

Introduction to the special issue on the 18th international conference on model driven engineering languages and systems (MODELS'15)

Jordi Cabot^{1,2} · Alexander Egyed³

Received: 22 December 2016 / Accepted: 25 December 2016 / Published online: 30 January 2017
© Springer-Verlag Berlin Heidelberg 2017

1 Introduction

MODELS is the premier conference series for model-based software and systems engineering. It has been established in 1998 and has since been covering all aspects of modeling, from languages and methods to tools and applications. *MODELS'15* was the 18th edition of the conference. It took place in Ottawa, Canada from September 27 to October 2, 2015.

MODELS'15 challenged the modeling community to promote the magic of modeling by solidifying and extending the foundations and successful applications of modeling in areas such as business information and embedded systems, but also by exploring the use of modeling for new and emerging systems, paradigms, and challenges including cyber-physical systems, cloud computing, services, social media, big data, security, and open source.

This challenge resulted in 216 abstract submissions that materialized in 172 papers, consisting of 132 technical papers (including 22 new ideas papers) and 40 in-practice papers. Of these, the Program Committee and Program Board accepted 35 foundations papers (26.5% acceptance rate) and 11 in-practice papers (28% acceptance rate). The program also included a diverse group of keynote talks, including presentations on climate models, automotive models, and software supply chains.

Out of the accepted papers, we invited the best ones for this special issue. This invitation was based on a careful

evaluation of all papers by the Program Board and Program Committee. The authors of these best papers were then asked to prepare a substantial improved and extended version for this special issue. Each article underwent a full journal review process and authors received anonymous feedback in two rounds of reviewing from three expert reviewers.

2 Selected papers

Nine papers have been finally accepted in this special issue, six from the foundations track and three more from the applications track. Altogether, they are a good example of the diversity, depth and maturity of the current research in model-driven engineering.

Papers from the Foundations Track:

- “Model development guidelines for UML-RT: conventions, patterns and antipatterns” by Tuhin Das and Juergen Dingel, presents a set of design guidelines for model-based development of complex real-time embedded software systems including nine design conventions, three design patterns and thirteen antipatterns for developing UML-RT models, based on the authors’ analysis of around 100 UML-RT models from industry and academia.
- “A framework for relating syntactic and semantic model differences” by Shahar Maoz and Jan Ringert defines Diffuse, a language-independent, abstract framework, combining syntactic and semantic differencing, the two main families of approaches to detect and understand model differences. The approach relates syntactic change operations and semantic difference witnesses.
- “Quick fixing ATL transformations with speculative analysis” by Jesús Sánchez-Cuadrado, Esther Guerra and

✉ Jordi Cabot
jordi.cabot@inria.fr

¹ ICREA, Barcelona, Spain

² Universitat Oberta de Catalunya, Barcelona, Spain

³ Johannes Kepler University Linz, Linz, Austria

Juan de Lara combines program analysis and constraint solving to identify errors in ATL model transformations and to propose suitable quick fixes for ATL transformation errors. The approach supports speculative analysis to help developers select the most appropriate fix by providing a previsualization of each fix and reporting on its consequences.

- “Full contract verification for ATL using symbolic execution” by Bentley Oakes, Javier Troya, Lucio Levi and Manuel Wimmer presents a static verification method for (ATL) model transformations. Using their DSLTrans language as intermediate representation they are able to use a symbolic execution approach to generate representations of all possible input models to the transformation and use them to verify its contract.
- “Model-based simulation of legal policies: framework, tool support, and validation” by Ghanem Soltana, Nicolas Sannier, Mehrdad Sabetzadeh and Lionel Briand develops a framework for legal policy simulation that helps governments predict how changes in the law affect their revenue or metric of interest. The framework uses models for specifying both legal policies and the probabilistic characteristics of the underlying population.
- “Testing Models and Model Transformations using Classifying Terms” by Frank Hilken, Martin Gogolla, Loli Burgueño, and Antonio Vallecillo aims at guiding the construction of more relevant tests by defining equivalence classes for partitioning the source and target model spaces of the transformation to be tested. Building tests for the different partitions maximizes the coverage of the test suite.

Papers from the In-Practice Track:

- “Multi-purpose, multi-level feature modeling of large-scale industrial software systems” by Daniela Rabiser, Herbert Prähofer, Paul Grünbacher, Michael Petruzelka, Klaus Eder, Florian Angerer, Mario Kromoser and Andreas Grimmer showcases a practical application of feature modeling in two industrial automation companies. The paper demonstrates the value of modeling product variations from different perspectives or levels of abstraction.
- “Configuring use case models in product families” by Ines Hajri, Arda Goknil, Lionel Briand, and Thierry Stephany also worked with industrial product families but investigates them from the perspective of requirements. Their use case-driven approach for product families can be incrementally configured by engineers to yield product specific use cases.
- “Modeling context-aware and intention-aware in-car infotainment systems—Concepts and modeling processes” by Daniel Lüddecke, Christoph Seidl, Jens Schneider, and Ira Schaefer reports about experiences on modeling user intentions to make them more transparent to engineers. They then further demonstrate the value of combining this information with contextual facts.