Guest Editorial

Andrew Ireland · Willem Visser

Received: 1 February 2010 / Accepted: 3 February 2010 / Published online: 11 February 2010
© Springer Science+Business Media, LLC 2010

This Special Issue is associated with the 23rd IEEE/ACM International Conference on Automated Software Engineering, which was held in September 2008 in central Italy within the beautiful medieval walled city of L'Aquila. The hospitality and dedication of Prof. Paola Inverardi (ASE-08 General Chair) and her colleagues made for a very special event.

Out of the 280 papers submitted to the conference, 34 were accepted for publication as full papers. Three of the best papers from the conference were selected for inclusion in this special issue, thereby giving the authors the opportunity to expand on their already excellent conference papers. The topics are as diverse as the conference itself and cover the areas of efficient runtime monitoring, accurate code completion and improved patching of evolving software.

Robbes and Lanza in their paper entitled “Improving Code Completion with Program Histories” illustrate that the now ubiquitous idea of code completion based on alphabetical ordering, as found in modern development environments, should not be taken at face value and in fact can be improved significantly by considering previous actions by the developer. In addition to new code completion algorithms the authors also present a benchmarking framework to allow one to compare these algorithms.

Andersen and Lawall looks at the interesting problem of determining how to patch software dependent on a fast changing environment, like the Linux kernel, in their paper “Generic Patch Inference”. The internal libraries of Linux change constantly and thus necessitate changes in device drivers depending on them; however many

A. Ireland (✉)

School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh, UK
e-mail: a.ireland@hw.ac.uk

W. Visser

Department of Mathematical Sciences, University of Stellenbosch, Stellenbosch, South Africa

of these changes have common elements and the focus of this paper is to infer one efficient generic change (patch). A formal model of a generic patch is presented and how to derive such a patch from a set of changes.

“Efficient Monitoring of Parametric Context Free Patterns” by Meredith, Jin, Chen and Rosu investigates how to make runtime monitoring more expressive but without incurring unacceptable execution overhead. Monitoring a program during execution for complex behavioural properties has been shown to be an effective error detection technique, but existing techniques are not good at describing properties dependent on complex data structures. The authors show here that runtime monitoring based on parametric context free grammars can solve this problem efficiently. The techniques are implemented in their JavaMOP tool.

In the Spring of 2009 L’Aquila was tragically hit by an earthquake which killed over 300 people, and left much of the city in ruins. Through the spirit and courage of its people, a vision for the regeneration of L’Aquila has grown out of the devastation.