

Golden Jubilee Year of the Software Journal: Celebrating its Vision, Ground-breaking Contributions and Impact

I am delighted that our favorite Journal of Software: Practice and Experience (SPE) turns 50 and the year 2021 is its Golden Jubilee year! The founding fathers of SPE Journal, Charles Lang and David Barron, in their editorial for the first issue published in January 1971, noted “the continuing development of the application of computers in all walks of life is held back by the difficulty of producing software”. They also wrote “As applications become more sophisticated and systems get larger, the effects of bad software will be less easy to ignore since the possibility of disastrous errors will increase.”

During the last 50 years there are many instances of software errors causing failures of big missions, critical business, and medical services. A Wikipedia list of software bugs (https://en.wikipedia.org/wiki/List_of_software_bugs) shows their serious consequences—either financially or as a threat to human well-being from many application domains including medical, space, telecom, finance, military, transportation, and government administration. Just to share a recent example: A day-long outage in trading at the Australian Securities Exchange (ASX) on Monday November 16, 2020 was traced to “software bug” in their new equity trading platform.

Lang and Barron identified seven major factors that have led to this situation, which continue to hold true even after 50 years. I would like to state the last two verbatim:

- “Lack of adaptability of software, which makes it impossible to include facilities not anticipated at the start, or to adapt someone else’s program to fit into our system.”
- “Lack of communication, which leads to repetition of work that has been done before and means that, as I-ramming has pointed out, whereas scientists make progress by standing on each other’s shoulders, software people trample on each other’s feet.”

Since its foundation, SPE primarily aimed at addressing these two problems by serving as a vehicle for dissemination of practice and experience of software researchers, developers, engineers, and practitioners to write up and present their work in a form that will be useful to others. Although software practice has greatly improved from then with the help of these continued efforts from SPE and community at large, many challenges yet to be addressed.

This unique aim of SPE has drawn me to it and publish my work 25 years ago when I was working on a Microkernel based Operating System for India’s PARAM Supercomputers at the Centre for Development of Advanced Computing (C-DAC). When I was invited to join SPE team of Editors in 2014, I was delighted and continue to consider it as my privilege to be part of this visionary and pioneering journal, whose founding year happens to be the same year in which I was born!

SPE FEATURES EARLY GROUND BREAKING WORKS

Right from its first issue, SPE has attracted articles that covered early breakthrough works from prominent researchers and companies on their work on operating systems software, programming languages, data structures, and applications. Let me share four ground breaking works published in the first volume of SPE in 1971:

- The first article in the first issue (Volume 1, Issue 1, Jan 1971) by C. R. Spooner from Control Data Corporation (Washington, D.C.) discussed software architecture in the context of a multi-programming operating system.
- The first article in the second issue by Donald E. Knuth (1974 recipient of the ACM Turing Award) presented empirical investigation of FORTRAN programs and offered guidelines for compiler developers.

- An article in the third issue by B. Landy from the University of Cambridge discussed their scheduling strategies in the TITAN operating system.
- The first article in the fourth issue by Niklaus Wirth (1984 recipient of the ACM Turing Award) presented the design of a Pascal compiler.

I am pleased to learn that some of the authors who published their early pioneering work in SPE went on to win ACM Turing Award, which is recognized as the highest distinction in Computing Science and Engineering, or the “Nobel Prize of Computing”.

During the last 50 years, SPE has attracted and disseminated many innovative software systems and applications revolutionizing the software discipline. I look forward for its significant contribution to software industry and society during the next 50 years.

ACKNOWLEDGMENTS

This glories and revolutionary contributions and impact of SPE is made possible by the vision and dedicated efforts of editors, guest editors, authors, reviewers, advisory board members, and journal staff members. It is my humble duty to thank you all for your contributions.

I glanced through the past issues of SPE and identified editors, years of their service, and their affiliation at the start of their service, which appears as follows:

1. Charles Lang (1971–1984), Cambridge University, UK
2. David Barron (1971–1983), University of Southampton, UK
3. David Hanson (1981–1987), Princeton University, USA
4. John Campbell (1984–1994), University College London, UK
5. Douglas E. Comer (1988–2006), Purdue University, USA
6. Andrew Wellings (1994–2014), University of York, UK
7. Nigel Horspool (2006–2017), University of Victoria, Canada
8. Rajkumar Buyya (2014–), University of Melbourne, Australia
9. Kendra Cooper (2016–2020), University of Texas at Dallas, USA
10. Richard Jones (2016–2019), University of Kent, UK
11. Agostino Poggi (2016–), University of Parma, Italy
12. Satish Srirama (2016–), University of Tartu, Estonia
13. Judith Bishop (2018–2020), Microsoft, USA/Stellenbosch University, South Africa

SPE has inducted the following two colleagues as its editors starting from 2021, I warmly welcome them:

- Daniel Lemire, Université du Québec, Canada
- Rami Bahsoon, The University of Birmingham, UK

I request your continued support in sustaining SPE as the premier forum for publishing new software research and innovation along with practice and experience focus. Your feedback and comments on further improving SPE are highly appreciated.

Rajkumar Buyya

Cloud Computing and Distributed Systems (CLOUDS) Lab, School of Computing and Information Systems, The University of Melbourne, Melbourne, Australia

Correspondence

Rajkumar Buyya, Cloud Computing and Distributed Systems (CLOUDS) Lab, School of Computing and Information Systems, The University of Melbourne, Melbourne, Australia.
Email: rbuyya@unimelb.edu.au