Chapter 29
Component Based Model Driven Development:
An Approach for Creating Mobile Web Applications From Design Models

Pablo Martin Vera
National University of La Matanza, Argentina

ABSTRACT

Current MDD methodologies are complex to use and require doing lots of models and configurations. Usually after all that effort only some part of the application source code can be automatically created. It would be desirable to have a more simple technique, but powerful enough for automatically creating a fully functional application. This works introduces a component based model driven development approach where a set of user interface components will be configured to define system behavior. Component configuration will be direct, simple and supported by a modeling tool which also includes automatic transformations for reducing the modeling task. The methodology requires the designer to build only two models: a class diagram, representing the data model of the application and a component diagram defining the user interface and the system navigation. Both components are based on UML extended with stereotypes and tagged values allowing configuring the system behavior.

INTRODUCTION

Application modeling is underestimated by the software industry. Many times software development companies, especially small and medium size, don't give enough importance to modeling and consider it a waste of time. In other instances modeling is only used in early stages of the development process for making a first definition of the problem and getting the requirements. Most of the models used in early stages are not updated with the changes that arise in the following stages of the process, so they quickly became obsolete. For that reason, the idea of giving models more importance appears. Models are used for the automatic creation of the application source code, or at least part of it. "Software development automation consists of starting from a high level (or early) representation of the desired software features

DOI: 10.4018/978-1-5225-3422-8.ch029

and deriving a running application out of it, possibly through a set of intermediate steps to enable some degree of user interaction with the generation process." (Brambilla, Cabot & Wimmer 2012).

Developing Software by Making Models and automatically creating source code is a tendency started several years ago. These techniques can be found with several names MDD (Model Driven Development), MDA (Model Driven Architecture), MDSE (Model Driven Software Engineering), MDE (Model Driven Engineering). All these techniques have something in common, they use models and transformations to generate source code. A transformation is a process that takes as input a model and creates a target model or source code. For example OMG MDA Approach (OMG 2003a; Kleppe, Warmer & Bast 2003) uses different kinds of models with different types of abstraction levels, starting from Platform Independent Models (PIM) to Platform Specific Models (PSM). "PIM allows visual representation of the model using a high level of abstraction. Details of the environmental models can be expressed clearly and precisely in UML as it does not use any particular implementation formalism … PSM is developed by mapping a PIM to a particular computer platform and a specific programming language" (Papajorgji, Beck & Braga 2004).

From PIM to PSM there are automatic or semi-automatic transformations. The final goal of these techniques is to automate the creation of the application source code, making designers focusing on the models rather than in the coding process. But most of the existing techniques are difficult to use and require to do a really complex process detailing models and configuring transformations in order to obtain useful code, and most of them only can create part of that code.

By taking the premise of automatic code creation and also making easier the modeling process, a new methodology was created. This methodology uses predefined and configurable user interface components to define system behavior. The use of components in the software development process is a very well establish technique by the software industry (Heineman & Councill 2001). Components are pre-defined pieces of software with a very well establish purpose. "Component-based software development stands for software construction by assembly of prefabricated, configurable, and independently evolving building blocks" (Keller Reinhard, 1998)

Components are excellent for re-use and they are reliable because once they were tested they can be re-use without modification. "Component software benefits include reusability and interoperability, among other" (Adler, 1995). Most of modern programming frameworks include pre-defined components for easiness the development process. For example login component or some UI components like grids, carrousels, etc.

This article is organized as following: first the methodology is introduced justifying why it is mainly focused on mobile web applications. Then Data and User interface models are explained with all configuration capabilities. After that, the modeling of complex applications is shown including views for different roles and complex database design. After describing all capabilities of the methodology, the support tool is shown including the transformation process and the resulting application source code. Finally related works, conclusions and future work are discussed.

Component Based Model Driven Development (CBMDD)

CBMDD is a modeling methodology for designing applications and generating the source code using extended Unified Modeling Language (UML) Models (OMG 2010). "The Unified Modeling Language is a language for specifying, constructing, visualizing, and documenting the artifacts of a software-intensive system." (Booch, Rumbaugh & Jacobson 1999).