Chapter VII
Model-Driven Engineering (MDE) and Model-Driven Architecture (MDA) Applied to the Modeling and Deployment of Technology Enhanced Learning (TEL) Systems:
Promises, Challenges, and Issues.

Pierre Laforcade
Université du Maine, France

Christophe Choquet
Université du Maine, France

Thierry Nodenot
Université de Pau, France

Pierre-André Caron
Université des Sciences et Technologies de Lille, France

ABSTRACT

This chapter deals with the application of model-driven engineering and model-driven architecture approaches in a technology enhanced learning (TEL) context. Such Software Engineering approaches provide concrete benefits (productivity, interoperability, adaptability) by means of intensive uses of models, meta-models and transformations. Such benefits can also be met in a TEL context. Because computer scientists or engineers cannot currently find well-defined frameworks about this new trend, we have chosen to report recent results of our working group (initiated in 2003) in order to provide readers with a survival kit. Our results, illustrated in this chapter, argue that model-driven engineering can help designers to reduce the gap between specific instructional requirements (domain point of view) and the software architectures that practically support the implementation, the run-time and the regulation of this instruction.

INTRODUCTION

Historically, e-learning is considered as a specific information-systems domain among others (see e-business, e-health, and so on) and it is interesting to notice that e-learning draws on research conducted in information technologies and also in the *software engineering* domain: service oriented architectures, component-based development, engineering methods and tools, ontology development, design patterns approaches, etc. are now well-known topics of interest for the e-learning domain.

In a same way, this chapter discusses the application of *model-driven engineering* and *model-driven architecture* approaches in a *technology enhanced learning* (TEL) context. Such *software engineering* approaches can provide tangible and varied benefits (theories or principles but also techniques and concrete tools). Because computer scientists or engineers cannot currently find well-defined frameworks about this new trend, we have chosen to report recent results of our working group (initiated in 2003) in order to provide readers with a survival kit. Our results showed that model-drien engineering can help designers to reduce the gap between specific instructional requirements (domain point of view) and the software architectures that actively support the implementation, the run-time and the regulation of this instruction.

To this end, this chapter is divided into four parts. Firstly, we present (cf. part 1):

1. Current difficulties encountered by designers and developers of TEL systems. They result from the gap between the educational intents and the expression of these intents in the hundreds of lines (XML code, java code, PHP code, SQL queries, etc.) required to correctly tune a given learning management system.
2. Some recent technological and methodological initiatives to counter this lack of an integrated view.

From this background, we present (cf. part 2) the aims and principles of MDE-MDA (such as the three distinct kinds of models: Computer-Independent-Models -CIM, Platform-Independent-Models -PIM, and Platform Specific Models -PSM), and their promises within our context, particularly to facilitate the mapping between the learning activities imagined by designers and the requirements of the infrastructures chosen to deploy these activities.

From various experiments conducted in our laboratories, four challenges are then covered (cf. part 3):

- From CIM to PIM models: development, use of Domain Specific Modeling languages.
- From PIM to PSM models: transformation of Domain Specific Models into code that can be exploited by learning management systems (Moodle, ATutor, etc.).
- From PSM to PIM models: re-engineering of learning activities from execution track exploitation (structured logs, …).
- From PIM to CIM models: mapping, binding and visualization of formal models in order to be human-readable by designers and adapted to their Domain Specific Modeling languages.

In the last part of the chapter (cf. part 4), we discuss current research results and obstacles encountered in deploying educational applications from a model-drien engineering perspective. Finally, from our experiences and the analysis of this growing trend, we propose a list of MDE issues for practitioners.