



## How to Write a Good Technical Article

Steve McConnell

*IEEE Software* receives about 200 manuscripts each year, of which we publish approximately 50 to 75. Each manuscript goes through an in-depth peer review process and is reviewed by our associate editors in chief and me. In addition, guest editors review special-issue manuscripts.

Four years as editor in chief of *IEEE Software* have shown me countless examples of the differences between good and not-so-good technical articles. You might have wondered what you need to get a technical article published, either in *IEEE Software* or elsewhere. Here are some pointers.

### Content

One frequent misperception about *IEEE Software* is that it is a “journal.” In common parlance, *IEEE Software* probably is a journal in the sense that it publishes substantive papers of interest to leading practitioners. In IEEE parlance, however, *IEEE Software* is a magazine rather than a transaction or journal. The difference between these terms in IEEE-speak is that a magazine publishes articles and columns of contemporary interest to practitioners. Transactions focus more on publishing research results. There is no strong expectation that a transaction’s readers will read its papers as soon as they are published; rather, we expect readers to archive a transaction for future research purposes. We expect magazine readers to read a magazine’s contents close to when we publish them.

As a magazine, *IEEE Software* has more latitude than a transaction about the kinds of articles it

than a transaction about the kinds of articles it publishes. We can publish reports of a single project or company's experiences, and the findings need not rise to the level of statistically significant research if they provide valuable insights to our readers. Articles can also describe experiences with a new tool or practice, new ways of using old practices, new combinations of old practices, and so on. However, experience backed by data has a better chance of being published than purely anecdotal experience reports. Of course, we are also always looking for research findings that do rise to the statistically significant level, as long as those findings are of interest to leading software practitioners.

An important factor in my decision about whether to accept an article is how clearly the article is focused. A good article addresses exactly one topic. I have been surprised at how many submissions cover one-and-three-quarters topics poorly rather than cover one topic well. The solution is often simply to remove extraneous material. As Voltaire pointed out, an article is finished not when there is nothing more to add, but when there is nothing more that can be taken away. Similarly, a good article has a clear purpose. If I can't determine the point the author is trying to make, I won't accept the paper.

Our referees and readers have told us they dislike articles that evangelize a specific tool or methodology, especially if one specific company sells that tool or methodology. Candid experience reports are always welcome; dressed-up marketing pieces rarely make it past my desk into the review process. In contrast, our readers like articles about real, hands-on experiences. I give leeway to articles based on hands-on experiences, submitted by working practitioners who might not be well-versed in the nuances of writing for publication. As long as you clearly and honestly report the experience, you don't need to worry about the writing being perfect. *IEEE Software's* excellent staff of editors can turn the report into a publication-quality article as long as the writing is technically accurate.

--- --

## Writing style

The best style for a technical article is to present background information and technical conclusions as clearly as possible. Other objectives are secondary to clarity.

I have received papers in which the author seems to be trying to impress readers with his or her intellect. Some papers use large words when synonymous shorter words would suffice, and others use sentences that are longer and more complicated than needed. Some papers create a profusion of needless acronyms or use mathematical formulas to present concepts that could be presented with one or two short sentences.

*IEEE Software's* audience is software practitioners. Leading practitioners are busy people who place a premium on accessible information. Big words, complicated sentences, and formulas don't impress them. Our readers value articles that quickly and blatantly cut to the heart of the issue at hand. The more practical the article, and the more it is directed toward practicing software developers and managers, the more likely we are to publish it.

The highest praise an *IEEE Software* article can receive is "It seems like common sense." If an author can present a new concept so clearly that readers view it as common sense, the author has accomplished something significant.

## Pitfalls

Papers sometimes fall short in ways that can be easily avoided.

### **Lack of focus in multi-author papers**

Many papers we publish have multiple coauthors. These papers frequently suffer from redundant sections written by different authors, from writing styles that differ grossly from one section to the next, and so on. Coauthor teams should appoint a lead

author or editor who can make a final pass through the paper to check for redundancy and inconsistencies.

### **Overgeneralizing from experience**

While our readers love experience reports, I find that authors sometimes draw conclusions that extend beyond their reported experiences. For example, a paper that reports how much developer morale improved after using certain technical practices should not speculate that productivity will “probably” or “obviously” improve. Similarly, a paper can’t claim that a method improves portability, maintainability, or adaptability until the software concerned is ported, maintained, or adapted. A paper provides significant value by reporting even simple, narrow findings, but speculation beyond what the experience or data supports detracts from a paper’s contributions.

### **Too much academic background information**

Many papers spend pages providing background information on a familiar topic—for example, the Software Engineering Institute’s CMM for software. A college term paper might require such background, but a paper submitted for consumption by practitioners does better to provide a one-paragraph summary of familiar topic areas and direct readers to seminal books or articles on the topic. Readers are more interested in the author’s specific experiences than in reviewing familiar background material.

### **Reluctance to submit a short paper**

When reviewing submitted manuscripts, I sometimes get the impression that the authors had a good, clear idea but felt it was too small by itself to submit for publication. The authors then submit a paper with detailed background information, collateral material, and so on—which has the cumulative effect of obscuring the article’s real contribution. Our readers have been exceedingly clear that shorter is better. A paper should make its point and then stop.

## Submitting a paper that is inappropriate for a theme

Occasionally we receive a submission for a theme issue that is outside the theme's scope. In the worst cases, the author has changed the paper's title to match the theme, but the contents don't match the title or relate to the theme. Submitting such papers is a waste of time for everyone involved. If you doubt whether your paper is appropriate for a theme, email the theme's guest editors. They will help you focus your ideas in ways consistent with their theme, which maximizes the chances that we'll ultimately accept your paper.

## Expect to be edited

It's natural to become attached to writing into which you've poured precious evening and weekend hours. No one will be more familiar with a paper's content than the author. However, *IEEE Software's* readers are accustomed to reading articles that are presented in a particular style. The job of *IEEE Software's* professional editing staff is to ensure that each article is ultimately published in a form that facilitates that connection between the author and our readers. Most authors don't enjoy being edited, but if you keep an open mind, editing will almost always improve the quality of your paper. If nothing else, you'll learn what someone else thinks is needed to connect with our magazine's readers.

## Details

Authors should also address a few details that ease the reviewers' and editors' jobs. Submissions should contain

- Page numbers on each page
- An abstract or executive summary of 150 or fewer words
- A list of keywords
- Contact information for each author on the first

page

In addition, *IEEE Software* does not generally publish articles that have been published elsewhere or that are simultaneously being considered for publication elsewhere.

My bottom-line question for an *IEEE Software* article is, “Does it make a contribution to the software engineering literature?” Some articles contribute by introducing a revolutionary concept or by synthesizing familiar concepts in a new way. Others contribute by providing an exceptionally accessible introduction to a specific topic or by providing an unusually clear and balanced survey of a topic area. A good article says something new or says something old in a new way. If you have ideas about an article you would like to publish, please do not hesitate to contact us at [software@computer.org](mailto:software@computer.org).

Copyright © 2002 Steven C. McConnell. All Rights Reserved.



---

Send general comments and any questions about the IEEE Computer Society's Web site to [help@computer.org](mailto:help@computer.org).  
Read our [Privacy and Security](#) guidelines.

This site and all contents (unless otherwise noted) are [Copyright](#) ©2002, IEEE, Inc. All rights reserved.