Chapter XVII

Model-Driven Integration in Complex Information Systems: Experiences from Two Scenarios

Sven Abels
Abelssoft GmbH, Germany

Wilhelm Hasselbring
University of Kiel, Germany

Niels Streekmann
OFFIS – Institute for Information Systems, Germany

Mathias Uslar
OFFIS – Institute for Information Systems, Germany

ABSTRACT

This chapter introduces model-driven integration in complex information systems by giving two practical examples. It relies on the experiences the authors have made in two different research projects at the public utilities domain. The chapter starts with a short introduction of the general problem domain and it gives detailed background information about the current state of the art in model-driven integration. Afterwards, the two research projects are introduced. The purpose of the first project (MINT) was to provide an integration approach allowing interoperability among several different legacy systems. Hence, the project itself was only acting as a “bridge” between the systems. The second project (DER) was built from scratch and got the challenge of integrating several existing third party systems into the newly designed system. In this project, the main system is a core element and only needed to integrate existing legacy systems for specific tasks.
1. INTRODUCTION

Business processes today usually involve several different information systems. A study by Marx Gómez and Brehm in 2007 with 658 participating SME companies in Germany turned out that 90.3% of all companies are using more than one product for their financial business needs (i.e. ERP related tasks). More then 53% are using 4 or more products and almost 15% of all companies are using 10 or more different products, most of them being produced by different software vendors (Details can be found in Marx Gómez & Brehm, (2007)). Considering those figures reveals the strong need for integrating different software systems into a coherent solution. This is usually achieved by creating interoperability between software systems. As defined by the IEEE, interoperability is “the ability of two or more systems or components to exchange information and to use the information that has been exchanged” IEEE, (1990). At the I-ESA 2007 conference, Jeusfeld argues that this topic is often neglected in the design of modern information systems (Jeusfeld, (2007)). For example, he describes that the well known Software Engineering Body of Knowledge (Swebok, (2004)) mentions interoperability only twice, once as an example for a system requirement and the second time as a title of a standard library of data models. Interoperability and the possibility to integrate different heterogeneous systems in a coherent architecture is, however, a key of the MDA strategy as defined by the OMG (Object Management Group) (OMG, (2003)). The following sections focus on this complex area and they put in context of the Model-Driven Software Development (MDSD) approach:

- CIM which is an abstract description of the system, mostly created by domain experts.
- PIM that defines the “What and How” of an information system independently from the actual technology.
- PSM that describes the “What and How” in a technologic dependent model and

2. BACKGROUND ON THE TECHNIQUES USED

Integration of software systems may take place on different levels. The OMG defines CIM (Computation Independent Model), PIM (Platform Independent Model), PSM (Platform Specific Model) and code levels. Those are defined and described in detail earlier in this book. We will therefore focus on putting those levels into the domain of our specific problem of integrating information systems. Considering this, the following figure visualizes the current state of the art using CIM, PIM and PSM as different stages of abstraction.

The figure shows two different information systems with their levels of abstraction. Within one system, the OMG defines the following levels that can be distinguished when modelling, creating and refactoring systems in the Model-Driven Software Development (MDSD) approach:

- CIM which is an abstract description of the system, mostly created by domain experts.
- PIM that defines the “What and How” of an information system independently from the actual technology.
- PSM that describes the “What and How” in a technologic dependent model and