

## Introduction to the theme issue on models for quality of software architecture

Dorina C. Petriu · Jens Happe

Received: 5 August 2013 / Accepted: 7 August 2013 / Published online: 14 August 2013  
© Springer-Verlag Berlin Heidelberg 2013

High quality of system architecture is essential for the long-term success of a software product. Software engineers need to design and implement secure, maintainable, usable, efficient, and reliable software systems. They have to design architectures that can fulfill quality requirements, judge quality tradeoffs, and ensure that the implementation adheres to the architecture. In case of long-living software systems, they have to monitor, evaluate, and improve quality continuously throughout the entire software lifecycle. Models play a key role in supporting software engineers to master these tasks.

The goal of this special theme issue is to emphasize the deep relationship between system modeling and the quality of system architecture. For example, how can model-driven techniques for system development be used to assess and improve the quality of system architecture? What are the connections between software modeling tools and quality assessment tools and environments? Papers in this theme issue are a step toward bridging the gap between theory and practice of architecture quality and system modeling.

The idea for this theme issue arose during the planning of the Seventh International ACM Sigsoft Conference on the Quality of Software Architectures (QoSA 2011) which was held in June 2011, in Boulder Colorado, USA. As part of CompArch, QoSA was held jointly with the 14th International ACM Sigsoft Symposium on Component Based Software Engineering (CBSE 2011), the 2nd ACM SigSoft International Symposium on Architecting Critical Sys-

tems (ISARCS 2011), and the 16th International Workshop on Component-Oriented Programming (WCOP 2011). Furthermore, it was collocated with the 9th Working International Conference on Software Architecture (WICSA 2011). With its rich history and the synergies of the broad software architecture community of CompArch, QoSA blends together topics of software architecture, quality analysis, and model-driven engineering. The call for papers for this theme issue drew heavily from the QoSA and CompArch communities, but also attracted submissions more broadly. We received a total of 25 submissions for this theme issue. These papers were rigorously reviewed by an international group of experts in the fields of quality analysis, software architecture, and model-driven engineering. Based on these reviews, we were able to recommend the following eight papers for publication.

In their paper *Reengineering Component-Based Software Systems with Archimatrix*, Markus von Detten, Marie Christin Platenius, and Steffen Becker introduce Archimatrix, a semi-automatic approach to recover a system's concrete architecture model. Archimatrix combines the detection and removal design deficiencies to (i) achieve a better result in reengineering software architectures and (ii) increase software quality.

Andrea Ciancone, Mauro Luigi Drago, Antonio Filieri, Vincenzo Grassi, Heiko Kozirolek, and Raffaella Mirandola present the reliability prediction capabilities of the KlaperSuite on a series of industrial case studies from the literature and industry in their paper *The KlaperSuite Framework for Model-Driven Reliability Analysis of Component-Based Systems*.

In their paper *Modeling Event-based Communication in Component-based Software Architectures for Performance Predictions*, Chris Rathfelder, Benjamin Klatt, Kai Sachs, and Samuel Kounev present an approach to model event-

---

D. C. Petriu (✉)  
Carleton University, Ottawa, Canada  
e-mail: Dorina.Petriu@sce.carleton.ca

J. Happe  
SAP Research, Karlsruhe, Germany  
e-mail: jens.happe@sap.com

based systems architecture-level and predict their performance.

Lucia Kapova, Barbora Buhnova, and Ralf Reussner evaluated the effect of state-reflecting attributes (like configuration, context, or history-related state of the system) on software performance. In their paper *Stateful Component-Based Performance Models*, they classify their observations and formulate heuristics guiding software engineering and state modeling.

In their paper *Deriving Performance-relevant Infrastructure Properties Through Model-based Experiments with Ginpex*, Michael Hauck, Michael Kuperberg, Nikolaus Huber, and Ralf Reussner introduce the Ginpex approach, which automatically captures the performance properties of (virtualized) infrastructure and include them in performance prediction.

The paper *Extraction and Evolution of Architectural Variability Models in Plugin-based Systems* addresses variability management issues. The authors Mathieu Acher, Anthony Cleve, Philippe Collet, Philippe Merle, Laurence Duchien, and Philippe Lahire develop an automated techniques to extract and combine different variability descriptions, including a hierarchical software architecture model, a plugin dependency model, and the software architect knowledge.

Based on the observation that many uncertain factors do not follow normal distribution, Indika Meedeniya, Irene Moser, Aldeida Aleti, and Lars Grunske looked into *Evaluating Probabilistic Models with Uncertain Model Parameters*. The authors propose an approach using Discrete-Time Markov Chains and probabilistic model checking to accommodate a diverse set of parameter range distributions.

*An architecture framework for enterprise IT service availability analysis* introduced in the paper of Ulrik Franke, Pontus Johnson, and Johan König combines formal availability methods (such as fault trees) and qualitative methods (such as maturity models) to provide a holistic analysis of the availability of IT services.

## Author Biographies



**Dorina C. Petriu** is a full professor in the Department of Systems and Computer Engineering at Carleton University, Ottawa, ON, Canada. Her main research interests are in the areas of software performance modeling and model-driven development, with emphasis on integrating the analysis of non-functional properties into the software development process. She served in the steering and program committees of numerous international conferences and workshops. Dr. Petriu is a Fellow of the Canadian Academy of Engineering and of the Engineering Institute of Canada.



**Jens Happe** is a senior researcher at SAP Research in Karlsruhe, Germany. His Lean Performance Engineering team addresses the integration of performance engineering methods and tools in the software life-cycle of large enterprises. Jens research interests include the automation of performance tests, problem detection through systematic experiments, and root cause isolation.