Readability concerns live on within several approaches to documentation quality.

Readability and Computer Documentation

Gretchen Hargis
Senior Software Engineer
IBM Corporation
Santa Teresa Laboratory
555 Bailey Avenue
San Jose, CA 95141
ghargis@us.ibm.com

Abstract

Traditional readability concerns are alive and well, but subsumed within several more recent documentation "quality" efforts. For example, concerns with interestingness and translatability for global markets, with audience analysis and task sufficiency, and with reader appropriateness of technical text all involve readability, but often in ways not easily measured by any formula.

***I.7.5** Document analysis—human factors*
***Keywords**: task sufficiency, minimalism, quality measurements*

> Have you seen this joke on the Web? High on the wish list of a reader (or maybe a writer!) is this computer command:
>
> reading_is_fun
>
> All the dry documentation would now be fun to read and understandable (Hubbard, 1997).

Not quite the means of achieving readability that George Klare had in mind when he published *The Measurement of Readability* in 1963. And yet in the almost 40 years since Klare summarized and interpreted the readability research, the computer has greatly affected not only our understanding of what readability is but also the need for readability (to deal with masses of computer documentation) and our ability to create and measure readable information.

Even though computer documentation has often been lampooned as hard to read, technical communicators have not seized on readability formulas to help with the problem. Producers of software and hardware have not added readability scores for their documentation as a marketing device or even as a major quality initiative. Does this mean that we technical communicators have avoided working on the readability problem in computer documentation? Not at all. I believe that we have absorbed several lessons from the readability debates and continue to apply these lessons to improve what we write. This paper deals with these lessons, particularly from the viewpoint of a writer and editor of IBM product documentation since the early 1980s.

Lesson 1: Levels of Readability

Dating from 1921 when Thorndike published a list of the most frequently used English words, modern research on readability crosses many disciplines, including education, psycholinguistics, rhetoric, cognitive psychology, artificial intelligence, human-computer interaction, and technical communication. In the early 1920s, Gray and Leary started with 289 possible variables involved in reading and grouped them

ACM Journal of Computer Documentation, 2000,24:122-131
 ISSN 1527-6805

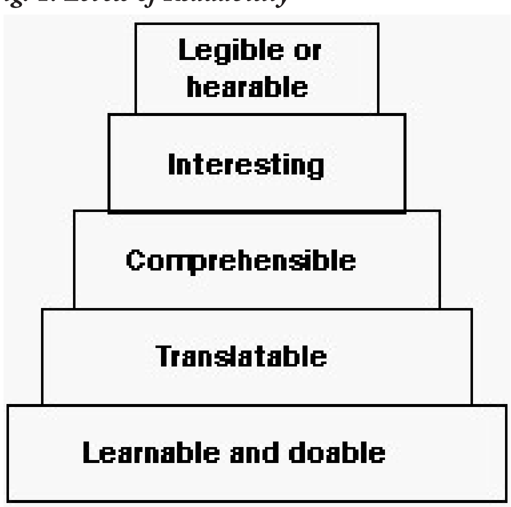
into four categories (Klare, 1988):

- Content
- Style of expression and presentation
- Format
- General factors of organization

What could easily be measured (by eye and hand) reduced the purview of the readability measurements to a few variables: notably the number of words in a sentence, number of syllables in a word, and percentage of unfamiliar words based on certain lists. Reducing readability to these measurements (sort of like putting a cloud into a bottle) gave rise to many criticisms, including mention of the many variables that the measurements did not (because they could not) measure, such as word order, sentence clauses, and illustrations.

I find it useful to think about the interaction of factors in readability as levels (see Figure 1). Each level makes it possible for a person to go deeper into the communication.

Fig. 1. Levels of Readability



Readable as Legible or Hearable

At the surface level, readability is legibility or hearability, depending on whether the communication uses the eyes or ears as the pathway to the brain. This aspect of readability has generated a lot of research, especially when information moved from paper to the computer screen. Layouts, colors and contrast, and type needed to be rethought amid the new online possibilities, including dynamic changes in any of these by users. Technical communicators involved with online information or assistance need to pay attention to the factors that affect reading or hearing online.

Already in 1963 Klare acknowledged research in listenability, but a major stumbling block was that the hearer could not replay the information. With a computer, however, the hearer can request audio help as needed and replay it. So script writing becomes a possible skill for a technical communicator to develop as a means of meeting users' needs for assistance online.

Most readability research has assumed that words are the vehicle for transmitting content. With improved computer displays, storage capacity, and transmission speeds, images including icons and animation can play a bigger role in communicating technical information. The user can also take a more active role in structuring and manipulating the communication than a reader could do with text in a book.

Readable as Interesting

At the next level, the major consideration for readable material is whether it is interesting, as the reading_is_fun command implies. When material is dull, a person must make an extra effort to stay focused on it. However, the capacity to arouse interest or hold attention does not necessarily inhere in a piece of writing. This characteristic depends on the particular person interacting with a particular subject, perhaps at a particular time. The answer to today's computer problem may be less interesting tomorrow.

Many technical communicators have taken the easy way out in terms of this type of readability. We decide that a programming language, for example, is dull (unless it's Java!) and there's no way to make it interesting. We may also decide that there's no need to make it interesting. The people who need the information have no choice but to read what's offered, and any techniques to make the information interesting may even be resented as an affront to the user's professionalism. A computer manual should not try to be a bestseller, right? Some writers of product documentation maintain this position even though some retail computer books sell well and receive kudos from users (albeit usually novices) for their helpfulness and understandability.

Readable as Comprehensible

In 1963 Klare described readability as the ease with which material can be read but not necessarily the ease with which it can be understood. The research at that time did not clearly show that increased readability correlated with increased comprehension. It took a couple more decades of readability research and use of readability formulas for readability to become synonymous with comprehensibility (Klare, 1988).

The most used traditional readability formulas (such as Flesch Readability Ease and the Fog Index) are calibrated to grade levels but at only 50% to 75% comprehension. This level of comprehension is enough in a classroom where a teacher expects to provide instruction on the material to help raise the students' understanding. However, many users of computer information depend on what they read as their main or sole source of information, and they need to reach 90% to 100% comprehension (Powell, 1981). To facilitate higher comprehension, technical communicators need to use a lower readability target than the grade level that they expect typical users to have attained.

Comprehension here is a catchall even in an educational setting. Other cognitive behaviors important in learning are knowing, applying, analyzing, synthesizing, and evaluating (Bloom and Krathwohl, 1984). In terms of computer documentation, applying is so important that it warrants separate consideration.

Readable as Translatable

Readable information is desirable and important today for the translator and also for the nonnative user of English. Compared with the number of native users of English (400 million people), almost twice as many people worldwide use English as a second language or as a foreign language, usually for business purposes. With increasing globalization, many more businesses need to offer appropriate material to people either in their language or in English that they can understand. Many of the issues that readability research has dealt with are important at this level in terms of removing ambiguity and complexity.

Readable as Learnable and Doable

Even comprehending 90% to 100% of a communication is not enough for users of computer documentation. The emphasis on readability as comprehensibility comes from an educational setting, where tests are used to determine how well a student understands or remembers what was read. But a person who needs, for example, guidance on a technical task faces putting the information into immediate use by performing the task, not by responding to test questions. The proof of learning in this case is in the doing. A user may be able to decipher the information, find it interesting, understand it, but still not be able to use or apply it. At this level, readability blurs into task orientation and usability.

Lesson 2: The Writer in Cooperation with the User

The importance of the audience is not new to technical communicators, but readability provides another way to filter this awareness of the audience.

The first important principle of readability that Klare puts forward is that "readable writing is desirable and important for the reader's sake. If it is not readable to an intended reader it is not readable, no matter how good a formula score it may receive." (Klare, 1963, p.11). Most technical writers would agree with this. In fact, some of us have probably voiced such sentiments while resisting attempts to enforce meeting a numerical standard for readability.

Knowing Our Audience

Klare backs up the principle of the importance of the reader with a list of what the writer should know about the typical reader:

- Educational and intellectual levels

ACM Journal of Computer Documentation, 2000,24:60-63
 ISSN 1527-6805

- Background or previous experience with the writer's topic
- Interests
- Level of motivation
- Voluntary or involuntary attention to the material.

Because Klare is considering a general American audience, including children, this list is different from the characteristics that technical writers in the software industry use for their audience. We tend to consider education and experience but to discount interests and voluntary or involuntary attention and to add characteristics like tasks and work environment. We also think of our audience as users rather than readers. What does the user need to do? What kind of access does the user have to the information? How stressful is the work environment?

In terms of the motivation of people using computer information, we understand that many users want to use the software to do their own tasks immediately and so consult our information only if they get into trouble. Among experienced users, there is almost a sense of stigma attached to consulting computer information. Did users stop turning to the information because they found it too hard to read and too dull? Or was it too unrelated to their problem at hand? Some studies show increased readership for documentation based on minimalism, which seeks to adapt the information to people's strategies for learning to use software (Merritt, 1997).

Our methods for coming to know our users also go beyond filling in certain classifications such as experience and tasks. Through usability testing we can observe people actually using our information as part of the product and refine our understanding of how they work. In the course of user-centered design, we can build a composite picture of our users. This process also helps writers who prefer to imagine a person for whom they are writing rather than consider just a set of attributes.

Specialized Audience

Klare acknowledged an easy way out of having to struggle with readability, one that I think the writers of some computer documentation have taken: "If [the writer] is interested only in a small, specialized, highly educated audience, the principles of readability presented here may not be of great concern." Some writers have argued that complex products necessarily have complex documentation and that the users have whatever it takes to deal with the complexity. However, many such users are reaching the age of retirement, and the new users seem much less tolerant of material that is hard to understand and use.

Lack of Homogeneity

The aspect of new users replacing old ones brings us to an issue that Klare also acknowledged: "The audience should be fairly homogeneous in these respects [education, background, and so on] for greatest effectiveness in writing." However, the audience for hardware and software documentation is seldom homogeneous. For one thing, we often need to make the information serve both an experienced user and a novice. Sometimes the experienced user has experience with a similar product, which might use different terminology and approaches. With the globalization of many companies and use of the Web, we have an international audience to consider as well. Research on cross-cultural characteristics such as individualism and collectivism (Hofstede, 1980), high context and low context communication (Hall, 1973), and universalism and particularism (Trompenaars, 1998) is now being applied to technical communication (Honold, 1999; Thatcher, 1999).

The Writer's Purpose

Klare puts forward several possibilities for the writer's purpose in relation to the use of readability principles or formulas, such as helping the reader understand, learn, or read efficiency. He also treats the writer's purpose separately from his description of the audience as if the writer and the audience were on parallel tracks. However, as technical communicators, we tend to derive our purpose for writing something from the user's purpose for reading it. It's not enough that the reader be able to understand, learn for the sake of learning, or read faster. Such purposes in the mind of the writer are more appropriate for a student audience than for an audience of users of computer products and computer documentation.

For technical communicators working on computer documentation, our purpose is to support the users' tasks in using the product. The ultimate test is whether the information or assistance satisfies the users' needs, preferably before they get frustrated with the product. We want the right information to be in the right place at the right time, which is probably in the product interface, so that users do not have to turn to a separate reservoir of online or printed information.

Lesson 3: Readability Pitfalls in Writing (or How Long Is Too Long?)

Many people agree that long sentences tend to interfere with readability in that readers have a harder time processing the pieces (especially subordinate clauses) and even remembering them by the end of a long sentence. Some researchers have pointed out, however, that breaking up sentences can not only produce a choppy, monotonous style but also interfere with understanding by removing relationships between clauses (such as cause-effect, sequence, condition-result, alternatives, assertion-reason, or assertion-example).

Writers who use readability measurements must understand that these measurements make calculations based on *average* sentence length, even though various word-processing tools will point out sentences over a certain length (which the writer may be able to set). Such tools give the impression that any sentence over a certain length (such as 25 words) should be changed, and yet the writer may well decide that some long sentences are indeed understandable and necessary.

Various people offer classifications and analyses of sentence length, two of which are shown in Table 1.

Flesch broke down the differences very precisely, using increments of three words for the easier levels and four for the difficult levels and stopping at 29 words. The numbers in Hartley's guidelines overlap at the ends, and the guidelines use increments of 10 and categorize also sentences longer than 30 and 40 words. For these reasons these guidelines give the impression of being rather loose, but for the purposes of technical communication (unlike writing for newspapers and magazines, which Flesch focused on) loose guidelines may be enough.

Gunning (1968) did not establish ratings for lengths of sentences. He even argued against a limit on sentence length, saying that a sentence can be any length and still be easily

Table 1. Comparison of Recommendations for Average Sentence Length

Source	Average Sentence Length (in words)	Description
Flesch Reading Ease score (Flesch, 1951)	8 or fewer 11 14 17 21 25 29	Very easy Easy Fairly easy Standard Fairly difficult Difficult (academic, scholarly) Very difficult (scientific, professional)
General guidelines (Hartley, 1981)	20 or fewer 20-30 30-40 40 or more	Probably fine Probably satisfactory Suspect Almost certainly benefit from rewriting

Table 1. Comparison of Recommendations for Average Sentence Length

understood if properly handled. Gunning put more emphasis on the nature of the words chosen. The words should be concrete and preferably create pictures that people easily recognize. Choosing such words is a major challenge in technical documentation, however, and so longer sentences can become a liability.

Under pressure to reduce the number of words in a sentence, writers sometimes cut out little words such as prepositions, *that* introducing a clause, and parts of verbs. Research has since identified some of these words as semantic cues and shown that they make a difference in how easily translators and nonnative speakers of English understand a sentence (Kohl, 1999).

The studies on sentence length should not be taken as substituting rules for judgment. Rather, our increased knowledge of the impact of sentence configurations on readers (including translators and nonnative speakers of English) should help writers make decisions about the best way to present information in particular situations.

IBM's Role in the Readability Saga

Partly because of its management policies and long emphasis on research and quality, IBM has helped the cause of readability. However, readability measurements have not been a major rallying point for technical communicators at IBM. We have struggled and continue to struggle with the problems that are associated with readability.

Eschewing Gobbledygook

Gunning (1968) consulted with IBM executives on clear communication in business. Perhaps this management briefing from Thomas J. Watson, Jr., when he was chairman of IBM shows Gunning's influence, a few years before the plain English movement gained momentum to end gobbledygook in government and business documents:

> A foreign language has been creeping into many of the presentations I hear and the memos I read…. I want your help in stamping it out. It's called gobbledygook…. Nothing seems to get finished anymore—it gets "finalized." Things don't happen at the same time but "coincident with this action." Believe it or not, people will talk about taking a "commitment position" and then because of the "volatility of schedule changes" they will "decommit"…. IBM was built with clear thinking and plain talk. Let's keep it that way (Watson, 1970).

Watson could have chosen examples from some documentation for IBM products of the time. However, readability of computer documentation was not a big concern yet. Even computer documentation was not much of a concern in that the people using it (apart from computer operators) were mainly computer professionals, people who seemed even to enjoy flexing their intellects on difficult material. Also, many of the people writing the documentation were computer professionals themselves; and accurate, complete documentation of product functions was as much as they expected.

Building Readability Tools

IBM developed an early grammar and style checker called EPISTLE, which was intended to help stamp out gobbledygook as well as poor constructions in business communications. In the 1980s, linguists and specialists in artificial intelligence at the IBM Thomas J. Watson Research Center expanded EPISTLE into a tool (called Critique) that technical writers could use. In addition to getting a Flesch-Kincaid score (an adaptation of the Flesch Reading Ease score for use by the U.S. Department of Defense), writers could specify which kinds of errors they wanted to know about, such as:

- Improper forms of verbs and pronouns
- Confusing words and awkward phrases
- Subject-verb agreement
- Punctuation and spelling.

Critique also provided information specific to technical writing such as infinitives and gerunds as subjects, passive verbs, and noun strings.

Another IBM tool, READABLE, offered counts of words and sentences, pointers to abstract words (such as *support*, *sequence*, and *modification*) and passive constructions, and counts of prepositions and frequently used words. One could, for example, find out how many times *is* occurred in a chapter! It could also give five readability scores, including the Flesch Reading Ease score, Fog Index, and Flesch-Kincaid score.

In the last few years, Critique has metamorphosed into a tool that points out problems that make text hard to translate. Rather than producing a readability score, this tool gives a clarity index as an indicator of readiness for translation.

Meeting the Needs of Users and Learners

Content is an aspect of readability that is much harder to measure than the length of sentences and words. What to include or not include in computer documentation involves not only investigation and testing while developing a product but also applying principles gained through experience and research. IBM has nurtured two major approaches to the content question: task orientation and minimalism.

Task orientation grew out of the experience of technical writers at IBM during the 1970s and was formalized as an internal guideline in 1982 (IBM, 1982). Through many conference presentations and articles, "by 1994, IBM's task orientation design principles had become an integral element of technical communication's paradigm" (Brockmann, 1998). The principles of task orientation involve understanding the audience, specifying the user's tasks, and organizing the information around sets of instructions with supporting conceptual and reference information.

Minimalism grew out of research by cognitive psychologist John Carroll and his colleagues at the Watson Research Center during the 1980s. Using protocol analysis to determine how people learn to use software, they observed aspects of both the software and the documentation that did not support users in the way that the users tried to learn (Carroll, 1998). Obviously, the problem required a bigger solution than lowering the readability score. They found that the documentation needed to be reengineered, cutting away much information and adding information to help users get started quickly on realistic tasks and to give them pointers when they made mistakes. The IBM Information Development group did not embrace minimalism with a design guideline devoted to it, however. Instead, writers in various areas implemented parts of minimalism as they saw fit from the research articles then available. One group created a tutorial tool that let users do lessons in the product (not a simulation) and monitored their progress and provided a "show me" function for users who wanted to see how to do a task (Chiang and McBride, 1992).

Improving and Measuring Quality

During the 1980s, managers of software development groups were very interested in quality and quality measurements, partly because they expected that getting defects out of the software early would result in increased customer satisfaction. Dealing with readability defects seemed to offer the opportunity to improve the information and measure the improvement.

Several writing groups at IBM worked on quality models for software documentation. They developed a set of characteristics by which to judge the quality of what was at that time mostly manuals. One such model used six characteristics: accuracy, appearance, extraneous information, missing information, readability, and usability (IBM, 1985a). This model defined readability defects as "flaws in the information that result in its being hard to read. Grammatical complexity, overuse of jargon, and poor uses of graphics (hard to read, or not pertinent) are all examples of readability defects. Poor audience descriptions may also lead to readability defects if they cause the information to be written at the wrong level of technical knowledge or educational background." In this model, extraneous information and missing information cover some aspects that both task orientation and minimalism deal with.

Readability defects were monitored from early in the process, when too many new terms might show up in planning documents. Many more readability defects were possible after the actual writing began, such as:

- Lack of headings and bulleted lists
- Too many or too small labels on illustrations
- Incorrect spelling, grammar, or punctuation
- Vague terms
- Acronyms not defined.

Long sentences or words longer than three syllables do not show up in the litany of ways to make something hard to read, although readability formulas focused on measuring these aspects of a piece of writing. Readability in IBM documentation had already encompassed a much larger set of concerns than what could be relatively easily measured.

Readability in IBM documentation had already encompassed a much larger set of concerns than what could be relatively easily measured.

Another quality model that some writers at IBM used had five characteristics: retrievability, readability, task-supportiveness, accuracy, and user satisfaction (IBM, 1985b). Here readability was defined as the information's "ability to be read and understood. Readability depends on things that affect the readers' eyes and minds. Type size, type style, and leading affect the eye. Sentence structure and length, vocabulary, and organization affect the mind." This model acknowledged task-supportiveness (or task orientation) as a separate consideration.

Another group of writers at IBM used a quality model that did not include readability per se. Instead it used these characteristics: task orientation, organization, entry points, clarity, visual communication, accuracy, and completeness (IBM, 1983). Aspects of readability were included in task orientation (which depends on knowing the audience), clarity (which includes sentence structure and choice of words), visual communication (presentation of the

Table 2. Quality Characteristics of Technical Information	
Easy to Use	
Task orientation	Helping users complete tasks associated with a product in relation to their jobs
Accuracy	Freedom from mistake or error; adherence to fact or truth
Completeness	The inclusion of all necessary parts—and only those parts
Easy to Understand	
Clarity	Freedom from ambiguity or obscurity
Concreteness	Inclusion of appropriate examples, scenarios, similes, and analogies
Style	Correctness and appropriateness of writing conventions and choices of words and phrases
Easy to Find	
Organization	A coherent arrangement of parts that makes sense to the user
Retrievability	Presentation of information in a way that enables users to find specific items quickly and easily
Visual effectiveness	Attractiveness and enhanced meaning of information through use of layout, illustrations, color, type, icons, and other graphical devices

Table 2. Quality Characteristics for Technical Information

information), and completeness (including information that users need but not information that they don't need). This quality model has been enhanced as a result of both focus groups with users of software and use of the model to measure quality through editing (Wilde and Colvin, 1996). With the addition of style and concreteness, grouping of the characteristics, and changes in a couple of names, this model (Table 2) has been published for use outside IBM (Hargis et al., 1998).

This multidimensional approach to quality clarifies how the many aspects of technical information fit together (Smart, 1999), including aspects that come under the umbrella of readability.

Looking Backward and Forward

Readability research has helped make technical writers very aware of the connection between what they write at a word and sentence level and the ability of our readers to understand and act on that information. For almost 20 years, several readability measurements have been available in grammar and style checkers and have now infiltrated word-processing programs. Technical writers have accepted the limited benefit that these measurements offer in giving a rough sense of the level of difficulty of material.

We have also assimilated readability as an aspect of the quality of our information through its pervasiveness in areas such as task orientation, completeness, clarity, style, and visual effectiveness. We have put into practice, through user-centered design, ways to stay focused on the needs of our audience and their problems in using the information or assistance that we provide with computer products.

Now we need to become more attuned to the needs of an international audience and to making better use of the computer (such as through audio and animation) to assist users. Perhaps readability research on deeper levels of communication, such as organization, will help us rethink what we do in these areas of usability and quality.

References

Bloom, B. S. and Krathwohl, D. R. (1984). *Taxonomy of educational objectives, handbook 1: Cognitive domain*. Reading, MA: Addison-Wesley Publishing Company.

Brockmann, R. J. (1998). Information transfer in technical communication. In Carroll (Ed.), *Minimalism beyond the Nurnberg funnel* (pp. 375-392). Cambridge, MA: The MIT Press.

Carroll, J. M. (1998). Reconstructing minimalism. In Carroll (Ed.), *Minimalism beyond the Nurnberg funnel* (pp. 1-18). Cambridge, MA: The MIT Press.

Chiang, A., and McBride, K. (1992). TIRS tutorial: New techniques in guided exploration. In *Proceedings of the 37th International Technical Communication Conference, May 10-13* (pp. 327-330). Arlington, VA: Society for Technical Communication.

Flesch, R. (1951). *How to test readability*. New York: Harper & Brothers.

Gunning, R. (1968). *The technique of clear writing (*revised ed.). New York: McGraw Hill Book Company.

Hall, E. (1973). *The silent language*. New York: Doubleday Books.

Hargis, G., Hernandez, A. K., Hughes, P., Ramaker, J., Rouiller, S., and Wilde, E. (1998). *Developing quality technical information*. Upper Saddle River, NJ: Prentice Hall PTR.

Hartley, J. (1981). Eighty ways of improving instructional text. *IEEE Transactions on Professional Communication*, *PC-24* (1), 17-27.

Hofstede, G. (1980). *Culture's consequences: International differences in work-related values*. Newbury Park, CA: Sage Publications.

Honold, P. (1999). Comparison between German and Chinese users. *Technical Communication*, *46* (2), 196-205.

Hubbard, V. (1997). Commands we wish could rule. http://www.fastlane.net/~camnet1/future.shtml.

IBM. (1982). *Publications guidelines: Designing task-oriented libraries for programming products*. Poughkeepsie, NY: IBM Corporation.

IBM. (1983). *Producing quality technical information*. San Jose, CA: IBM Corporation.

IBM. (1985a). *Information development quality improvement program*. Kingston, NY: IBM Corporation.

IBM. (1985b). *Information development guideline:*

information quality measurement. Poughkeepsie, NY: IBM Corporation.

Klare, G. R. (1963). *The measurement of readability*. Ames, IA: Iowa State University Press.

Klare, G. R. (1988). The formative years. In Zakaluk and Samuels (Eds.), *Readability: Its past, present, and future* (pp. 14-34). Newark, DE: International Reading Association.

Kohl, J. R. (1999). Improving translatability and readability with semantic cues. *Technical Communication*, *46* (2), 149-166.

Merritt, L. (1997). Is quality worth it? *Intercom*, *45* (5), 39-41.

Powell, K. B. (1981). Readability formulas: Used or abused? *IEEE Transactions on Professional Communication*, *PC-24* (1), 43-45.

Smart, K. L. (1999). Establishing dimensions of quality for technical information. *Intercom*, *46* (5), 40-41.

Thatcher, B. L. (1999). Cultural and rhetorical adaptations for South American audiences. *Technical Communication*, *46* (2), 177-195.

Trompenaars, A. (1998). *Riding the waves of culture: Understanding cultural diversity in business*. London, UK: The Economist Books.

Watson, T. J., Jr. (1970). Management briefing 2-70. In *Thirty years of management briefings: 1958 to 1988*. Armonk, NY: IBM Corporation.

Wilde, E., and Colvin, R. (1996). Editing for quality: How we're doing it. *Proceedings of the 1996 IEEE International Professional Communication Conference, IPCC* (pp. 296-303). Piscataway, NJ: IEEE Professional Communication Society.