SPLASH
VANCOUVER
2017
OCTOBER 22–27

REBLS 2017

4TH WORKSHOP ON REACTIVE AND EVENT-BASED LANGUAGES & SYSTEMS
CO-LOCATED WITH SPLASH 2017

DEADLINES

Aug 1st - Submission
Sept 5th – Notification

ORGANIZERS

Guido Salvaneschi
TU Darmstadt, Germany

Wolfgang De Meuter
Vrije Universiteit, Belgium

Patrick Eugster
TU Darmstadt and Purdue University
Germany, USA

Lukasz Ziarek
SUNY Buffalo, USA

WEBSITE

CALL FOR PAPERS

Reactive programming and event-based programming are two closely related programming styles that are becoming ever more important with the advent of advanced HPC technology and the increasing requirement for our applications to run on the web or on collaborating mobile devices. A number of publications on middleware and language design - so-called reactive and event-based languages and systems (REBLS) - have already seen the light, but the field still raises several questions. For example, the interaction with mainstream language concepts is poorly understood, implementation technology is in its infancy and modularity mechanisms are almost totally lacking. Also, large applications are to be developed and patterns and tools for developing reactive applications is an unexplored area.

We welcome all submissions on reactive programming, and event-oriented systems, including but not limited to:

- Language design, implementation, runtime systems, program analysis, software metrics, patterns and benchmarks.
- Study of the paradigm: interaction of reactive and event-based programming with existing language features such as object-oriented programming, mutable state, concurrency.
- Advanced event systems, event quantification, event composition, aspect-oriented programming for reactive applications.
- Functional-reactive programming, self-adjusting computation and incremental computing.
- Applications, case studies that show the efficacy of reactive programming.
- Empirical studies that motivate further research in the field.
- Patterns and best-practices.
- Related fields, such as complex event processing, reactive data structures, view maintenance, constraint-based languages, and their integration with reactive programming. IDEs, Tools.
- Implementation technology, language runtimes, virtual machine support, compilers.
- Modularity and abstraction mechanisms in large systems.
- Formal models for reactive and event-based programming.