Combining Design and Engineering of Interactive Systems through Models and Tools (ComDeisMoto)

Stefan Sauer, Kai Breiner, Heinrich Hussmann, Gerrit Meixner, Andreas Pleuss, Jan Bergh

HAL Id: hal-01597019

https://hal.inria.fr/hal-01597019

Submitted on 28 Sep 2017

Combining Design and Engineering of Interactive Systems through Models and Tools (ComDeisMoto)

Stefan Sauer[1], Kai Breiner[2], Heinrich Hussmann[3], Gerrit Meixner[4],
Andreas Pleuss[5], Jan Van den Bergh[6]

[1] University of Paderborn, s-lab – Software Quality Lab, D-33095 Paderborn, Germany
[2] University of Kaiserslautern, Erwin-Schroedinger-Str, D-67663 Kaiserslautern, Germany
[3] University of Munich, Amalienstr. 17, D-80333 Munich, Germany
[4] DFKI, Trippstadter Str. 122, D-67663 Kaiserslautern, Germany
[5] Lero, University of Limerick, Limerick, Ireland
[6] UHasselt – IBBT, Wetenschapspark 2, 3590 Diepenbeek, Belgium
sauer@s-lab.upb.de, breiner@cs.uni-kl.de, heinrich.hussmann@ifi.lmu.de,
gerrit.meixner@dfki.de, andreas.pleuss@lero.ie, jan.vandenbergh@uhasselt.be

Abstract. Development of interactive systems and their user interfaces combines engineering and design, formal and informal development methods from different domains. Diverse models and tools can be used to support the developers' work. In model-driven development, software systems are specified systematically using dedicated models for different aspects of the system. Yet, appropriate design of user interfaces is as important as functional correctness. This workshop provides a forum of multi-disciplinary discussion on how to integrate model-driven development with the often more informal methodologies used in user-centered design and engineering. Starting point of the discussion are the tools, models, methods and experiences of the workshop participants.

Keywords: user-interface engineering, interaction design, model-driven development, user-centered design, models.

Model-Driven Development of Interactive Systems

Model-Driven Development (*MDD*) has become an important paradigm in software development. In MDD, models are the primary artifacts in the development process. Models are often visual models, like UML models, but can also be represented in textual formats like XML. Each model is denoted by a modeling language that can be tailored to the according stakeholders while still remaining machine applicable.

Analogous to traditional software development, MDD processes leverage models on different levels of abstraction. But unlike traditional software development, the transition between the levels can be described in a formal way and enable (semi-) automatic transformations between the levels, even finally into implementation source code. Having models as development artifacts also enables the application of model checking methods for verification, early testing or tool support.

MDD provides a large number of powerful concepts and tools to deal with models, meta-models, and model transformations. Together, they provide support for systematic and efficient software development.

Model-driven development of interactive systems (MDDIS) and, in particular, their user interfaces (UI) applies the principles of MDD to the target domain of interactive systems and UIs. While traditional m*odel-based user interface development* approaches have not spread widely into practice in the past, requirements of new interaction devices, e.g. demanding for device independence, make model-driven approaches more and more important. One major advantage of MDD is that there are different models for particular purposes. One well-known use case is to separate the content (*what* is displayed) from the design (*how* it is displayed) into distinct models.

This workshop follows the series of workshops on model-driven development of advanced user interfaces (MDDAUI), held in 2005-2010. In the five editions of that workshop series we have seen a lot of models, transformations and tools for model-driven user interface development. These approaches also tackled advanced UI features like multi-platform development and plasticity, context-sensitiveness, multimedia, 3D and augmented reality, ambient production environments, wearable sensors, interactive TV, and many others.

Design and Engineering: Enhancing the User Experience

The previous workshop editions have shown that concepts for model-driven user interface development are already becoming mature. However, the quality of the resulting UIs, in terms of the user experience, has always been one of the most important challenges. In some cases, the UIs generated from the models can even improve the usability as they are for instance very consistent. However, it comes to a drawback if some parts of the UI require individual design e.g. because of its complexity or to increase the likeability of the product.

For these reasons, an optimal development method should support both: systematic MDDIS and incorporation of individual design knowledge, the latter usually resulting from manual, informal methods in user-centered design.

Workshop Goals

This workshop focuses on challenges, opportunities, practical problems, and proposed solutions to integrate MDDIS and informal design methods and tools. It aims at fostering interdisciplinary discussions between the respective expert groups and viewpoints. Workshop participants investigate how models, tools and current or future MDD techniques can be combined with design expertise in a way that improves the development process for interactive systems, better supports the work of UI creators and eventually benefits the user experience of end users. Our objective is improving the tools for software engineers and UI developers, being the creators and users of models, e.g. by integrating design measures in engineering models, providing modeling tools for designers or novel methods for MDD, etc.