

## Guest editorial: emerging areas in automated software engineering research

Tevfik Bultan<sup>1</sup> · Andreas Zeller<sup>2</sup>

Received: 2 August 2017 / Accepted: 4 August 2017 / Published online: 31 August 2017  
© Springer Science+Business Media, LLC 2017

As everything becomes programmable, the significance of research on automation of software engineering tasks keeps increasing. This special section on Emerging Areas in Automated Software Engineering Research highlights the developments in automated debugging support and software analytics for incident management.

The article titled “Lightweight Control-Flow Instrumentation and Postmortem Analysis in Support of Debugging” co-authored by Peter Ohmann and Ben Liblit presents automated techniques that assist programmers in debugging by providing information about program activity before failure. Authors show that latent information in postmortem core dumps can be augmented with lightweight, tunable runtime-tracing, resulting in significant slice reductions. Experimental evaluation provided by the authors indicate that presented techniques provide significant debugging support for programmers in realistic scenarios with low overhead.

The article titled “Experience Report on Applying Software Analytics in Incident Management of Online Service” co-authored by Jian-Guang Lou, Qingwei Lin, Rui Ding, Qiang Fu, Dongmei Zhang, and Tao Xie focuses on application of data-driven techniques to large-scale online service incident management. The authors share their experiences in using software analytics to assist engineers in incident management, development of novel data-driven techniques for large-scale online services, and the lessons learned from research development and technology transfer in this area.

---

✉ Tevfik Bultan  
bultan@cs.ucsb.edu

Andreas Zeller  
zeller@cs.uni-saarland.de

<sup>1</sup> University of California, Santa Barbara, CA, USA

<sup>2</sup> Saarland University, Saarbrücken, Germany

Both articles demonstrate the significance of automation in software engineering and identify directions for future research. We hope that you enjoy reading these contributions.