Chapter 19

EMC – A Modeling Method for Developing Web-based Applications

Peter Rittgen
University Koblenz-Landau, Germany

Early information systems were mainly built around secondary, administrative processes of the value chain (e.g., accounting). But since the internet came into use, more and more primary processes have become accessible to automation: customer acquisition, ordering, billing and, in the case of intangible goods such as software, even delivery. Hence an increasing part of an enterprise has to be modeled and a substantial part thereof is implemented, usually in an object-oriented programming language like Java. To facilitate this complex task, the MEMO methodology (Multi-perspective Enterprise MOdeling) allows the description of the enterprise on three levels – strategy, organization and information system – and from four angles – process, structure, resources and goals. All partial models for the views are integrated via a common object-oriented core. In this framework we suggest a modeling method for the IS layer, the Event-driven Method Chain (EMC). It is based on the Event-driven Process Chain (EPC) by Scheer, which we adapt to fit both the MEMO methodology and the object-oriented paradigm thus making it suitable for the development of web-based applications. To illustrate this we use the example of a software trading company.

INTRODUCTION

Early information systems were mainly built around secondary, administrative processes of the value chain (e.g., accounting). But since the internet came into use, more and more primary processes have become accessible to automation: customer acquisition, ordering, billing and, in the case of intangible goods such as software, even delivery. Hence an increasing part of an enterprise has to be modeled and a substantial part thereof is implemented. To create such an information system and to adapt it constantly to a changing environment requires a much more efficient software development process than the one suggested by the traditional methods, namely the separation into the phases analysis, design and implementation where each phase is usually performed by a different team, each relying on the documents produced in the previous phase (possibly with backtracking). In these approaches, the coupling between the phases is weak: changes to an analysis model typically require a substantial reorganisation of the design models, which in turn slows down software development considerably. The ARchitecture of Integrated information Systems (ARIS, [Scheer, 1999]) is one such traditional method with a focus on analysis. It received substantial attention both from researchers and practitioners thanks to its close relation to the SAP suite of business applications.

Now, there are several reasons why such methods are not suitable for the development of web-based applications (leading to corresponding requirements):
1. The increasing number of automated business processes requires the integration of these processes with the relevant data. But the parts (views) of conventional models are only loosely coupled (e.g., the data and process models of ARIS). A suitable method for developing web-based applications should integrate partial models, especially the process and the data/object model (model integration).
2. Existing methods do not cater for the needs of object orientation. But the predominant use of object-oriented programming languages in developing web-based software demands the compatibility of the modeling method with object-oriented concepts.
3. Traditional software development is usually quite slow. But electronic markets require a quick adaptation of web applications to changing needs.
4. The development of design models typically consists of reinventing the analysis models in a different (more formal) language. A smooth transition from analysis to design, by merely refining the existing analysis models, is preferable (phase integration).

To solve these problems, we make use of the MEMO framework, which represents a multi-perspective approach to enterprise modeling where all views are strongly connected via a common object-oriented core. Within this frame-
Related Content

The Anatomy of the ArchiMate Language
[www.irma-international.org/article/anatomy-archimate-language/40951/](http://www.irma-international.org/article/anatomy-archimate-language/40951/)

Fault-Tolerant Protocols Using Aspect Orientation
[www.irma-international.org/chapter/fault-tolerant-protocols-using-aspect/5127/](http://www.irma-international.org/chapter/fault-tolerant-protocols-using-aspect/5127/)

Variant Logic for Model Driven Applications
[www.irma-international.org/chapter/variant-logic-for-model-driven-applications/188244/](http://www.irma-international.org/chapter/variant-logic-for-model-driven-applications/188244/)

Assessing the Value of Formal Control Mechanisms on Strong Password Selection
[www.irma-international.org/article/assessing-the-value-of-formal-control-mechanisms-on-strong-password-selection/83632/](http://www.irma-international.org/article/assessing-the-value-of-formal-control-mechanisms-on-strong-password-selection/83632/)

An Improved Motion-Sensing Based Monitoring System for Smart Health Care