Validation challenges in model composition: The case of adaptive systems*

Freddy Munoz, Benoit Baudry

INRIA Bretagne Atlantique
Campus de Beaulieu F-35042, Rennes cedex
{fmunoz,bbaudry}@irisa.fr

Abstract. Model Driven Engineering helps dealing with complexity by promoting models as abstraction units. Aspect Oriented Modeling helps separating concerns that crosscut across different models. MDE and AOM have well identified challenges that need to be addressed. However, there are new challenges that appear when combining both techniques. In this paper we present the challenges that appear when validating the model composition in the context of MDE and AOM applied to adaptive systems.

Keywords: Model Driven Engineering, Aspect Oriented Modeling, Model Composition Validation, Model Validation, Model Driven Engineering for Adaptive Systems.

1 Introduction

Model Driven Engineering (MDE) promotes abstraction as a basis for managing complexity. MDE proposes the systematic use of models as primary engineering artifacts. Such models can have a variety of natures and range in abstraction and complexity. Models from higher abstraction level are refined (transformed) into lower levels until the implementation. Besides, models can be transformed from one domain to another in order to ease the resolution of a defined problem.

Aspect Oriented Modeling (AOM) helps separating crosscutting concerns at model level by encapsulating them into different modeling dimensions referred as *aspect*. AOM enables a clear modularization of the different concerns constituting a design model. It also allows designers to reason about each concern separately, and later composed into a global model.

Research challenges have been widely identified for AOM and MDE [11]. Model transformation testing[5], model verification and validation [9], and AOM weaving mechanism definition [2] are just a few examples of the challenges faced by these technologies.

Nevertheless, not all the challenges have been identified as far as validation in MDE and AOM is concerned. Challenges regarding the composition and refinement of aspect models need to be identified. This is critical to ensure that composed

* This work was partially funded by the DiVA project (EU FP7 STREP)

models will perform as expected, and therefore, their refined implementation will do so [4, 8]. This is a fundamental issue for the adoption of AOM as a mechanism for separation of concerns and MDE as a complexity coping mechanism.

In this paper we present the challenges that arise from the validation of model composition. We specially address the challenges that arise in the case of adaptive systems, where models are used to abstract from the executing platform and aspects represent the dynamic variability of the system. Such challenges range from the combinatorial explosion produced by the composition order of different aspects, to the specification of the model resulting from the composition.

The remainder of this paper is organized as follows. Section 2 presents MDE and AOM applied to adaptive systems. Section 3 presents the challenges that arise when validating AOM and MDE in the context of adaptive systems. Finally, section 4 concludes.

2 MDE and AOM for adaptive systems

Designing, developing, maintaining and executing adaptive systems is very complex and error prone. Model Driven and Aspect Oriented techniques can help dealing with this complexity. In this section we present the contribution of MDE and AOM to handle the complexity when dealing with adaptive systems.

Adaptive systems are software systems capable of change their internal structure and behavior in response to changes in their environment [1]. They are typically deployed in heterogeneous computing devices ranging from mobile devices such as phones or PDAs to large computer systems. Generally, several variation points are defined in order to develop an adaptive system. Each variation point represents a different option in the system implementation that might be chosen to adapt the system. The selection of different variation points to derive the adapted system leads to a huge number of possible configurations. Reasoning over that huge set of configurations to choose the best possible configuration to adapt is too time consuming because of the large number of evaluations needed. Moreover, the adaptation logic relies on reconfiguration policies that are generally complex low-level and hand-written in the application producing large and complex reconfiguration files. These factors make the construction, execution and maintenance of adaptive systems highly complex. MDE and AOM help dealing with this complexity by the meaning of abstraction and separation of concerns [19].

MDE techniques provide the means to automate and optimize the creation of reconfiguration scripts. Besides, MDE helps abstracting from the target platform by defining models independent of target devices and technologies. Models representing the system in execution (models at runtime) help to manage the execution at more abstract level; therefore, they enable designers to reason about the system properties and adaptation logic at higher level.

Aspect-Oriented modeling techniques [10, 13, 15, 18] help encapsulating distinct variation points into aspects separated from the base model functionalities. Different aspects might be composed with the base model in order to obtain different configurations. This reduces the reasoning space to a limited number of aspects, therefore avoiding the combinatorial explosion due to different variants.

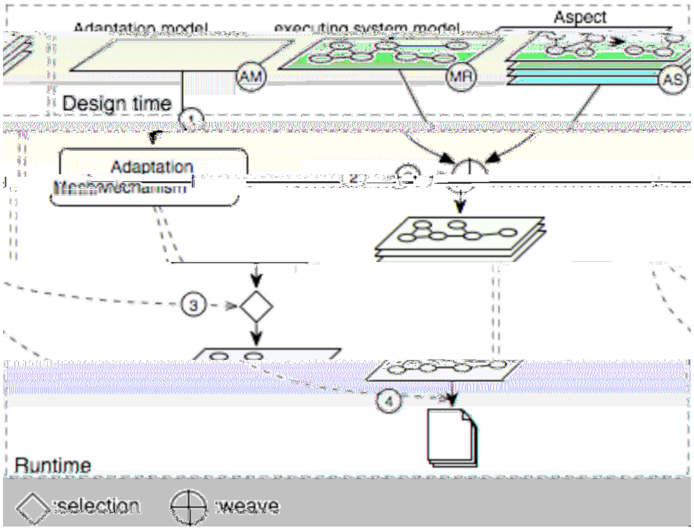

Fig. 1. Overall MDE/AOM approach for adaptive systems

Figure 1, presents an overall approach for adapting systems by using MDE and AOM techniques. At design-time, the application base (AM) and variant architecture (AS) models are designed. At this time, the adaptation model, which states when, and how to adapt is built. At runtime, the adaptation mechanism processes the adaptation model in order to adapt when needed (1). When an adaptation is required, the adaptation mechanism chooses (driven by the adaptation model) a set of aspects (variants) and weaves them into the base (2). This weaving results in multiple models that could be used to adapt the executing system. The adaptation mechanism chooses only one model (3) and then automatically generates the reconfiguration scripts used to adapt the executing system (4).

3 Challenges

MDE and AOM can help dealing with the complexity involved in the life cycle of adaptive systems. However, their usage raises new challenges regarding the validation of model composition. In the following we summarize the challenges and research questions related to the validation in the context of adaptive system.

Validation of composed models (3 in figure 1): The selection of the best possible configuration is critical for adaptive systems. The aspects modifying the base configuration must produce configurations that will not break down the system and that response in the best possible way to environmental changes . Therefore, it is crucial to ensure that the composed models fit the adaptation requirements. A model

issues can be beyond functional interactions. Aspects can have a qualitative impact over the system, for instance making the quality of service better or worst. At code level, the characterization of interactions could be used to determine patterns of interactions for instance to detect aspect interferences [14].

Runtime / Design time validation: Since adaptation happens at runtime, adaptive systems have to respond to hard time and hardware constraints when adapting, a fundamental question is how much of the validation and analysis can be done statically? The ideal will be to calculate at design time *all* the possible interactions and effect of aspects and their possible weaving orders. However, this may not be possible due to the huge amount of possible weaving orders

An idea to cope with these issues is defining contracts on the aspect models. These contracts may be an abstract specification of the effect of the aspects over the system and the interactions between aspects. For instance, they can explicitly declare that an aspect will increase the overall system security but making it slower. They could also allow us to calculate optimal and valid weaving orders. By specifying include/ exclude relations between aspects. Moreover, they may be helpful to detect interactions conflicts at design time, thus saving some computation time when adapting at runtime. The abstract description of the aspects' effect will help determining whether aspects may be valid or not in relation to a base model.

4 Conclusions

In this paper we have identified the challenges that appear when validating the model composition in the context of adaptive systems. Issues such as the weaving order of aspect models, interaction issues and the validation of the composed model are not trivial. Tackling these issues is fundamental to assess the usage of MDE and AOM.

We have pointed out possible ways to address the validation challenges presented here. We specially suggest the definition of contracts to tackle several challenges and ease the solution of others. In future work we will explore how contracts must look like, which information they must contain and how to successfully use them at design time and runtime.

References

1. Aksit, M. and Z. Choukair, *Dynamic, Adaptive, and Reconfigurable Systems Overview and Prospective Vision*, in *23rd Int'l Conf. Distributed Computing Systems Workshops (ICDCSW)*. 2003: Providence, Rhode Island USA.
2. Aldawud, O., et al. *AOM: 11th International Workshop on Aspect-Oriented Modeling*. in *10th International Conference On Model Driven Engineering Languages And Systems*. 2007. Nashville, TN, USA.
3. Aldrich, J., *Open Modules: Modular Reasoning About Advice*, in *ECOOP 2005: 19th European Conference on Object-Oriented Programming*. 2005: Glasgow, UK.

4. Baleani, M., et al. *Correct-by-construction transformations across design environments for model-based embedded software development*. in *Design, Automation and Test in Europe, 2005. Proceedings*. 2005.
5. Baudry, B., et al. *Model Transformation Testing Challenges*. in *IMDT workshop in conjunction with ECMDA-FA 06*. 2006. Bilbao, Spain.
6. Clifton, C. and G.T. Leavens. *Observers and Assistants: A Proposal for Modular Aspect-Oriented Reasoning*. in *FOAL 2002: Foundations of Aspect-Oriented Languages (AOSD-2002)*. 2002. Enschede, The Netherlands.
7. Detlef, P., *Hypergraph rewriting: critical pairs and undecidability of confluence*, in *Term graph rewriting: theory and practice*. 1993, John Wiley and Sons Ltd. p. 201-213.
8. Dion, B., *Correct-by-Construction Methods for the Development of Safety*