

Chapter 28

Viewpoint-Based Modeling: A Stakeholder-Centered Approach for Model-Driven Engineering

Klaus Fischer

German Research Center for Artificial Intelligence, Germany

Julian Krumeich

German Research Center for Artificial Intelligence, Germany

Dima Panfilenko

German Research Center for Artificial Intelligence, Germany

Marc Born

ikv++ technologies, Germany

Philippe Desfray

SOFTEAM, France

ABSTRACT

Viewpoint-based modeling is an important recent development in software engineering. It is likely to boost the wider use of modeling techniques because it allows the tailoring of existing tools with respect to the different stakeholders in software design. This chapter reports the results of the project ViBaM, in which viewpoint concepts are investigated. In doing so, the authors give an overview of the most important contributions from literature regarding viewpoint concepts, from which they derive the position that they take in the ViBaM project. After presenting ViBaM's position, the authors derive features that they consider important for tools that support viewpoints. Afterwards, they present use cases, in which the viewpoint concepts are illustrated and discuss which of the viewpoint features are relevant in these use cases.

DOI: 10.4018/978-1-5225-3923-0.ch028

INTRODUCTION

In recent years, the complexity of software development has been continuously growing (Lehtola, Gause, Dumdum, & Barnes, 2011), since developers have to face more and more technical and business requirements in shorter time periods while dealing with an increasing number of internal and external stakeholders and participants (Bosch & Bosch-Sijtsema, 2010; Mohagheghi, Fernandez, Martell, Fritzsche, & Gilani, 2009). As a result, a new software development concept called Model-driven Software Engineering (MDSE) has recently emerged in research, but is also more and more used in business practice. MDSE aims at a stronger and more efficient incorporation of domain knowledge into the software development process to overcome the just mentioned difficulties (Schatz, 2011). In doing so, MDSE considers models as a basis for illustrating the system to be implemented. The basic idea is to create and refine the models of the desired system by using elaborated tools. As one of the main objectives of MDSE (Schatz, 2011), special attention is given to a clear distinction between the specification of the modeled system and its technical implementation in order to foster the consideration of domain knowledge.

The Model Driven Architecture (MDA) (Object Management Group, 2003) of the Object Management Group (OMG) has established as the most common approach to MDSE. To follow the idea of MDSE, MDA introduces a strong separation of concerns regarding modeling a system at different abstraction levels. This starts from computational (CIM) and platform-independent (PIM) models and uses transformations to produce the actual code for the selected programming language and platform (PSM).

In present days modeling techniques are, not only due to MDSE, widely used—in particular standard modeling languages such as the Business Process Modeling Notation (BPMN) (Object Management Group, 2011a) or the Unified Modeling Language (UML) (Object Management Group, 2011b) but also proprietary and domain specific modeling languages. However, according to our estimation, usage of these techniques has already reached a significant threshold in software development but a wider usage and diffusion needs more added values and perceived ROI by end users. Modeling can gain a stronger acceptance by supporting cooperative and collaborative system design in a more efficient and simplified manner. In such settings, different stakeholders work together to design a system. However, each of them has his or her own perspective to the system design and would like to view and manipulate the models or model fragments according to his or her own needs. In addition, the usage of models to support method frameworks (most of them using viewpoints) will be much improved through this new capacity.

To foster the modeling usage with regard to these demands, the concept of Viewpoint-based Modeling is an emerging approach in the course of software development. However, the concept does not have its origins in software development, but its general idea can already be found in the overall context of constructing large-scale systems several years ago (Finkelstein, Easterbrook, Kramer, & Nuseibeh, 1992; Wood-Harper, Antill, & Avison, 1985). Since 1990, this concept is largely used in method frameworks such as RM-ODP (International Organization for Standardization, 1996), RUP (Kruchten, 2003), or TOGAF (The Open Group, 2011). Accordingly, its major goal is to reduce complexity by adapting the overarching model to stakeholder specific fragments in a successful manner. In doing so, stakeholders are being more put into the focus of the modeling process, which results in a higher perceived value by them. By utilizing stakeholder-specific viewpoints on a model, its overall understanding and productivity increases (Finkelstein et al., 1992). As a direct consequence, the viewpoint concept leads also to better conceptual models, which was proven by several studies. For example, according to the study conducted by Easterbrook et al. (2005) applying the viewpoint concept significantly helps to cope with the overall size of a given problem domain.

24 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/viewpoint-based-modeling/192898

Related Content

Periodic Patterns in Dynamic Network: Mining and Parametric Analysis

Hardeo Kumar Thakur, Anand Gupta, Anshul Garg and Disha Garg (2018). *Multidisciplinary Approaches to Service-Oriented Engineering* (pp. 244-264).

www.irma-international.org/chapter/periodic-patterns-in-dynamic-network/205302

Semi-Automated Tool Support for Identification and Prioritization of Impacted Functions in Software Systems

Chetna Gupta and Varun Gupta (2018). *Multidisciplinary Approaches to Service-Oriented Engineering* (pp. 168-181).

www.irma-international.org/chapter/semi-automated-tool-support-for-identification-and-prioritization-of-impacted-functions-in-software-systems/205298

Cloud Build Methodology

Richard Ehrhardt (2021). *Research Anthology on Recent Trends, Tools, and Implications of Computer Programming* (pp. 108-132).

www.irma-international.org/chapter/cloud-build-methodology/261024

Methodology for ISO/IEC 29110 Profile Implementation in EPF Composer

Alena Buchalcevova (2021). *Research Anthology on Recent Trends, Tools, and Implications of Computer Programming* (pp. 422-438).

www.irma-international.org/chapter/methodology-for-isoiec-29110-profile-implementation-in-epf-composer/261037

Cloud Security Threats and Techniques to Strengthen Cloud Computing Adoption Framework

Nabeel Khan and Adil Al-Yasiri (2018). *Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications* (pp. 268-285).

www.irma-international.org/chapter/cloud-security-threats-and-techniques-to-strengthen-cloud-computing-adoption-framework/203510