call for papers

search based software testing 2014

Held in conjunction with ICSE 2014 - IEEE International Conference on Software Engineering

May 31—June 7, 2014
Hyderabad, India

Keynote Speakers

Lionel Briand

Professor and FNR PEARL Chair, Interdisciplinary Centre for ICT Security, Reliability and Trust (SnT)
University of Luxembourg

Cristian Cadar

Leads the Software Reliability Group in the Department of Computing at Imperial College London

About the workshop:

Search-Based Software Testing (SBST) is the application of optimizing search techniques (for example, Genetic Algorithms) to solve problems in software testing. SBST is used to generate test data, prioritize test cases, minimize test suites, reduce human oracle cost, verify software models, test service-orientated architectures, construct test suites for interaction testing, and validate real-time properties.

The objectives of this workshop are to bring together researchers and industrial practitioners both from SBST and the wider software engineering community to share experience and provide directions for future research, and to encourage the use of search techniques to combine aspects of software testing with other aspects of the software engineering lifecycle.

Program Committee

Wasif Afzal (Mälardalen University, Sweden) **Rob Alexander** (University of York, UK)
Giuliano Antoniol (École Polytechnique de Montréal) **Andrea Arcuri** (Simula Research Labs, Norway)
Mariano Ceccato (FBK (Fondazione Bruno Kessler) Trento, Italy) **Francisco Chicano** (University of Málaga)
John Clark (University of York, UK) **Myra Cohen** (University of Nebraska at Lincoln, USA)
Massimiliano Di Penta (University of Sannio, Italy) **Gordon Fraser** (University of Sheffield, UK)
Vahid Garousi (University of Calgary, Canada & Atilim University, Ankara, Turkey)
Gregory Kapfhammer (Allegheny College, USA) **Yvan Labiche** (Carleton University, Canada)
Simon Poulding (University of York, UK) **Marc Roper** (University of Strathclyde)
Paolo Tonella (Fondazione Bruno Kessler–IRST, Italy) **Tanja Vos** (Universidad Politecnica de Valencia)

Format and Submission:

All papers must conform, at time of submission, to the ACM Formatting Guidelines (LaTeX users, please use the "Option 2" style). All submissions must be in PDF format, and submitted via the website.

Important Dates:

Paper submission: **January 24, 2014**
Notification to authors: **February 24, 2014**

Workshop Chairs:

Phil McMinn
University of Sheffield, UK
(p.mcminn@sheffield.ac.uk)

Mark Harman
University College London, UK
(mark.harman@ucl.ac.uk)

Researchers and practitioners are invited to submit:

* **Full papers** (maximum of 10 pages) to the workshop on original research--either empirical or theoretical--in SBST, practical experience of using SBST, or SBST tools.

* **Short papers** (maximum of 4 pages) that describe novel techniques, ideas and positions that have yet to be fully developed; or are a discussion of the importance of a recently published SBST result by another author in setting a direction for the SBST community, and/or the potential applicability (or not) of the result in an industrial context

* **PhD papers** (maximum of 4 pages). PhD papers are invited for students to showcase their research and to receive feedback from senior members of the SBST community. See the website for eligibility constraints and requirements.

In all cases, papers should address a problem in the software testing/verification/validation domain or combine elements of those domains with other concerns in the software engineering lifecycle. Examples of problems in the software testing/verification/validation domain include (but are not limited to) generating testing data, prioritizing test cases, minimizing test suites, verifying software models, testing service-orientated architectures, constructing test suits for interaction testing, and validating real-time properties.

The solution should apply a metaheuristic search strategy such as (but not limited to) random search, local search (e.g. hill climbing, simulated annealing, and tabu search), evolutionary algorithms (e.g. genetic algorithms, evolution strategies, and genetic programming), ant colony optimization, and particle swarm optimization.

WWW.SEARCHBASEDSOFTWARETESTING.ORG