JOURNAL OF SOFTWARE: EVOLUTION AND PROCESS
J. Softw. Evol. and Proc. 2013; **00**:1–3
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/smr

Editorial: Special Section on ICSM 2011

James R. Cordy[1] and Paolo Tonella[2,*]

[1]*Queen's University, Kingston, Canada*
[2]*Fondazione Bruno Kessler, Trento, Italy*

SUMMARY

The special section contained in this issue features selected papers from the 27^{th} IEEE International Conference on Software Maintenance (ICSM), held in Williamsburg, VA, USA, on September 25-30, 2011. The Program Co-Chairs of the conference, James R. Cordy and Paolo Tonella, acted as guest editors and supervised the selection and reviewing process for this special section. The included papers are representative of three core research areas in software maintenance and evolution: (1) clone detection; (2) traceability recovery; and, (3) feature location. Copyright © 2013 John Wiley & Sons, Ltd.

Received . . .

KEY WORDS: ICSM 2011 special issue; clone detection; traceability; feature location

EDITORIAL

The IEEE International Conference on Software Maintenance (ICSM) is the premiere international venue in software maintenance and evolution, where participants from academia, government, and industry meet and share ideas and experiences for solving critical software maintenance problems. In 2011, the 27th edition of ICSM was held at the Williamsburg Lodge, in the heart of Williamsburg, VA, USA. Williamsburg, once the capital of England's oldest, wealthiest and most populous North American colony, is one of the most important American historical landmarks.

ICSM 2011 was held from September 25 to 30, 2011. The conference was preceded by two co-located events, the 11th IEEE International Working Conference on Source Code Analysis and Manipulation (SCAM) and the International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA). It was followed by two other co-located events, the 13th IEEE International Symposium on Web Systems Evolution (WSE) and the 6th IEEE International Workshop on Visualizing Software for Understanding and Analysis (VISSOFT).

ICSM 2011 attracted 127 submissions, each of which was reviewed by at least three members of the ICSM Program Committee. Following a week and a half of online discussion supported by EasyChair, 36 papers were selected for publication and presentation in the Technical Program of the conference. Consensus was aimed for and reached in almost all cases. The topics of accepted papers ranged from Reverse Engineering and Program Comprehension to Impact Analysis, Traceability, Migration and Evolution, Refactoring, Software Clones, Linguistic Analysis and Regression Testing. The conference program included the ICSM Doctoral Symposium, Tutorials, Tool Demonstrations, Posters, the Industry track and the Early Research Achievements track.

This special section of the Journal of Software: Evolution and Process contains extended versions of three of the papers selected by the ICSM 2011 Program Committee as among the best papers

*Correspondence to: Fondazione Bruno Kessler, via Sommarive 18, 38123 Povo (Trento), Italy. E-mail: tonella@fbk.eu

presented at the conference. These extended versions have additionally undergone the rigorous JSEP journal review process: they were reviewed by three anonymous referees under the supervision and coordination of the guest editors. We are proud to present you the three excellent papers that are the results of this process.

Paper [1], "An Empirical Study of Faults in Late Propagation Clone Genealogies", by Liliane Barbour, Foutse Khomh and Ying Zou, investigates the history of clone evolution over time, focusing on the phenomenon of late propagation. Late propagation occurs whenever one of two clones diverges from the other, due to code evolution, while later such changes are reconciled, so as to remove the divergence. The authors have defined eight types of late propagation and they have evaluated empirically the relationship between fault-proneness and late propagation of various type. They have also investigated whether reconciling changes are fault fixing changes. This paper contributes to our knowledge of the risks associated with clones and with their evolution over time.

Paper [2], "Evaluating Test-to-Code Traceability Recovery Methods through Controlled Experiments", by Abdallah Qusef, Gabriele Bavota, Rocco Oliveto, Andrea De Lucia and David Binkley, presents a combination of two controlled experiments comparing the effectiveness of four different approaches to recovering traceability links between test cases and code. Traceability addresses the problem of identifying links between software artifacts of different types, and test-to-code traceability addresses the issue of identifying precisely which sections of code are actually tested by a given test case. In the experiments presented in this paper, the effectiveness of the four approaches on three large Java systems is compared using a manual consensus of three PhD student programmers as a reference answer. Both accuracy and practical usefulness to programmers is explored, demonstrating that the authors' own SCOTCH method can be significantly more effective than existing techniques.

Paper [3], "How Developers Perform Feature Location Tasks: A Human-Centric and Process-Oriented Exploratory Study", by Jinshui Wang, Xin Peng, Zhenchang Xing and Wenyun Zhao, describes an empirical study of feature location in code by software developers. Feature location addresses the problem of identifying which parts of the source code of a system are concerned with implementing a specific business or technical concern. Automation in feature location has been a popular topic in the software maintenance research community for some time. In this paper, the authors choose instead to seek insight from how programmers actually perform the task in practice, concluding that the process can be understood at three levels of granularity, each of which is influenced by a number of external factors. The empirical insights reported here can be used to inform future work in supporting and automating this important aspect of software maintenance.

We wish to thank the authors for having contributed to this special section, the ICSM 2011 Program Committee for their indications on which papers to include in the special section, and the journal referees, for their detailed and constructive comments, which greatly helped the authors to improve their papers.

REFERENCES

1. Liliane Barbour, Foutse Khomh, and Ying Zou. An empirical study of faults in late propagation clone genealogies. *Journal of Software: Evolution and Process*, 2013.
2. Abdallah Qusef, Gabriele Bavota, Rocco Oliveto, Andrea De Lucia, and David Binkley. Evaluating test-to-code traceability recovery methods through controlled experiments. *Journal of Software: Evolution and Process*, 2013.
3. Jinshui Wang, Xin Peng, Zhenchang Xing, and Wenyun Zhao. How developers perform feature location tasks: A human-centric and process-oriented exploratory study. *Journal of Software: Evolution and Process*, 2013.

AUTHORS' BIOGRAPHIES

James R. Cordy is Professor and past Director of the School of Computing at Queens University at Kingston, Canada. From 1995 to 2000 he was Vice President and Chief Research Scientist at Legasys Corporation, a software technology company specializing in legacy software system analysis and renovation. Dr. Cordy received his PhD from the University of Toronto, where he served several years as lecturer and senior research associate before moving to Queen's University in 1985. As leader of the TXL project, with hundreds of academic and industrial users worldwide, he is the author of more than 140 refereed contributions in programming languages, software engineering and artificial intelligence and has served widely as member and chair of many conferences and workshops in programming languages and software

DOI: 10.1002/smr

engineering, most recently as Program Chair of ICSM 2011, IWSC 2011 and CASCON 2013, and as General Chair of SCAM 2012 and WCRE 2012. Dr. Cordy is an ACM Distinguished Scientist, a senior member of the IEEE, and an IBM Visiting Scientist and Faculty Fellow.

Paolo Tonella is head of the Software Engineering Research Unit at Fondazione Bruno Kessler (FBK), in Trento, Italy. He received his PhD degree in Software Engineering from the University of Padova in 1999, with the thesis "Code Analysis in Support to Software Maintenance". In 2011 he was awarded the ICSE 2001 MIP (Most Influential Paper) award, for his paper: "Analysis and Testing of Web Applications". He is the author of "Reverse Engineering of Object Oriented Code", Springer, 2005. He participated in several industrial and EU projects on software analysis and testing.

Dr. Tonella was Program Chair of ICSM 2011 and ICPC 2007; General Chair of ISSTA 2010 and ICSM 2012. Among the others, he served in the program committees of ICSE, FSE, ICSM, ISSTA, ICST, ICPC. In 2007, Paolo Tonella was ranked among the top-50 Software Engineering scholars (Communications of the ACM, vol. 50, n. 6, pp. 81-85). He has been recognized as a distinguished TOSEM referee several times in recent years (including 2011-2012). His key contributions to the research in software engineering include: a comprehensive reverse engineering method for object oriented software, pioneering research on model extraction for web testing and foundational work on evolutionary testing of object oriented software.

DOI: 10.1002/smr