Readers, Writers, Reviewers, and Editors. Four important constituencies that make the *IEEE Transactions on Software Engineering* (TSE) tick. My first two months as TSE Editor-in-Chief have been a rewarding learning experience, and I have had many interesting conversations with authors, reviewers, and associate editors. But alas, nobody has written to me identifying themselves as readers! I realize that many authors, referees, and editors do also read TSE, but are there no readers other than those?

Dave Parnas, TSE’s Emeritus Associate Editor, often reminds me that TSE should serve its readers above all others. I agree with him, although I also realize of course that it is authors and reviewers who keep the TSE ecosystem running. However, even as authors, reviewers, and editors complain that bibliometrics dominate our research assessments, very few of us also complain that our readers are not assessing what we write. Why is this so?

If you are a reader, are you satisfied with what you are getting in each issue of TSE? Are you reading at least some of the papers? If so, what papers in 2009 did you like most and why? If we are to reduce our reliance on citation numbers and impact factors, perhaps we should look for other forms of assessing significance and influence? What better than getting the qualitative feedback from readers—researchers and practitioners—who have benefitted from reading papers published in TSE?

The above are not rhetorical questions. As I mentioned in my editorial to the previous issue, I would like to understand the incentives of readers, writers, reviewers, and editors, and reward each accordingly, to recognize excellence and improve performance. Please do write to me to tell me what you like and dislike about TSE and its published papers. You can email me of course (tse-eic@computer.org), but why not try posting a comment on to the TSE Forum (http://www.computer.org/portal/web/tse/forum)?

**Special Section on Exception Handling**

This issue contains a special section on exception handling, 11 years after TSE first published a special issue on this topic. Exceptions are often the cause of serious failures in software systems, and software engineering research has delivered many innovations that help deal with exceptions. I’d like to thank guest editors Alessandro Garcia, Alexander Romanovsky, and Valérie Issarny for putting together this section of four fine papers, which they introduce in their own guest editorial that follows.

**Welcome**

Last but not least, I would like to welcome Professor Bev Littlewood to the TSE Editorial Board. Bev has a distinguished research record in software engineering, and has also served TSE with distinction as a TSE Associate Editor in the past. I am delighted that despite being “semi-retired” (his words, not mine), he continues to offer his professional services so generously to the community.

Bashar Nuseibeh

Editor-in-Chief
Bev Littlewood has degrees in mathematics and statistics, and a PhD in statistics and computer science; he is a Chartered Engineer, and a Chartered Statistician. He has worked for more than 30 years on problems associated with the dependability of software-based systems, and has published many papers in international journals and conference proceedings and has edited several books. His technical contributions have largely focused on the application of probabilistic and statistical techniques in software systems engineering. He has a long term interest in the question of how far these approaches can take us in providing assurance that systems are fit for purpose, in particular whether certain critical systems are sufficiently safe to be allowed to operate. He has encouraged and taken part in debate about such issues in both technical and non-technical forums. In 1983, he founded the Centre for Software Reliability (CSR) at City University, London, and was its director from then until his semi-retirement in 2003. During this period CSR attracted many millions of pounds of research funding from various European and UK national agencies, and gained an international reputation for the quality of its research. He is currently a professor of software engineering at City University. From 1990 to 2005, he was a member of the UK Nuclear Safety Advisory Committee, in which role he played a part in the extensive discussions, and controversy, concerning the first use of a software-based protection system for a UK power reactor. He is a member of IFIP Working Group 10.4 on Reliable Computing and Fault Tolerance, of the UK Computing Research Committee, and is a fellow of the Royal Statistical Society. From 2007 until 2009, he was a member of the IEEE John von Neumann Award Committee. He is on the editorial boards of several international journals. In 2007, he was the recipient of the IEEE Computer Society’s Harlan D Mills Award.