edmond weiss consulting

115 Cooper Rd., Voorhees, NJ 08043
856-753-3440/edweiss@aol.com/www.edmondweiss.com

How to Write an Unclear Instruction

Edmond H. Weiss, Ph.D.

Although published over 25 years ago, the problems raised in this article are still as relevant as ever. Unclear instructions—for software, programmable phones and consumer products, government forms—are a continuing threat to productivity.

DATA TRAINING
The Monthly Newspaper For Information Trainers

Ten Ways to Write an Unclear Instruction

DOCUMENTATION

Edmond H. Weiss

For a system to be usable—"friendly"—it must have readable user documentation. For a users' manual to be readable, it must have been reviewed by an editor who knows the difference between clear and unclear writing, between a readable and an unreadable sentence. (Note: Among the sentences that need the most editing are the prompts and messages that appear on screens!)

Unfortunately, though, many of the firms that produce user documentation neglect editing. First, they plan and produce user publications so inefficently that they leave barely enough time for a careful editorial review. Second, many have no one on staff who knows enough about clear, readable English to do the editing. (And not all "technical writers" can do it; I've met many professional documentors who could not even spell supersede.)

Editing is never a matter of aesthetics alone. It is assuredly not a frill! Unedited passages are just plain harder to use. And the worse they are, the fewer the people who can read them, and therefore, the higher the number of breakdowns, bad runs, and complaints. Unreadable (that is, unedited) documentation leads to the need for extra personal attention: supervision, training, hand-holding, overtime mop-ups, midnight troubleshooting. All these cost immensely more than the necessary editing would!

And the problem is even more serious when the intended readers—the clerks, operators, technicians, and even programmers—have limited reading skills.

Alas, there are scores of ways to write badly, to make an otherwise clear, readable instruction into an obscure, unreadable mess. And it is particularly in the sentences that give instructions that the consequences of bad editing are worst: frustration, expensive errors, sometimes complete shutdowns of service. Let me show you ten of the most common, most dangerous, and most easily corrected "bugs."

1. Long Words and Vogue Words

Unnecessarily long words (like utilize, prioritize, and facilitate, instead of use, rank, and help, respectively) increase the reading difficulty. Although no writer should replace a long correct word with a short incorrect word, many of the long words in first drafts can be replaced easily, with no loss of meaning at all.

Vogue words ("buzzwords") are terms that everyone uses—like environment and capability. The trouble with such words is that, the more often they are used, the less precise their meanings. For example, is *the environment* the hardware (as in "an HP 3000 *environment*"), or everything but the hardware (as in "HP 3000 would be best in this *environment*")? Is a *capability* something a system can do, or something it will be able to do under the right circumstances?

No: In the Information Center *environment*, the manager should *utilize* a *prioritization* ranking to *facilitate equitable* scheduling.

Yes: In the Information Center, the manager should rank all jobs to yield a fair schedule.

No: Be sure to *initiate* an *interface capability* between the user departments and Systems & Procedures.

Yes: Be sure the user departments and Systems & Procedures can communicate frequently.

2. Wordiness

Wordiness is the effect of many causes, the most common of which are the substitution of long-winded phrases for simple words (for example, *by means of the utilization* for *with*)

and the smothering of verbs inside of nouns (for example, *perform a calculation of* instead of *calculate*).

This problem can often be blamed on the English teacher's habit of assigning 100-word themes, a practice that quickly teaches bright students to write phrases like *should it prove to be the case that* instead of *if*, or *have a lack of* instead of *lack*.

No: *In the event that* you *have a lack of knowledge regarding* which files you *have permission* to write in, make use of the PRIFIL command.
Yes: If you do not know which files you may write in, use the PRIFIL command.

No: *By means of utilizing* the TRND model, the analyst can *do projections of the sales forecasts* and *make decisions regarding* the best sales targets.
Yes: Using the TRND model, the analyst can forecast sales and set the best sales targets.

3. Compression (Too Few Words)

Still other writers go to the opposite extreme. Impatient with writing, annoyed that they are asked to explain things, they compress and compact language to the point of incomprehensibility. They drop articles and prepositions, and string nouns and modifiers into chains of obscurity.

No: *Column heading revision permission* may be obtained by *HCOL entry*.
Yes: To get permission to change the headings of the columns, type HCOL.

No: *Manual data verification and correction* prior to *automatic editing routine utilization* yields *occasional operational efficiency*.
Yes: Sometimes you will save money by verifying and correcting the data manually before using the automatic editing routine.

4. Misplaced Words and Phrases

According to the rules of English grammar, modifying words and phrases must be close to the terms they modify. The greater the distance between them, the greater the chances of an ambiguous or unreadable instruction.

No: *Only* write corrections, not changes on the worksheet.
Yes: On the worksheet, write only corrections, not changes. NOTE: *only* is the slipperiest modifier.

No: You will *almost* save 30% by updating the daily transactions of the worksheet *before entering*.
Yes: You will save almost 30% if, before entering, you update the worksheet to show today's transactions.

5. Backwards Conditionals

It is a fundamental, but little appreciated, principle of effective writing that the main words in a sentence should generally appear at the end. (If you don't believe me, read Strunk & White's *The Elements of Style.)* Nowhere is the importance of this principle more evident than in the backwards conditional, an "if-then" instruction written in the form of "then-if."

Such backwards instructions tend to be difficult and confusing; they demand at least two readings. They are operator-errors waiting to occur.

Backwards conditionals are easy to spot when the if-clause actually starts with the word *if*. Unfortunately, though, many if-clauses start with *to, when, whenever,* or even *should it prove to be the case that*.

No: Press the Clear Rest key *if* you want to erase everything after the cursor.
Yes: If you want to erase everything after the cursor, press the Clear Rest key.

No: Use CNAMES to review the file names already in use *whenever* you want to name a new file.
Yes: Whenever you want to name a new file, use CNAMES to review the file names already in use.

6. Dangling Introductions

Once documentors get the knack of putting the conditional clause (if-clause) first, they must then be wary of a common error of grammar: the dangling introductory phrase. (Even people who know no grammar at all have almost surely heard of the "dangling participle.")

An introductory phrase must be tied grammatically to the subject of the sentence, which must follow that phrase immediately.

No: When reconciling the account, the encumbrance file must be frozen. (This clumsy sentence says that the encumbrance file is reconciling the account!)
Yes: When reconciling the account, the auditor must freeze the encumbrance file.

No: To coldstart the system, the tape containing the operating software is loaded. (Again, this sentence says the tape is starting the system!)
Yes: To coldstart the system, you must load the tape.

7. Empty Predicates

In first drafts, many authors blurt out everything they want to write before they even reach the predicate of the sentence. What should have been the verb in the sentence is hiding somewhere in the subject. The resulting utterance is clumsy, wordy, and awkward. Even though such sentences are grammatically correct—even though they have predicates—their predicates tend to be empty, functionless, even silly. Watch out for such predicates as *exists, is mentioned, is worthy of note, is to be observed...*

No: A need for developing a much more ambitious orientation program *exists*.
Yes: We need to develop a much more ambitious orientation program.

No: The efficiency of spot-checking the coding sheets before commencing keypunching *is worthy of note*.
Yes: It is efficient to spot-check the coding sheets before you start to keypunch.

8. Tangled Passives

The passive voice of the verb, in which the subject is "acted upon," is the bane of technical writers and editors. Take any course on clear

writing, and the instructor will be sure to rail against the passive (usually to a group of engineers or programmers not quite sure what the difference between "passive voice" and "active voice" is).

Actually, there is nothing inherently wrong with the passive voice of the verb; sometimes the passive can make a sentence shorter and clearer. Usually, though, the passive will turn a simple instruction into a tangled mess two or three times harder to understand than it should be. Be especially wary of sentences with the verbs *engaged in, evidenced by, exhibited by, exercised in, demonstrated by.* These and similar verbs are often a sign of an unclear instruction.

No: Care must be *exercised in* labeling prorietary data.
Yes: Label proprietary data carefully.

No: Only the simplest equipment maintenance should be *engaged in* by the operators.
Yes: Operators should perform only the simplest equipment maintenance.

9. The Third Person

Some instructions are written in the third person. That, is the subject is *he, she, it, they,* or the name of the person doing the task. Other instructions are in the second person. That is, the subject is *you,* expressed or implied.

There is no easier way to improve the clarity and readability of an entire instruction book or operators' guide than to switch from the third to the second person. Of course, there are times when the second person will not work, or when a tyrannical editor or supervisor insists on the third. Even so, unless you have a powerful reason for putting your instructions in the third person, write them in the second. Not only will your instructions become lighter, crisper, shorter, and easier to understand, but you will never have to write *his or her* again.

No: The *operator* then enters *his or her* security status code.
Yes: Enter your security status code.

No: The *user* wishing a deeper view of the data base must get permission from the data base manager.
Yes: To see a deeper view of the data base, (you must) get permission from the data base manager.

10. Cowardly Commands

Every day, programmers and documentors write hundreds of direct instructions and commands that scarcely sound like directives at all. Instead, these writers suggest, they discuss, they proffer, they ruminate, they conclude...They do everything but what they should: give direct orders.

Be wary of instructions containing such words as *requirement, required, necessary, necessity, mandate, mandatory, responsibility, responsible,* and a host of others.

No: It is a *requirement* that operators be aware of the room temperature at all times.
Yes: Operators must know the room temperature at all times.

No: It is the responsibility of the project manager to schedule the functional review.
Yes: The project manager must schedule the functional review.

Remember: The Problem of Editing

The ten "bugs" I have described here account for many of the difficult, unreadable instructions now in print. (There are, of course, many more "bugs.") The point to be stressed, though, is that these problems are "debugged" by editing. No one, not even a professional technical writer, can write a first draft that is free of these flaws.

The rational, efficient solution is to edit—to accept the inescapable fact that every first draft is filled with bugs. Even under the very best conditions, first drafts are hard to read, and first drafts of manuals are unreliable and hard to use. If these drafts are not edited by a competent editor, then, inevitably, the documents will be ineffective.

Clear, clean writing—no matter what programmers think—is not a matter of pedantic aesthetics: it is a matter of efficiency. It costs more, far more, to correct the consequences and clean up the messes caused by unreadable manuals than it would to have edited them in the first place. (Skipping editing to save time and money is like skipping testing—reckless and expensive.)

And those organizations that have no employees with editing skills are at an important disadvantage.

Most programmers, by their own account, have little aptitude and less patience for writing and editing. And those "technical writers" hired because they know electronics or COBOL (rather than English), are often not much more help either.

Again, the issue is not aesthetics; the goal is not high style or "literature." The true issue is "usability." Even though a good manual cannot compensate for the defects in a badly-designed system, an otherwise good system can be ruined (or rendered inefficient) by unreadable instructions.

Edmond H. Weiss is an independent consultant, author, and lecturer based in Cherry Hill, NJ. His seminars on communication and documentation have been attended by thousands of managers and DP professionals, in four countries. Weiss is the author of The Writing System for Engineers and Scientists *(Prentice-Hall, 1982). His articles on documentation have appeared in* ComputerWorld, InfoSystems, *and* Creative Computing.

Data Training
38 Chauncy Street
Boston, MA 02110
(617)542-0146

Seminars, Courses & Speeches from Edmond H. Weiss

(Each course can be tailored to the client's interests and time requirements.)

Business/Professional Communication	How to Sell in Writing (Proposals & Business Cases)
	How to Write *Globally*
	Final Draft: The *Especially* Clear Sentence
	The Art of the Pitch
	The Art of the *Engineered* Memo
	The Art of Effective E-Mail

IT/Technical Communication	A Writing System for IT/Technical Professionals
	Preparing English Tech Documents for International Readers
	Effective Quality Manuals/ Usable Procedure & User's Manuals

Organizational Communication	Turning Words into Money: Business Plans & Cases
	Meetings that Work
	There's Only Now: Managing the Professional's Time
	Raising Culture Consciousness

Books by Edmond H Weiss

These and other works are available at Amazon and other book sources.

The Elements Of International English Style: A Guide To Writing Correspondence, Reports, Technical Documents, And Internet Pages For A Global Audience by Edmond H. Weiss (**Paperback** - Feb 28, 2005)

Buy new: **$21.95** 21 Used & new from $14.27

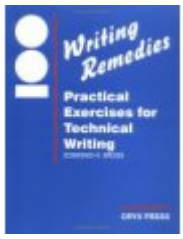
★★★★★ (4)

How To Write Usable User Documentation: Second Edition by Edmond H. Weiss (**Paperback** - Jun 26, 1991)

Buy new: ~~$32.95~~ **$19.67** 28 Used & new from $6.98

★★★★★ (5)

Books: See all 45 items



100 Writing Remedies: Practical Exercises for Technical Writing by Edmond H. Weiss (**Paperback** - Sep 4, 1990)

Buy new: **$25.00** 15 Used & new from $2.50