

# Modeling in Software Engineering

Joanne M. Atlee  
Cheriton School of Computer  
Science  
University of Waterloo  
Waterloo, Ontario, Canada  
jmatlee@uwaterloo.ca

Robert France, Geri Georg  
Department of Computer Science  
Colorado State University  
Fort Collins, Colorado, USA  
{france,georg}  
@cs.colostate.edu

Ana Moreira  
Computer Science Department  
Universidade Nova de Lisboa  
2829-516 Caparica, Portugal  
amm@di.fct.unl.pt

Bernhard Rumpe, Steven Völkel  
Software Systems Engineering  
Technische Universität Braunschweig  
38106 Braunschweig, Germany  
{b.rumpe, voelkel}@sse-tubs.de

Steffen Zschaler  
Institute for Software & Multimedia Technology  
Technische Universität Dresden  
01062 Dresden, Germany  
steffen.zschaler@inf.tu-dresden.de

## ABSTRACT

The Modeling in Software Engineering (MiSE) workshops are a collaboration between the ICSE and MoDELS research communities, with a focus on using models to facilitate software development.

## Categories and Subject Descriptors

D.2 [Software Engineering]: General.

## General Terms

Documentation, Performance, Design, Economics, Reliability, Experimentation, Security, Human Factors, Standardization, Languages, Verification.

## Keyword

Modeling notations, design methodology, design analysis, model-driven engineering, model transformations, model management

## 1. WORKSHOP OVERVIEW

The goals of the software-modeling community are to improve the productivity of software developers and to improve the quality of the resulting software products. Whereas in the past software models might have been viewed primarily as documentation artifacts, the MiSE workshops focus on using models to facilitate software development. Models can be used in all phases and activities surrounding software development and deployment: comprehension of domain, requirements, or design alternatives; analysis of designs or quality attributes; detection of interactions among features or components; reuse of requirements, components, frameworks; automatic generation of software implementations or test suites; or run-time abstractions of the program state. Software models can be highly abstract and informal (e.g., depictions of early design proposals) or can be

precise enough to enable mathematical inference or reasoning. The common theme among all of these uses and representations is the notion of *abstraction* – a software model omits certain details in order to focus attention on other details. The decision of which details to omit depends on how the model is to be used.

## 2. WORKSHOP DESCRIPTION

The one and a half day workshop consists of paper presentations, group discussions, and a panel presentation. The discussions will concentrate on the following research topics:

- **Model composition:** Challenges, methods, and tools to compose models. Model composition is the counterpart of separation of concerns, in which a problem is decomposed into subproblems that are modeled separately. Papers in this category also investigate problems associated with integrating models that describe different and possibly overlapping aspects of a system using different languages.
- **Model-based evaluation of quality:** Model-based techniques for evaluating properties of software designs or programs.
- **Modeling tools:** Design, implementation, and integration of model-development environment and modeling tools. Papers in this category address problems associated with modeling early designs, execution and simulation of aspect-oriented models, and model management and manipulation.
- **Domain- and context-specific modeling and analysis:** Significant advances in modeling and in software engineering can be made by narrowing the scope of the problem under study, and focusing on the problem with respect to a specific domain. Papers in this category look at modeling and analysis of automation machines, embedded systems, and features for automotive systems.
- **Adoption of model-based technologies in practice:** This category discusses experiences, problems, and challenges when adopting Model Driven Engineering in industrial

contexts. Empirical studies serve as a basis to identify future research directions.

### **3. WORKSHOP STRUCTURE**

The MiSE workshop will be held over one and a half days, starting on the afternoon of the first day. On the first day and the morning of the second day, paper presentations will be structured into sessions of thematically related papers. In order to foster lively discussions, each session will be structured as follows:

1. A 15-20 minute presentation per paper by an author, summarising the key points in the paper.
2. A 30-minute slot at the end of the session for open discussions related to or stimulated by the papers presented.

The remaining half-day will be spent in working groups or an open-plan discussion (depending on the number of participants) based on topics collected by session chairs during paper presentations and discussions during the first day of the workshop. Additionally, on the second day there will be a joint session with the workshop on the Role of Abstraction. The workshop will be summarized either in a half-hour of summary

presentations from each working group or a summary presentation by one of the organizers.

### **4. PROGRAM COMMITTEE**

Lionel Briand, Simula, Norway

Manfred Broy, TU Munich

Marcha Chechik, U. of Toronto, Canada

Betty Cheng, Michigan State University

Krzysztof Czarnecki, U. of Waterloo, Canada

Holger Giese, Universität Paderborn

Jeff Gray, University of Alabama at Birmingham

Mats Heimdahl, University of Minnesota

Jeff Kramer, Imperial College London

Michael Jackson, Open University

Awais Rashid, Lancaster University

Jon Whittle, George Mason University