Chapter XV

Configuration Management for Reference Models

Robert Braun, Technische Universität Dresden, Germany

Werner Esswein, Technische Universität Dresden, Germany

Andreas Gehlert, Technische Universität Dresden, Germany

Jens Weller, Technische Universität Dresden, Germany

Abstract

In this chapter we analyse how a configuration management system can support reference modelling activities. We argue that a configuration management system can not only record the changes made to a single model but also administer different relations between models. To show how such a system can be useful for reference modelling, we analyse the different relations between models from a reference modelling perspective and formulate our findings as requirements for an ideal system. In the next step, we show how configuration management operations map to these requirements. Finally, we demonstrate that the usage of a configuration management system can significantly increase the productivity of the reference modelling activities.
Introduction

Since their first appearance, models have become increasingly important in the economy. Together with modelling languages and methods, models have significantly influenced the way software systems are developed today. At the beginning of the 1990s, accompanied by new findings in management science, the generally positive experiences with modelling were transferred from software systems engineering to organisational issues (Fettke & Loos, 2003). Established modelling techniques are broadly applied for describing business processes and corporate structures in an organization (Scheer, 2000). The significance of the modelling discipline is embodied by the proposal to define it as the core of the information systems discipline (Weber, 2003).

Understanding models as outcomes of processes and means for knowledge management within an enterprise, all active models are constantly changing. These changes are induced by the flow of the environment of that enterprise. As a consequence, a permanent evolutionary cycle of each model results after the model was initially constructed (see Figure 1).

In order to plan and execute a modelling project, it is essential to record all changes made to all models. This documentation can be used, first, to go back to a model version in the case of a faulty development, and second, to learn from the modifications for future modelling projects (see also method use rationale in Rossi, Ramesh, Lyytinen, & Tolvanen, 2004).

Documenting the modifications of an artefact in general has a long tradition within the software development and production industry. This tradition was manifested in the ISO 9001:2000 norm. According to this norm, a certification of a company requires a configuration management system (ISO, 2000), which enables a systematic recording of all changes made to artefacts within this company. Additionally, such a configuration management system is a prerequisite to reach CMM level two (SEI, 2002).

Contrary to the software industry, where the artefacts are mainly the application’s source code, we focus here on models only. Consequently, the need for a model configuration management system arises (Esswein, Kluge, & Greiffenberg, 2002). By definition, a model configuration management system includes methods to record, control and manage all changes made to models.

Figure 1. The general evolution cycle of a model