7th International DAAAM Baltic Conference
INDUSTRIAL ENGINEERING
22-24 April 2010, Tallinn, Estonia

SERVICE ORIENTED AND MODEL-DRIVEN DEVELOPMENT METHODS OF INFORMATION SYSTEMS

Lemmik, R.; Karjust, K.; Koov, K.

Abstract: *The main purpose of the current paper is to introduce information systems development concepts and chosen technologies for E-Manufacturing solutions. The main goal is to achieve interoperable systems with high efficiency and reliability by using MDA (Model Driven Architecture) and SOA (Service Oriented Architecture) technologies. Current scope focuses to MDA development where custom application is automatically deployed into Alfresco environment by using Eclipse modeler with Topcased layer together with SIDE-Labs application generator. The MDA toolset will be evolved to generate ERP implementation model with custom functionality for MS Dynamics NAV.*

Key words: model driven architecture (MDA), service oriented architecture (SOA), information systems development, enterprise information systems, enterprise service bus (ESB)

1. INTRODUCTION

The increasing competitiveness in global market highlights the importance of design quality, productivity, multi-company collaboration, optimal price levels and predictability. The manufacturers are now under increasing pressure to maintain their places in the market. To improve their ability to innovate, get products to market faster, and reduce errors, the manufacturers have been continuing to improve their product development and management abilities. Because of that in the past years have seen growing investments in the area of product lifecycle management (PLM) and enterprise resource planning (ERP) [1, 2, 3, 4].

To implement new standards and evolve new methodologies for information systems development was in 1989 founded the Object Management Group (OMG), an international, open membership, not-for-profit computer industry consortium.

OMG's Mission: With our members, to create software standards that improves the process of developing complex applications while increasing ROI [5].

OMG's modeling standards, including the Unified Modeling Language™ (UML®) and Model Driven Architecture® (MDA®), enable powerful graphical or textual design, execution and maintenance of software and other processes, including IT Systems Modeling and Business Process Management [5].

OMG issues rapidly growing suites of both industry-specific and general standard software. Based on OMG's flagship Model Driven Architecture® (MDA®), these standards implement business functions equivalently and interoperably on virtually all popular middleware platforms including Web Services, XML/SOAP, C#/.Net, Enterprise JavaBeans, and others in addition to OMG's own CORBA [5].

2. SERVICE ORIENTED ARCHITECTURE

SOA is concerned with the independent construction of services which can be combined into meaningful, higher level business processes within the context of the enterprise.

Typical “Spaghetti” Enterprise Application Integration where adapters interface with applications at their integration points is presented in Fig.1.

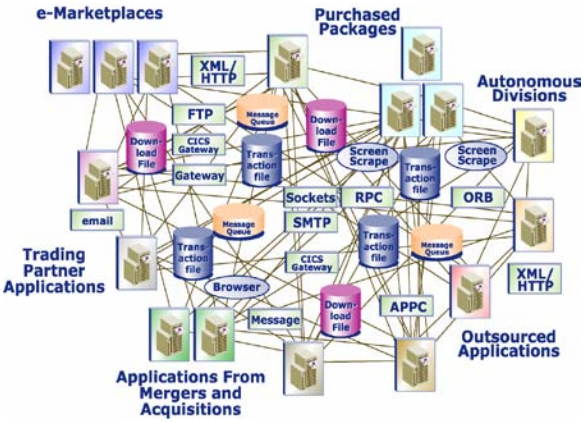

Fig.1. “Spaghetti” Enterprise Application Integration [5]

A Service Oriented Architecture presented in Fig.2 describes several aspects of services within an enterprise: [5]

- The granularity and types of services
- How services are constructed
- How the services communicate at a technical level
- How the services are combined together (i.e. orchestrated)
- How the services interoperate at a semantic level (i.e. how they share common meanings)
- How services contribute to IT and Business Strategy

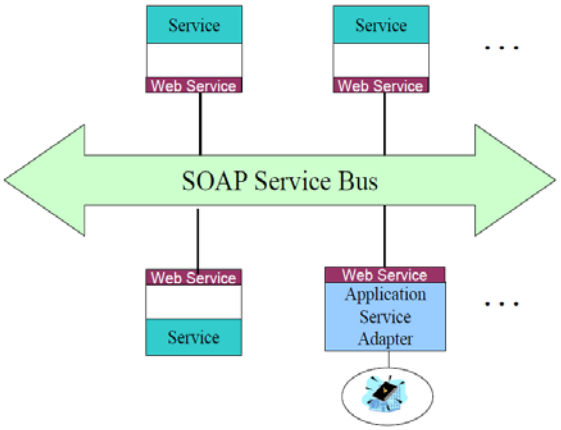

Fig.2. SOA Enterprise Application Integration [5]

Based on SOA the special Service Oriented Integration (SOI) methodology is evolved:

- An architectural and technology based approach to exposing and integrating existing applications as services
- Builds on EAI technology, using new Web services based platforms
- Exposes services to a bus, not point-to-point
- Extends SOA to integration solutions

Benefits of SOI Approach presented in Fig.3 [5].

- Each system is integrated once into the service bus, rather than many time for each point-to-point connection
- Multiple services can be easily constructed from the integration of existing applications
- New processes can be constructed from the service
- Layered SOI approach enables quickly reconfiguring processes or services without needing to change operational systems
- Layered SOI approach allows operational systems to change without affecting business processes

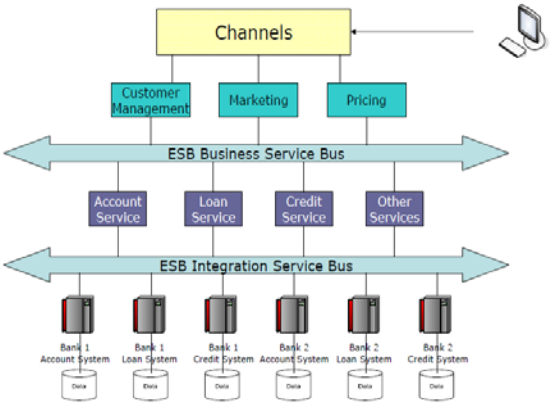

Fig.3. SOI Enterprise Application Integration [5]

The Enterprise Service Bus (ESB) provides an ideal platform for SOI applications and SOI combines Web service, EAI and SOA.

3. MODEL-DRIVEN ARCHITECTURE

Model-driven architecture (MDA) presented in Fig.4 is a software design approach for the development of software systems. It provides a set of guidelines for the structuring of specifications, which are expressed as models. Model-driven architecture is a kind of domain engineering, and supports model-driven engineering of software systems [8].

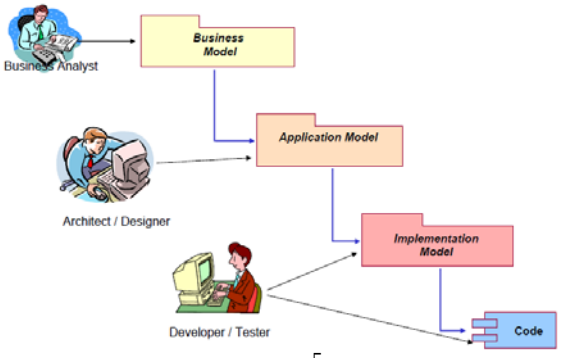

Fig.4. MDA Overview [5]

Basically, an MDA tool is used to develop, interpret, compare, align, measure, verify, transform, etc. models or meta-models. In any MDA approach we have essentially two kinds of models: initial models are created manually by human agents while derived models are created automatically by programs [8].

An MDA tool may be one or more of the following types [6]:

- **Creation tool** used to elicit initial models and/or edit derived models.
- **Analysis tool** used to check models for completeness, inconsistencies, or error and warning conditions.
- **Transformation tool** used to transform models into other models or into code and documentation.
- **Composition tool** used to compose (i.e. to merge according to a given composition semantics) several source models, preferably conforming to the same meta-model.

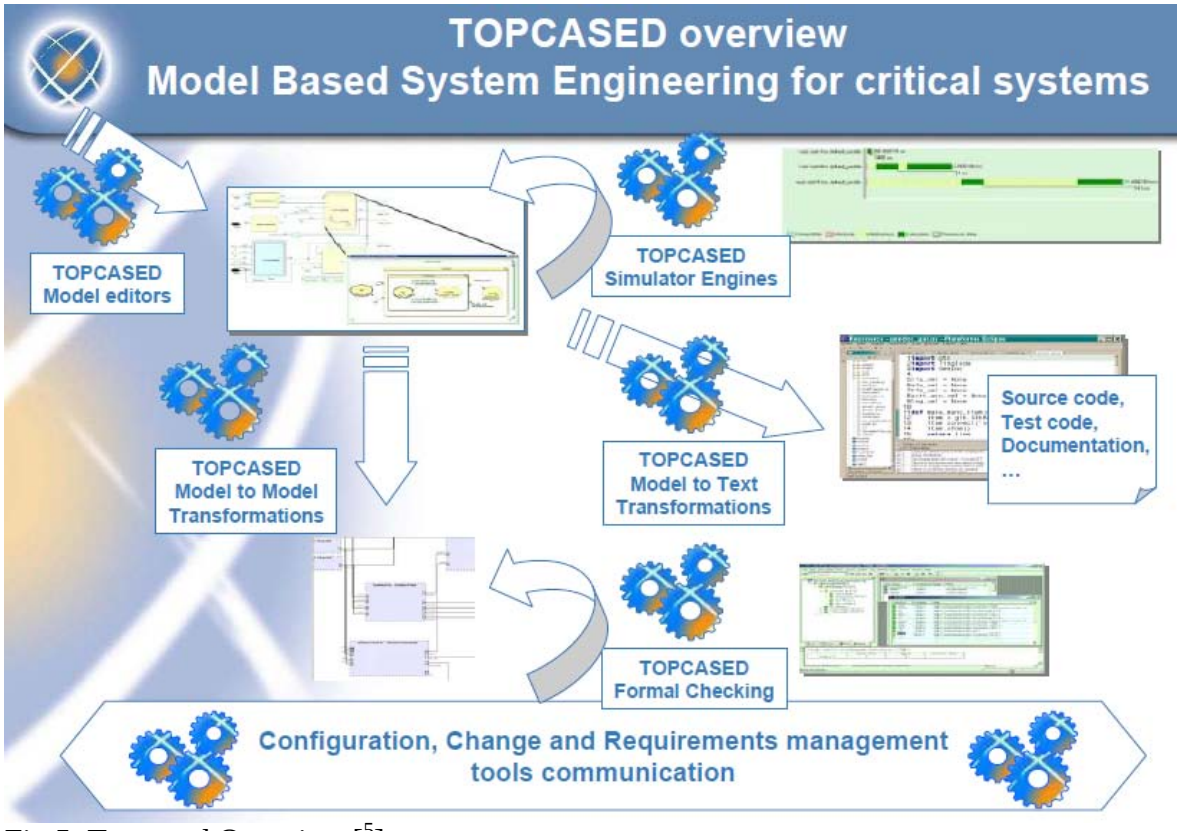

Fig.5. Topcased Overview [5]

- **Test tool** used to "test" models as described in Model-based testing.
- **Simulation tool** used to simulate the execution of a system represented by a given model. This is related to the subject of model execution.
- **Metadata Management tool** intended to handle the general relations between different models, including the metadata on each model and the mutual relations between these models
- **Reverse Engineering tool** intended to transform particular legacy or information artifact portfolios into full-fledged models.

Topcased toolkit presented in Fig.5 is used for the current project. Topcased is a software environment primarily dedicated to the realization of critical embedded systems including hardware and/or software. Topcased is based on Eclipse and it promotes model-driven engineering and formal methods as key technologies.

4. SIDE-LABS FOR ALFRESCO

SIDE is an Open Source environment to model vertical web application layers and to generate the artifacts of these layers on specific vertical web frameworks.
Modeling allows capturing the complete sets of functional needs of an artifact for the benefits of one single or many web applications during all their life-cycle.
Generation presented in Fig.6 allows producing the code and configuration of an artifact for the concerned applications on one single or many targeted web frameworks successively during all the applications' life-cycle.
SIDE is composed of 5 main products: [7]

- A set of Meta-models: they describe a precise layer of the targeted web applications. For instance, the data meta-model describes the content types and their association which may be declined in an ECM application.
- A set of Modelers: a SIDE modeler is an Eclipse plug-in or a web application enabling the graphical modeling of artifacts of targeted functional layers through the production of models. It exists at least one modeler per meta-model and a modeler is conform to a single meta-model.
- A set of Generators: a SIDE generator is a Model to Model and Model to Text Java-based program enabling the transformation of the models on other models for refinement or on a specific web framework for application instantiation. A generator processes a set of models of a single meta-model.
- A set of Deployers: a SIDE deployer is a Java-based component which produces the packaging of the generating artifacts and deploys it on the targeted framework. For instance, the Alfresco deployer produces war files from the generated AMP files. A deployer is associated to a specific framework and is able to package generated artifacts relative to various meta-models.
- A set of advanced server components: In order to extend some features of the targeted frameworks, SIDE provides a set of modules which allows benefiting of very practical and intuitive features associated to the generated artifacts of designed models. These modules are provided by some generators and are deployed on requests by the concerned deployer. For instance, the facet navigation module allows integrating facet navigation on Alfresco.

SIDE is based on the MDA (Model Driven Architecture) to model and to generate web application components.
Currently, SIDE generates these artifacts on the following frameworks:

- ECM Alfresco 2.x and 3.x,
- Chiba 1.x Xforms engine,
- Liferay 5.x portal.

The SIDE modeling environment and the generators are mainly based on Eclipse and the MDA standard libraries like Topcased, EMF and ATL.

The MDA approach on which is based SIDE suggests to model functional layers using DSL (Domain Specific language) languages able to capture the functional invariant in models and able to generate these models on specific frameworks: this allows to make perennial a part of the application through models which offers a longer life cycle than the code of the frameworks, always outdated. [7]

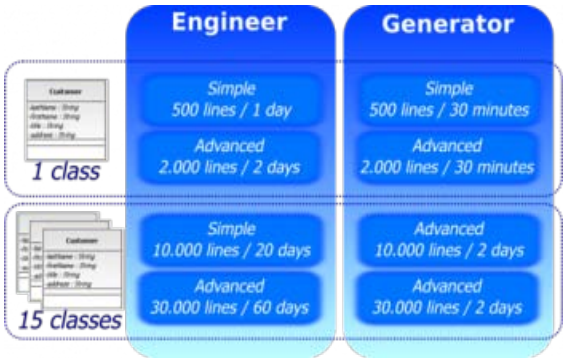

Fig.6. SIDE Code Generation [7]

5. CONCLUSION

During current project will be evolved SOI framework for integration of the E-Manufacturing information systems, with integration adapters for Alfresco Enterprise Content Management, MS Dynamics NAV Enterprise Resource Planning and Siemens Product Lifecycle Management systems. MDA tool Topcased with SIDE-Labs is used to implement custom content models in Alfresco. In the second phase of the project the MDA toolset will be evolved to generate ERP implementation model with custom functionality for MS Dynamics NAV.

6. ACKNOWLEDGEMENTS

This research was supported by Innovative Manufacturing Engineering Systems Competence Centre IMECC co-financed by Enterprise Estonia (EAS) and European Union Regional Development Fund (project EU30006).

7. REFERENCES

1. Tang, D. Qian, X. Product lifecycle management for automotive development focusing on supplier integration. Computers in Industry, 2008, **59**, 2-3, 288-295.
2. Sudarsan, R. Fenves, S.J. Sriram, R.D. Wang, F. A product information modeling framework for product lifecycle management. Computer-Aided Design, 2005, **37**, 13, 1399-1411.
3. Küttner, R. A Framework of collaborative Product and Production Development System. In Proc. 3rd International Conference "Industrial Engineering – New Challenges to SME", Tallinn, 2002, 34-37.
4. Berchet, C. Habchi, G. The implementation and deployment of an ERP system: An industrial case study. Computers in Industry, 2005, **56**, 6, 588-605.
5. Object Management Group; http://www.omg.com
6. Topcased; http://www.topcased.org
7. Sustainable IDE; http://www.side-labs.org
8. Lemmik, R. (2008); An OLAP Cube Based Development Method of Information Systems, Tallinn University, Master Thesis

8. CORRESPONDING AUTHORS

Rivo Lemmik, MSc, Innovative Manufacturing Engineering Systems Competence Centre (IMECC), Akadeemia tee 19, 12618 Tallinn, Estonia, E-mail: rivo@raadi.net
Kaspar Koov, IMECC, Akadeemia tee 19, 12618, Tallinn, Estonia.
E-mail: kasparkoov@gmail.com
Assoc. Prof. Kristo Karjust, Department of Machinery, Tallinn University of Technology, Ehitajate tee 5, Tallinn, 19086, Estonia, E-mail: kristo@staff.ttu.ee