

The Need for a Writing *Style* Checker

Gerald Cohen

**We have spell checkers. We have grammar checkers.
What we really need today is a *style* checker.**

Most word processing programs come with a spell checker, a useful tool indeed. Most also come with a grammar checker. What we *really* need today is a *style* checker. Why?

Buyers of hardware and software products are *not* complaining about bad spelling or bad grammar. They are complaining that they cannot understand what they have to read in order to use the products they buy.

Actual example:

“This manual is impossible to follow. So far we have met with two of your system engineers who have been having difficulty decoding your book. We have also spoken to another customer who has been trying to put the system up to no avail. At this time, our people are trying to locate someone in your company who can explain in clear, simple terms what we must do.”

Now, was there anything in that complaint about bad spelling or bad grammar? So *something else* must be the problem. That something else is a writing style that makes the writing incomprehensible. It's bad style.

The answer is good style. Good style in technical writing means that the

text has one meaning understood at the first reading. Good style promotes clarity and good customer relations. Bad style promotes confusion and frustration. It multiplies the workload on the help-line support people.

A customer once told me that “a good manual can even make up for shortcomings in the product itself.”

Bad style guarantees that there will be more articles in the trade press and the public press blasting the quality of today's technical documentation. (Even Dilbert has touched on the problem.)

What would a writing style checker do? Unlike a grammar checker, the style checker addresses questions of readability and understandability. It probes the very meaning of *meaning*.

There is no secret about what constitutes good style in writing. The principles were spelled out decades ago by the great teachers of writing. Here, for example, are ten principles proposed by Robert Gunning:

1. Keep sentences short.
2. Prefer the simple to the complex.
3. Prefer the familiar word.
4. Avoid unnecessary words.

5. Put action in your verbs
6. Write like you talk.
7. Use terms your reader can picture.
8. Tie in with your reader's experience
9. Make full use of variety.
10. Write to express not impress.

So, we *know* what the principles of good style are. The difficulty is in actually *applying* those principles to our work.

Just as we have a spell checker that catches our misspellings, we need a style checker that catches our lapses into bad style.

Obviously, a computer program designed to promote the ten principles listed above would be immense and complex. A solution must never be more complex than the problem.

My style checker is programmed to focus on only three defects that I believe are the real *root* causes of bad style. If we can cut just those roots, the writer will almost be compelled to follow the ten principles of good style listed above.

My program reads a document, checking for the presence of these three stylistic problems:

1. An overuse of acronyms (including abbreviations);
2. An overuse of abstract nouns;
3. An overuse of passive verbs (and you thought that this monster was extinct).

Notice that in each case, I say "overuse." There is nothing wrong with acronyms, or abstract nouns, or pas-

sives, as such. There is something terribly wrong with their overuse.

The style checker highlights those problems and assigns a rating to a document as follows:

Computer rating

Very easy [to understand]
 Easy
 Fairly easy
 Acceptable
 Fairly difficult
 Difficult
 Very difficult
 Fogbound

If you meet with a technical person—a programmer or an engineer—and imply that their writing is hard to understand, they will scoff: "That's *your* opinion." But if you show them a printout, where the *computer* says their writing is very difficult or even fogbound, then you will get an entirely different reaction.

The next step is to add a section to the computer program that will automatically convert a sentence like—

"The capability of performing the report printing function is provided by the 5400 printing device." (15 words)

to—

"The 5400 prints reports." (4 words)

Can you imagine what such a program would do to that 300-page manual (or proposal) you were struggling with yesterday?



**He sat there, staring into the tube,
his ego deflated. The computer
displayed its rating of his writing:
FOGBOUND.**