

How to write a successful EuSpRIG paper

If you have not written a conference paper before, you may be wondering how to organise your thoughts in a form that is acceptable to EuSpRIG. Here are a collection of suggestions intended to encourage the practitioner; if you have any questions, please email Simon Thorne at the address sthorne@uwic.ac.uk

1. **Say what you have learnt in your practice, why it is better than other ways, and show others how they can do it too.**
2. Start by familiarizing yourself with previous EuSpRIG papers. Search for ‘EuSpRIG’ on Arxiv.org and choose those that address issues that you relate to.
3. The key to a successful practitioner paper is:
 - a. Describe one current practice
 - b. Say what problems are associated with it
 - c. Say how those problems can be detected, if they do nothing else
 - d. Say how those problems can be prevented
 - e. Show how your approach both prevents those problems and (this will really get their attention) is no more costly (or little more) than the current practice.
4. www.cs.colorado.edu/~grudic/teaching/CSCI4202_2005/How%20to%20write%20a%20Paper%20or%20Presentation.pdf
How to write a Journal/Conference Paper (Greg Grudic’s sure fire recipe)
 - a. Part 1: Specify what problem (topic) is being addressed.
 - b. Part 2: What is the current state of the art of solutions?
 - c. Part 3: What is the proposed solution?
 - d. Part 4: Discussion: How does the proposed solution compare with others? In theory? In practice? What are the weaknesses?
 - e. Part 5: Conclusion. Summarise. Identify open questions.
5. From Bill Stewart (Slashdot, May 7, 2006), edited :
 - a. Write like a newspaper reporter, not a grad student.
 - b. Your objective is clear communication to the reader, not beauty or eruditeness or narration of your discoveries and reasoning process. Don't waste their time, or at least don't waste it up front.

Tips for writing conference papers

- c. Hit the important conclusions in the first few sentences so your reader will read them. If you'd like to wrap up with them at the end of your memo, that's fine too, in case anybody's still reading by then, but conclusions come first.
 - d. If you're trying to express something complex, simplify your writing so it doesn't get in the way. For something simple, 10th grade language structures will do, but if it's really hairy stuff, back down to 8th grade or so.
 - e. Think about what your audience knows and doesn't know, and what they want and don't want. Express things in terms of what they know and want, not what you know.
6. Read Strunk and White, *Elements of Style*. Again. Or *Modern English Usage* by Fowler.
7. www.usenix.org/events/samples/submit/advice.html
These thirty-odd questions can help you write a better technical paper. Consult them often as you organize your presentation, write your first draft, and refine your manuscript into its final form. Some of these questions address specific problems in "systems" papers; others apply to technical papers in general. Writing a good paper is hard work, but you will be rewarded by a broader distribution and greater understanding of your ideas within the community of journal and proceedings readers.
8. www.jobs.ac.uk/careers/tips/556/Top_Ten_Tips_for_giving_a_conference_paper
Top Ten Tips for giving a conference paper
9. <http://www.chi2005.org/cfp/guide.html>
Guide to Successful Submissions (a serious read)
10. <http://research.microsoft.com/en-us/um/people/simonpj/papers/giving-a-talk/giving-a-talk.htm>
“Writing papers, giving research talks, and writing research proposals are key skills, but they aren't easy. This page describes how I approach each of these three challenges, in the hope that they may be useful to you. “
11. Don't get hung up about academic standards and definitions. They are not necessary for respectability. Don't get into a fight about terminology - if you use computer science terms you are putting yourself into the crosshair sights of those who use these terms in very carefully defined ways; the same applies to software engineering terms like ‘Agile’.