CrossMark

Guest editorial: special multi-issue on selected topics in Automated Software Engineering

Tim Menzies[1] · **Corina Pasareanu[2]**

This special multi-issue of Automated Software Engineering is an attempt by editors to pull together articles from a range of current interesting topics from the field of Automated Software Engineering. We feel the topics and research reported here form an interesting cross section of current work. While these research efforts have published in recent conferences, we have allowed time for the authors to significantly extend and complete their work, so each of these contributions is novel in its own right and has reached a level of maturity appropriate to an archival publication.

This issue is divided into three parts. In the first part, we include diverse papers from the frontier of software engineering research.

1. In "Monitoring Energy Hotspots in Software", Adel Noureddine. Romain Rouvoy and Lionel Seinturier discuss ways to control the energy usage of software. This field of "Green IT" is concerned with the optimization of software solutions with regards to their energy consumption. In this domain, most current solutions concentrate on coarse-grained approaches to monitor the energy consumption of a device or a process. This new paper reports on a fine-grained runtime energy monitoring framework we developed to help developers to diagnose energy hotspots with a better accuracy.

2. In "User-Aware Privacy Control via Extended Static-Information-Flow Analysis", Xusheng Xiao, Nikolai Tillmann, Manuel Fahndrich, Jonathan de Halleux, Michal Moskal and Tao Xie discuss the problem of privacy "leakage" in mobile

✉ Tim Menzies
 tim.menzies@gmail.com

 Corina Pasareanu
 corina.s.pasareanu@nasa.gov

[1] Department of Computer Science, North Carolina State University, Raleigh, NC, USA

[2] Carnegie Mellon University and NASA Ames, Pittsburgh, PA, USA

apps. Existing mobile platforms provide little information on how applications use private user data. This makes it difficult for experts to validate applications and for users to grant applications access to their private data. These authors discuss propose a user-aware-privacy-control approach based on static information flows (which are classified as safe/unsafe). When applied to hundreds of scripts developed for a mobile platform, this approach effectively reduces the need to make access-granting choices to only 10.1 % (54) of all scripts.

3. Our next paper is "Balancing Precision and Performance in Structured Merge" by Olaf Leßenich, Sven Apel, and Christian Lengauer. It addresses an old problem with some startling new results. When combining code from different sources, developers use some kind of unstructured-merge (i.e., line-based) tools. These are fast, but imprecise. This paper reports a new structured-merge approach with auto-tuning that tunes the merging process on-line by switching between unstructured and structured merge (depending on the presence of conflicts). In the case studies explored here, this method ran 92 times faster than purely structured merge (10 times on average).

4. Finally, in "A model-driven framework for guided design space exploration", Abel Hegedus, Akos Horvath, and Daniel Varro discuss methods for design space exploration that aim at searching through various models representing different software design candidates. A full and exhaustive exploration of the design space is infeasible for large models. Hence, in this approach, the traversal is guided by hints, derived by system analysis, to prioritize the next states to traverse (selection criteria) and to avoid searching unpromising states (cut-off criteria). The technique is assessed using a a cloud infrastructure configuration problem.

The next two issues in this series will cover

- Testing & verification
- Reasoning about source code.

The guest editors are indebted to the authors of submitted papers and our small army of reviewers who offered extensive and timely reviews of the papers. Thanks to you all!