The Early Research Achievements (ERA) Track aims at providing researchers with a forum for discussing novel research ideas in early stages of development. The topics of interest for this track are the same as the main research track, but we aim at creating a stimulating atmosphere where researchers can present and get early feedback on promising work that has not yet been fully evaluated.

This year’s Early Research Achievements track attracted 40 submissions written by authors from 22 countries on 4 continents. After an intense reviewing process in which each paper was reviewed by at least 3 members of the ERA program committee, 15 papers were accepted by the 20 members of the ERA program committee. As expected, the majority of the papers covers traditional CSMR topics such as reengineering and software analysis. However, the ERA track is proud to also provide a platform for CSMR-related topics that have not been discussed before. The following paragraphs summarize the papers of the ERA track and thereby provide a guide for interested readers.

1) Repository mining
Tóth et al. use the version control history to follow the changes of source code elements. Thung et al. study the network structure of social coding in GitHub. Xia et al. present a comparative study of supervised learning algorithms for re-opened bug prediction. Stevens et al. have developed a history querying tool and applied it to detect multi-version refactorings. Mihancea and Marinescu study whether there is a correlation between changes, defects and polymorphism.

2) Software maintenance
Csaba et al. relate clustering measures to software quality. Polychniatis et al. present a technique to detect cross-language dependencies generically. Kochar et al. perform a preliminary study of 50,000 open source projects to see how they test their software. Molitorisz presents a pattern-based approach for refactoring of sequential source code. Sasaki et al. investigate the reordering of program statements to improve source code readability.

3) Software analysis
Gravino et al. present an early investigation into the contribution of class and sequence diagrams in source code on program comprehension tasks. Kazato et al. investigate Incremental feature location and identification in source code. Jezek et al. explain an approach to supply a compiler with static compatibility checks through the analysis of third-party libraries. Ghaith presents a profile-based, load-independent bug detection and analysis mechanism in regression testing of software systems. Scaniello et al. explore how to use the GPU to green an intensive and massive computation system.

The track chairs would like to thank the members of the CSMR ERA 2012 program committee and the additional reviewers for their effort!

Andy Zaidman  
Delft University of Technology  
The Netherlands

Ali Mesbah  
University of British Columbia  
Canada