	*56*
GPPU 1.1	25	25	50	50	50	0	0	12	87
GPPU 3.1	84	10	5	63	36	0	5	0	94
GPPU 4.3	88	3	8	18	71	12	56	0	44
GPPU 5.5	70	12	17	21	74	6	32	0	61
GPPU 6.6	57	4	39	15	83	2	39	0	61
GPPU Average	*55*	*8*	*23*	*21*	*70*	*5*	*35*	*1*	*65*
Overall Average	*61*	*17*	*21*	*33*	*64*	*4*	*35*	*4*	*61*

of simple and compound sentences and the high percentage of complex sentences indicate that the prose is too wordy, a fact that is also attested to by the high sentence-length figures in Table 1.

Not a single first-person sentence was found in the sections chosen for the Gunning analysis, which speaks volumes about the impersonal tone of the manuals; they reinforce the impression that Intergraph is a monolithic company, unable or unwilling to display empathy for the user. That only 4 percent of the verbs were second person further indicates that the tone is very harsh, and passages including instructions invariably use the imperative voice -- this is likely to evoke both anxiety and hostility in the reader. The remaining 61 percent of the sentences are in the impersonal, system-based third person.

Mr. Hankins summed up the users' attitudes to the manuals' readability by saying that "none of the manuals are easily understood." Not only does the difficulty of the prose prevent efficient use of the manuals, it also virtually guarantees that they will not be read in their entirety as Intergraph intends. This reinforces the contention of the TVA interviewees that *no one* reads the manuals from cover to cover. While each of the changes suggested throughout Part II of this report would have a salutary spillover effect on readability (e.g., increased graphics and white space would inevitably necessitate streamlining the prose), much more direct emphasis on avoiding wordiness is crucial to improving the usability of all three Intergraph documents.

Conclusion

The field usability test of the three Intergraph manuals suggests many areas for improvement, most notably the need for greater task orientation and for clear distinctions to be drawn between the functions of Reference and Instruction (preferably via the creation of separate manuals). The manuals' graphic design and layout could easily be enhanced by varying typefaces and by offering more detailed and appropriate illustrations; the implementation of two-page modules would do much to improve both the layout and the overall organization. Finally, the manuals' prose could be carefully re-worked to enhance readability and to ensure that the level of reading difficulty matches Intergraph's target audience.

In a broader sense, the issues raised in Part II are also applicable to a wide range of technical documents. The solutions offered are based on sound principles gleaned from external sources aimed at a general audience, and the concerns that they address are universal in the field of document preparation. Although the greatest benefit to be derived from Part II is likely to be in the area of computer manual preparation, the usability of virtually any document could be improved by judiciously applying the concepts and prescriptions contained in both Part I and Part II.

The field usability testing method used in Part II is a technique that, unlike the potentially expensive laboratory method, is an option available to nearly any company preparing a technical document. The purpose of this part of the paper, however, is to confront the specific variables affecting the usability of the Intergraph manuals rather than to describe the testing methods that were used. (See the passage in Part I that deals with Doheny-Farina's article for an in-depth description of field testing procedures.) The relative advantages and disadvantages of the two testing methods should be explored at length in the future, for it may well be that the less-costly field method yields equally valuable insights.

Taken as a whole, the two parts of this paper represent an attempt to expand the general themes in the existing testing-methodology literature to include more specific information about how to heighten usability. Only by integrating these concerns can a truly comprehensive view of the field be achieved. If testing techniques continue to evolve as slowly as they have thus far, future essays on the subject are likely to be simply redundant if they concentrate on methodology to the exclusion of studying the specific variables whose importance is stressed throughout this paper.

Bibliography

Bibliography

W. A. Arnold. "Learning Modules in Minimalist Documents," in *Proceedings of the 35th International Technical Communications Conference*, May 1988, pp. WE-16-19.

M. A. Atlas. "The User Edit: Making Manuals Easier to Use," *IEEE Transactions on Professional Communication* **24**: 1 (1981) pp. 28-29.

M. Bednarz and P. Berger. "Storyboards: Make It Real, Make It Happen," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, p. 309.

___________. "Storyboarding: Make It Real and Make Things Happen," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, p. VC-23.

C. Berkenkotter. "Understanding a Writer's Awareness of Audience," *College Composition and Communication* 32 (1981) pp. 388-97.

C. M. Bibus. "Seven Steps to Overcoming Problems in Visual Organization," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 318-21.

__________. "The New Visual Techniques: Learning and the Page," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-25-28.

R. J. Brockmann. *Writing Better Computer User Documentation*. New York: J. Wiley, 1986.

B. Bryant. *Personal Interview*. Norris, Tennessee: December 1986.

H. H. Clark and S. E. Haviland. "Comprehension and the Given-New Contract," pp. 1-40 in *Discourse Production and Comprehension*, R. O. Freedle, ed. Norwood, N.J.: Ablex., 1978.

N. E. Cohen. "Graphic Design 101: A Short Art Course for Writers and Editors," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, p. VC-11.

__________. "Graphic Design Do's and Don't's," in *Proceedings of the 35th International Technical Communications Conference*, May 1988, p. VC-103.

S. Doheny-Farina. "Usability Testing in the Field," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 252-55.

K. M. Doyle and P. L. Jones. "Modularizing Software Documentation," in *Proceedings of the 35th International Technical Communications Conference*, May 1988, pp. WE-49-50.

L. L. Griffith. "Job Aids: When a Manual is Too Much or Not Enough," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. WE-6-9.

D. H. Grimstead. "Quality Graphics: Writers Draw the Line," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-66-69.

R. Gunning. *The Technique of Clear Writing*. New York: McGraw-Hill, 1952.

D. Hankins. *Personal Interview*. Norris, Tennessee: December 1986.

J. S. Hill. "Language, Art, and Documentation: Exploring the Visual Process for Greater Audience Comprehension," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-93-96.

R. W. Hornbeck. "Common Cognitive Mistakes in Technical Writing," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 315-17.

W. Horton. "Toward the Four-Dimensional Page," in *Proceedings of the 32nd International Technical Communications Conference*, May 1985, pp. RET-83-86.

________. "Templates of Thought," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 302-5.

J. Hutton. "The Character and Its Role in Message Conveyance," in *Proceedings of the 33rd International Technical Communications Conference*, May 1987, pp. VC-41-43.

J. L. Iler. "The CAMS Usability Laboratory," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-81-83.

Intergraph Corporation. *Architectural IGDS Commands User's Guide*. Huntsville, Alabama: 1986.

Intergraph Corporation. *Graphic Polygon Processing Utilities User's Guide.* Huntsville, Alabama: 1986.

Intergraph Corporation. *World Mapping System User's Guide.* Huntsville, Alabama: 1986.

P. Lawson. "Job Aids: Give the Readers What They Want," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 339-42.

P. Lillies. "The Readability of Technical Manuals," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. WE-13-16.

E. McMullen. *Personal Interview.* Norris, Tennessee: December 1986.

M. A. Moore, J. M. Eyre, and T. B. Rideout. "Testing Software Installation Procedures: An Integrated Approach," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 377-86.

P. I. Mulcahy. "Testing Documentation Using Protocol Analysis," in *Proceedings of the 35th International Technical Communications Conference*, May 1988, pp. WE-27-29.

R. H. Nielsen. "A Library of Graphic Devices," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-53-56.

E. K. Odescalchi. "Productivity Gains Attained by Task-Oriented Information," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, p. 359.

L. L. Oestreich. "Mind-Mapping: The Road to Well-Organized Technical Publications," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. RET-66-69.

R. C. Parker. *The Aldus Guide to Basic Design.* Seattle: Aldus Corp., 1987.

C. Parrott. "Computer Manuals, Audience, and Wheels." Unpublished report, 1986.

R. G. Ray, A. Meyers, and K. Gomoll. "Effectiveness of Tabs and Dictionary Heads as Retrievability Aids," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. RET-45-47.

J. C. Redish. "Integrating Art and Text," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-4-7.

P. Reitman. "Quick References: Documents in the Fast Lane," in *Proceedings of the 35th International Technical Communications Conference*, May 1988, pp. WE-93-96.

P. M. Rubens. "Question Type and User Performance," in *Proceedings of the 35th International Technical Communications Conference*, May 1988, pp. WE-151-54.

H. N. Shirk. "A New Look at 'White Space'," in *Proceedings of the 35th International Technical Communications Conference*, May 1988, pp. WE-130-33.

A. Simpson. "Task Oriented Writing: Using Action Sentences to Get the Job Done," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 446-49.

C. Soderston. "The Usability Edit: A New Level," *Technical Communication* 32: 1 (1985) pp. 16-18.

S. E. Spragg. "Symbols, Schemas, and Mind Maps," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. RET-74-76.

J. H. Spyridakis. "Authors' Plans Meet Readers' Plans," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. RET-70-73.

S. Swain. "An Approach towards Task-Oriented Design in Intergraph User Manuals." Unpublished report, 1986.

H. E. Vogt. "Graphic Ways to Eliminate Problems Associated with Translating Technical Documentation," in *Proceedings of the 33rd International Technical Communications Conference*, May 1986, pp. 330-33.

S. Watzman. "Visual Literacy and Document Productivity," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. ATA-48-50.

__________. "The Approachable Page," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-85-87.

E. Weiss. *How to Write a Usable User Manual.* Philadelphia: ISI Press, 1985.

M. Willoughby. "Information Design: Determining Access-Aid Tradeoffs," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-89-92.

W. D. Witman. "Using Reading Protocols to Edit Documents: A Case Study," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. RET-159-61.

G. Wyman and L. Rothberg. "Why Typestyles Convey Mood and Image: A History," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-37-40.

__________. "Selecting Type for Mood and Image," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. VC-41-43.

M. A. Zirinsky. "Usability Testing of Documentation," in *Proceedings of the 34th International Technical Communications Conference*, May 1987, pp. WE-10-12.

Vita

Thomas J. Garritano was born in Hinsdale, Illinois, on December 30, 1958. He attended Gower Grammar School in that town and was graduated from Hinsdale South High School in June 1976. After alternating between attending college and working full-time for several years, he received a Bachelor of Arts degree in English from Northern Illinois University (NIU) in May 1986. In the fall of that year he accepted an assistantship to study Technical Communication at the University of Tennessee, Knoxville (UTK), and was subsequently awarded a Master's degree in August 1988.

Mr. Garritano is a member of the Society for Technical Communication (STC), and he has held several writing and editing positions in both Illinois and Tennessee. At NIU he served as assistant entertainment editor and principal features writer for the student daily paper. During his first year of study at UTK, he was employed as an editor for JBF Associates, an engineering consulting firm located in Knoxville. He also served a three-month internship as an editor with the Information Resources Organization at Oak Ridge National Laboratory, a Department of Energy outpost managed by Martin-Marietta Energy Systems, Inc. His second year at UTK was spent as technical writer/editor for the Science Alliance, one of the State of Tennessee's designated Centers of Excellence.