

Chapter 29

DSLs in Action with Model Based Approaches to Information System Development

Ivan Luković

University of Novi Sad, Serbia

Vladimir Ivančević

University of Novi Sad, Serbia

Milan Čeliković

University of Novi Sad, Serbia

Slavica Aleksić

University of Novi Sad, Serbia

ABSTRACT

*In this chapter, the authors give an overview of the evolution of Information System (IS) development methods used in the last few decades and show how model driven approaches and Domain Specific Languages (DSLs) have managed to take an often essential role in the modern IS development process. To present an overall picture, the authors discuss significant breakthroughs, popular approaches, their strong and weak points, along with the examples of their practical use in the domain of IS development and generation of software applications. In order to further support the aforementioned points, the chapter offers a synopsis of Integrated Information Systems CASE Tool (IIS*Case), a model driven software development tool for IS modeling and prototype generation. A special attention is drawn to its evolution and position relative to some of the key changes in IS development approaches in recent history. The authors highlight the significance of DSLs in this context and present a DSL featured in the tool. The DSL was created to provide platform independent model (PIM) IS specifications which can be transformed into executable application prototypes through a chain of model-to-model and model-to-code transformations. Since the authors have developed both a textual DSL, and visual repository-based tools (visual DSLs) for this purpose, a discussion of pros and contras of textual vs. visual DSLs in the context of creating PIM specifications is also included. Furthermore, the chapter communicates practical experiences about creating meta-meta models of PIM concepts by means of attribute grammars and MOF meta-modeling language.*

DOI: 10.4018/978-1-4666-4301-7.ch029

INTRODUCTION

As the information technology constantly grows and evolves, new possibilities for its incorporation in ISs continue to appear. Although this enriches those systems with novel capabilities, “optimal” methods and techniques to develop an IS are still far from obvious. There are many ways and approaches that may be deployed to improve the process of IS development. Our focus is on the usage of DSLs based on model driven approaches. Through our research, we have created a textual language aimed at modeling PIM specifications of an IS. Our research goals are to couple it with Integrated Information Systems CASE Tool (IIS*Case). IIS*Case is a model driven software development tool that provides IS modeling and prototype generation. At the level of PIM specifications, IIS*Case provides conceptual modeling of database schemas and business applications. Starting from such PIM models as a source, a chain of model-to-model and model-to-code transformations is performed in IIS*Case to obtain executable program code of software applications and database scripts for a selected target platform.

One of the main motives for developing IIS*Case is in the following. For many years, the most favorable conceptual data model is widely-used Entity-Relationship (ER) data model. A typical scenario of a database schema design process provided by majority of existing CASE tools is to create an ER database schema first and then transform it into the relational database schema. Such a scenario has many advantages, but also there are serious disadvantages (Luković, 2009). Deploying IS design methodologies based on the techniques such as ER modeling or general purpose family of languages UML, and even the relational data model and an appropriate CASE tool, requires advanced knowledge, skills, and high perception power. Failing to find an appropriate number of designers that possess these properties may lead to a risk of designing poor quality ISs (Kosar et

al., 2010). Besides, these methods and techniques are often incomprehensible to end-users. In practice, that may lead to problems in communication and to misunderstanding between designers and end-users. As a rule, misunderstanding results in a poorly-designed database schema or a software application, because support of all the specified user requirements is not ensured. Usually, both designers and end-users become aware of that too late, when the database schema or a software application is already implemented.

To overcome these disadvantages, we created an alternative approach and related techniques that were mainly based on the usage of model driven software development (MDSD) (Bézivin, 2004) and DSL paradigms (Deursen et al., 2000; Mernik et al., 2005). The main idea was to provide the necessary PIM meta-level concepts to IS designers, so that they can easily model semantics in an application domain. Afterward, they may utilize a number of formal methods and complex algorithms to produce database schema specifications and IS executable code, without any considerable expert knowledge.

In order to provide design of various PIM models by IIS*Case, we created a number of modeling, meta-level concepts and formal rules that are used in the design process. In the following text we name all these concepts and rules as IIS*Case Meta-model. In early stages of our research, we formally specified IIS*Case Meta-model by means of predicate calculus formulas, and additionally described its semantics in a form of free text (Mogin et al., 1994). Later on, we also developed and embedded into IIS*Case visual and repository based tools (visual DSLs) that fully apply IIS*Case Meta-model. They assisted designers in creating formally valid models and their storing as repository definitions in a guided way.

Apart from having IIS*Case Meta-model represented as repository definitions, there is a strong need to have its equivalent representation given in a platform independent way. By this, it becomes fully independent of repository definitions that typically may include some imple-

29 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/dsls-action-model-based-approaches/77724

Related Content

Trends and Prospects of Telemedicine

A. H.M. Razibul Islam, Rashida Begum and A. B.M. Shawkat Ali (2009). *Handbook of Research on Modern Systems Analysis and Design Technologies and Applications* (pp. 584-608).

www.irma-international.org/chapter/trends-prospects-telemedicine/21089

Knowledge Extraction from a Computational Consumer Model Based on Questionnaire Data Observed in Retail Service

Tsukasa Ishigaki, Yoichi Motomura, Masako Dohi, Makiko Kouchi and Masaaki Mochimaru (2010). *International Journal of Systems and Service-Oriented Engineering* (pp. 40-54).

www.irma-international.org/article/knowledge-extraction-computational-consumer-model/44685

Input/Output Ports

(2017). *Microcontroller System Design Using PIC18F Processors* (pp. 94-119).

www.irma-international.org/chapter/inputoutput-ports/190446

What Practitioners Think of Inter-Organizational ERP Requirements Engineering Practices: Focus Group Results

Maya Daneva and Niv Ahituv (2013). *Frameworks for Developing Efficient Information Systems: Models, Theory, and Practice* (pp. 270-297).

www.irma-international.org/chapter/practitioners-think-inter-organizational-erp/76627

Metamorphic Relations Based Test Oracles for Image Processing Applications

Tahir Jameel, Mengxiang Lin and Liu Chao (2016). *International Journal of Software Innovation* (pp. 16-30).

www.irma-international.org/article/metamorphic-relations-based-test-oracles-for-image-processing-applications/144139