3rd International Workshop on Models@run.time

Nelly Bencomo[1] Gordon Blair[1] Robert France[2]
Freddy Muñoz[3], Cédric Jeanneret[4]

[1] Computing Department, Lancaster University, InfoLab21,
Lancaster, UK
[2] Computer Science Department, Colorado State University
Fort Collins, CO, USA
[3] IRISA, INRIA, Equipe Triskell,
Rennes, France
[4] Department of Informatics, University of Zurich,
Zurich, Switzerland

{nelly, gordon}comp.lancs.ac.uk, france@cs.colostate.edu,
fmunoz@irisa.fr, jeanneret@ifi.uzh.ch

Abstract. The third edition of the workshop Models@run.time was held at the ACM/IEEE 11th International Conference on Model Driven Engineering Languages and Systems (MODELS). The workshop took place in the beautiful city of Toulouse, France, on the 30th of October, 2008. The workshop was organised by Nelly Bencomo, Robert France, Gordon Blair, Freddy Muñoz, and Cédric Jeanneret. It was attended by at least 44 people from more than 10 countries. In this summary we present an overview of the presentations and fruitful discussions that took place during the 3rd edition of the workshop Models@run.time.

Keywords: model-driven engineering, reflection, runtime adaptation.

1 Introduction

This year's workshop aimed to build upon the insights gained at workshops held in 2006 and 2007 to better understand the relationship between models produced during development and models used to support and enable runtime monitoring, adaptation and evolution of software. The workshop successfully brought together researchers from different communities. At least forty-four (44) people attended from: Canada, Colombia, France, Germany, Ireland, Israel, Norway, Spain, Switzerland, UK, and the US.

This is the third in a series of MODELS workshops. Therefore, we wanted to take advantage of the experience gained at the two previous editions and focus the discussions of this workshop on the topic: "*from abstract concepts to concrete realizations*".

We aimed to provide a forum for exchange and collaboration among researchers from different communities, including researchers working on model-driven software engineering, software architectures, computational reflection, adaptive systems, autonomic and self-healing systems, and requirements engineering. Thus, the workshop covered a wide range of topics, including relevance and suitability of different model-driven approaches to monitoring and managing systems during runtime, compatibility (or tension) between different model-driven approaches, the role of reflection in maintaining the causal connection between models and runtime systems, experience related to the use of runtime models to adapt software systems, and the use of models to validate and verify behaviour at runtime.

In response to the call for papers, twenty (20) papers were submitted, of which six (6) papers were accepted. Additionally, six (6) short papers were invited for short presentations and a demo was also presented. Each submitted paper was reviewed by at least 3 program committee members. After discussions, two papers were selected as the best papers. The decision was not easy and took into account the relevance of the papers to the workshop theme, the impact on the workshop discussions and outcomes, and the quality of the papers and presentations. We also held a poll of participants to determine their views on which papers were the most noteworthy. The authors of these two papers have now extended and improved their manuscripts taking into account the discussions of the workshop. The extended papers are published in this proceedings.

2 Workshop Format

The workshop was designed to primarily facilitate focused discussion on the use of models during run time. It was structured into presentations, discussion sessions, and a panel. The opening presentation was given by *Nelly Bencomo*. *Nelly* set the context of the workshop reminding the audience of the general goal, and presenting some results from the last two editions of the workshop in MoDELS'06 and MODELS'07. She also described the specific goals of the third edition of the workshop "*from abstract concepts to concrete realizations*" and presented the path to follow during the rest of the day.

After the opening presentation, the paper sessions followed. There were 6 long and 6 short presentations divided in two sessions during the morning. During the afternoon a demo that supports the use of models@run.time was presented, followed by discussion sessions. A panel consisting of three experienced researchers in the area and three representatives from each discussion group discussed how current visions of runtime models can be realized and exploited in practice.

During the presentation session, authors presented their papers. Long presentations were limited to twenty minutes, including five minutes for questions and discussion. Short presentations were limited to five minutes. Presentation sessions were co-chaired by *Øystein Haugen* and *Robert France*. At the end of the presentation session, research interests and questions were discussed. This discussion led to the formation of three breakout groups charged with carrying out more focused discussions during the afternoon.

The panel was chaired by *Gordon Blair* and included *Bran Selic*, *Øystein Haugen*, and *Jean-Marc Jézéquel* who prepared their presentations in advance. The other three members of the panel were chosen by their colleagues during discussion groups. The workshop was closed by a final discussion session, including an evaluation of the workshop made by the attendees. Details of the sessions and panel are provided in Section 4.

3 Session Summaries

The 6 long and 6 short presentations were divided into the following two categories according to their topics and contributions:

Session 1: Specific Techniques for Models@run.time

Long papers

- *Runtime Models for Self-Adaptation in the Ambient Assisted Living Domain*, Daniel Schneider and Martin Becker.

- *FAME---A Polyglot Library for Metamodeling at Runtime*, Adrian Kuhn and Toon Verwaest.

- *Modeling and Validating Dynamic Adaptation*, Franck Fleurey, Vegard Dehlen, Nelly Bencomo, Brice Morin, and Jean-Marc Jézéquel.

Short papers

- *A Runtime Model for Monitoring Software Adaptation Safety and its Concretisation as a Service*, Audrey Occello, Anne-Marie Dery-Pinna, and Michel Riveill.

- *Runtime Models to Support User-Centric Communication*, Yingbo Wang, Peter J. Clarke, Yali Wu, Andrew Allen, and Yi Deng.

- *An Execution Platform for Extensible Runtime Models*, Mario Sanchez, Ivan Barrero, Jorge Villalobos, and Dirk Deridder

Session 2: Architecture and Frameworks for Models@run.time

Long papers

- *Embedding State Machine Models in Object-Oriented Source Code*, Michael Striewe, Moritz Balz, and Michael Goedicke.

- *Model-Based Traces*, Shahar Maoz.

- *Mutual Dynamic Adaptation of Models and Service Enactment in ALIVE*, Athanasios Staikopoulos, Sebastien Saudrais, Siobhan Clarke, Julian Padget, Owen Cliffe, and Marina De Vos.

Short papers

- *A Framework for bridging the gap between design and runtime debugging of component-based applications*, Guillaume Waignier, Prawee Sriplakich, Anne-Francoise Le Meur, and Laurence Duchien.

- *A Model-Driven Approach for Developing Self-Adaptive Pervasive Systems*, Carlos Cetina, Pau Giner, Joan Fons, and Vicente Pelechano .

- *Model-driven Management of Complex Systems*, Brian Pickering, Sylvain Robert, Stephane Menoret, and Erhan Mengusoglu.

A demo illustrating the use of models at runtime opened the afternoon session:

- *K@RT: An Aspect-Oriented and Model-Oriented Framework for Dynamic Software Product Lines*, Brice Morin, Olivier Barais and Jean-Marc Jézéquel.

Following this demonstration, discussions groups were established. Each group received the same questions to discuss. These questions were based on the specific theme of the workshop for that day, "*from abstract concepts to concrete realizations*":

- *Are we ready to make an impact (assessment of state of the art/ promising ideas/ gaps)?*
- *What are the next steps (how to make this impact/ from abstract concept to concrete realization).*

4 Discussions and Panel

After the presentations, the participants were organized into three groups that met in the afternoon. After spending some time discussing the presentations and shared research interests, the groups came back to the meeting room to present a summary of their discussions and positions. The summaries were presented in a panel by panellists representing the groups. The representatives the discussion groups were *Frank Fleurey, Peter J. Clarke*, and *Stéphane Ménoret* who joined *Bran Selic, Øystein Haugen*, and *Jean-Marc Jézéquel*.

The panel started with *Bran Selic* presenting his position. He defined a "runtime model" as a model that is required in the course of software execution. Similar to a design-time model, a runtime model supports reasoning about a complex system, and can assist in the automated generation of implementations. However, in addition, a runtime model supports dynamic state monitoring and control of complex systems during execution, and supports semantic (re-)integration of possibly heterogeneous software elements at runtime (e.g. through the use of dynamically adaptable metamodels).

According to Bran, the role of these runtime models implies some form of automated treatment that involves access to, interpretation, and generation of the model. Bran suggested the following research challenges:

- Develop methods and standards for specifying semantics suited to automated interpretation.
- Achieve reversible model transformations, to deal with synchronization issues.
- Provide support for dynamic model synthesis, for runtime adaptation (e.g. a tool which builds a model of its users as they use it so that it can adapt its user interface to their habits).
- Develop standardized reflective interfaces.
- Discover mechanism for dynamic reclassification of models, which shall bring into light some patterns and methods of modelling or analyzing models.
- Build model execution engines which can execute specification models.

Bran added that these automated treatments must be efficient and responsive to the changing demands of the running system.

Øystein Haugen continued the panel presenting his views on models@run.time by attempting to provide answers to the questions what is it?, what was it?, and what might it be? *Øystein* also talked about his experience working with several software systems such as SIMULA runtime libraries, runtime modelling in the Specification and Description Language (SDL) for a Train Control system (done in the early nineties), and UML model based on state machines an their relevance to the theme of the workshop. *Øystein* also argued that there is no reason why a modeller should be forced to think in terms of a specific programming language, like Java, rather than in UML terms or any domain specific language (DSL) when dealing with a program that is being executed. He supported his claim while running an explaining a demo of a runtime model (in UML2) based on state machine models. The demo showed how runtime models can be used to visualize the position of mobile devices on a map. The adaptive software updates *GoogleEarth* images according to the position of the mobile devices.

Frank Fleurey, as the representative of his discussion group, noted last year’s edition was devoted to the definition of models@run.time and their possible usage and that papers of the workshop edition this year showed that some progress has been made in these directions. In some of these papers, an application is instrumented and feedback is presented to the users at the model level. It has been shown how models@run.time can be used to monitor, validate or debug an application, and to support dynamic adaptation and evolution of software.

Frank’s discussion group agreed that there are several ideas and technologies available to support models@run,time in practice. For example, programming languages include introspection or reflection mechanisms and component frameworks provide adaptation mechanisms. All these elements present a great potential to support the use of models@run.time and many papers presented at this workshop leverage them.

However, one of the papers (“FAME---A Polyglot Library for Meta-modelling at Runtime” by Adrian Kuhn and Toon Verwaest) proposed a modelling framework dedicated to models@run.time and thus casted some doubt on the adequacy of current MDE techniques. If a new modelling formalism must indeed be developed for models@run.time, then practical realizations of models@run.time will be inevitably delayed until appropriate modelling formalism is developed.

Peter Clarke continued the panel presenting the position of his discussion group. *Peter* talked about how maintaining a model of the system at runtime can support the adaptation of a system during execution. Indeed, the runtime model of the system potentially provides an abstraction of the running system allows the administrator or any other monitoring systems to determine properties of the running system and take some action to heal, adapt or evaluate the system.

There was an agreement in his group that one promising use of models@runtime is for answering dynamic "what if" questions during execution. At runtime, models potentially allow the user to "play" with the model before a change is made to the running system. Also, models@runtime may be more effective for those systems that are domain specific or user-centric. Such systems tend to focus on a smaller application space and can be represented at a level of abstraction that can benefit the particular stakeholder. Peter, emphasized that for this to happen, the visualization of models@runtime must be improved so a domain specialist can effectively analyze it.

Stéphane Ménoret represented a discussion group in this panel. According to his group, models@run.time is a research area that requires more attention from industry making reference to the position given by Bran Selic who is also from industry. His discussion group also considered important the maintenance of requirement models during execution to check during execution how requirements agree with the capabilities of the "current" system . A similar initiative is supported by Anthony Finkelstein (requirement reflection) [1]. He also stressed the importance of an international distributed lab initiative focused on models@run.time where different people from different research areas in academia and industry could collaborate together.

For *Jean-Marc Jézéquel*, the notion of models@runtime is an idea that has been (implicitly) around at least fifty years and that was already used implicitly in Simula and more recently with Java, and also with UML/matlab/simulink as presented by Ostein. For example, the class-object pattern can be seen as a model that allows the modification of behaviour. What is new is the possibility to make models evolvable. Jean-Marc sees models@runtime as the intersection of computational reflexion and models and make possible to explore dynamic "what if" situations, to decide whether or not to take a given path during execution. To illustrate his position, *Jean-Marc* gave the example of a fire-fighter (in this case the hypothetical running system) in a room suddenly breaking ablaze, with huge flames and rapidly raising temperature. The fire-fighter builds quickly a mental model of his situation and his options. Once he finds an escape that he estimates safe enough, he runs for it.

Final Remarks at the end of the workshop

[1] Requirements Reflection a short talk presented by Anthony Finkelstein at the Dagstuhl workshop on self-adaptive systems (January, 2008)

A general wrap-up discussion was held at the very end of the workshop. The organizers asked for anonymous written feedback about the selection of the best two papers to publish in this proceeding.

The workshop was closed with a warm "thank you" from the organizers to all participants for another successful workshop. We regret that this time the big number of attendees in the workshop did not allow the organization of the usual dinner we have after the workshop. Instead, attendees dispersed to choose from the many good culinary options that Toulouse offers.

After the workshop

After the workshop and conference, more work was needed. Organizers used the feedback from attendees and program committee members to select the best two papers. After discussion the following papers were selected as the best two papers and are published in new versions in this proceeding:

- *Model-Based Traces* by Shahar Maoz.
- *Modeling and Validating Dynamic Adaptation* by Franck Fleurey, Vegard Dehlen, Nelly Bencomo, Brice Morin, Jean-Marc Jézéquel.

A survey was prepared after the workshop and 24 people answered to this survey. People confirmed that they were pleased with the discussions carried out during the workshop and considered them useful to their own research. They also appreciated the introduction of the panel in the format of the workshop. From the survey and comments at the end of the workshop in Toulouse, it was agreed that the topic models@run.time is relevant for the MODELS community and that this community should be encouraged to continue the study the issues related to the topic.

Acknowledgments. No workshop is successful by the efforts of only a few people. We would also like to thank the members of the program committee who acted as anonymous reviewers and provided valuable feedback to the authors: *Betty Cheng, Fabio Costa, Jeff Gray, Oystein Haugen, Jozef Hooman, Gang Huang, Paola Inverardi, P.F.Linington, Jean-Marc Jézéquel, Rui Silva Moreira, Andrey Nechypurenko, Oscar Nierstrasz, Eugenio Scalise, Arnor Solberg, Thaís Vasconcelos Batista*, and *Steffen Zschaler*. We specially thank to *Bran Selic, Oystein Haugen, Jean-Marc Jézéquel, Peter Clarke, Frank Fleurey*, and *Stéphane Ménoret* for the feedback and ideas provided for this summary. Last but not least, the authors for their interesting submissions are thanked for helping us making this workshop possible.