ARCHITECTING SOA SOLUTIONS FROM ENTERPRISE MODELS

A Model Driven Framework to Architect SOA Solutions from Enterprise Models

Xabier Larrucea, Gorka Benguria
European Software Institute, ParqueTecnologico de Zamudio ,E-48170 Zamudio-Bizkaia, Spain

Keywords: Enterprise Models, Service Oriented Architecture, Platform Independent Model, Model Transformation.

Abstract: The improvement of the operational efficiency is an important concern in the several kinds of enterprises, but it involves the management of a multitude of elements. To be able to cope with such as complexity several enterprises are relaying in the use of enterprise modelling tools. This usually becomes a starting point for business process automation initiatives towards the improvement of the organisation. However, there is still a large gap from these enterprise models to the infrastructure systems. The current paper presents a MDA (Model Driven Architectures) framework over eclipse platform to address this gap for SOA (Service Oriented Architecture) based solutions and more in deep the notation and transformation aspects of the framework. The framework provides a systematic approach for deriving SOA solutions from enterprises models, ensuring that the information systems really implements the models developed by the business experts and no partial interpretations from IT experts.

1 INTRODUCTION

The improvement of the operational efficiency is an important concern in several kinds of organisations from banking (Allen et al., 1996) to healthcare (HayGroup, 2005). This continuous activity requires leveraging the interaction among a multitude of organisational elements from clients and providers, to employees and existing IT systems. To deal with this complexity, organisations describe their structures using enterprise models.

The usage of enterprise models brings several advantages when an organisation is planning to change its actual structure to achieve a greater efficiency. On the one hand, enterprise models describes in a coherent and consistent way all elements involved the operational processes of the organisation at conceptual level. On the other hand, these models allow users to have a common understanding of the enterprise model from different views. Besides, enterprise models establish the basis for the performance analysis of the new models and their latter automation through the intensive usage of enterprise information systems.

Unfortunately, the automation of the enterprise models into enterprise information systems is not a straightforward activity: the business expert develops the improved enterprise model, he meets the information system expert and in somehow he explains what he requires from the information systems; then the information system expert implements what he has understood.

Clearly, there is a gap between enterprise models and their information systems implementations. This separation and differentiation of concerns cause mainly a loss of information, a lack of flexibility, traceability and makes more difficult a consistency check between the enterprise layer and the system layer.

Moreover the introduction of new standardised approaches such as service oriented architectures (SOA) to implement information systems provides many benefits (Fiorano Software, 2004). In summary, they are allowing fast, secure, flexible and automated relationships between enterprises. This makes it possible to achieve higher automation levels of the enterprise models, as this technology allows us to automate our relationship with external partners. The level of automation is increasing and it becomes more necessary to resolve the gap between the enterprise models and their information systems implementations.

Larrucea X. and Benguria G. (2006).
ARCHITECTING SOA SOLUTIONS FROM ENTERPRISE MODELS - A Model Driven Framework to Architect SOA Solutions from Enterprise Models.
In *Proceedings of the Eighth International Conference on Enterprise Information Systems - ISAS*, pages 551-554
DOI: 10.5220/0002461605510554

SOA implementations can be combined to implement ICT systems. For example Web services technology could be used with peer to peer and agents technologies to provide new capabilities. MDA allows the separation of concerns between the logical solutions and the technology used avoiding organisations to reinvent the wheel when there are changes at conceptual or technical layer.

This paper presents a framework to bridge the gap between enterprise layer and technical layer from a Model Driven Architecture (MDA) viewpoint, and the specific mechanisms that uses to represent service architectures and to transform those representations into a platform independent model for service oriented architectures.

This paper is structured in three main sections. Firstly a brief state of the art on this area is provided emphasising the motivation and the start point of our work and approach. The second part describes the mechanisms to architect service oriented architectures solutions taking into account several important enterprise aspects to be modelled. In addition, this section sets up the relationships between business layer and technical layer from a model driven point of view using model transformations. The last section sums up our work and outlines future directions.

2 CONTEXT AND STATE OF THE ART

Enterprise models (EM) allow stakeholders to model their organisations and dimensions (Vernadat, 1996) described in terms of enterprise architectures in a coherent and consistent way. Most of these enterprise models are related to EM tools (GRAI tools, Metis, MO2GO, e-MAGIM, etc) (ATHENA DA1.1,2005)(UEML D1.1,2002). Interoperability problems arise when those organisations aim to achieve enterprise interoperability at a conceptual level. Much effort is spent in European projects (ATHENA, 2005), (INTEROP,2005) to alleviate interoperability issues. Most of these problems are related to the technologies and languages used.

The definition of a well defined metamodels allows a common understanding of the elements described. The standardisation of these metamodels allows tools to interoperate amongst these tools. One of these metamodels is Unified Modelling Language (UML) (UML 2.0, 2003) and its metamodel Meta Object Facility (MOF) (Meta Object Facility,2004). These metamodels are standardised by the Object Management Group (OMG). MOF allows the specification of well defined languages like UML. UML is a de-facto industry standard to specify and to design software systems. UML2.0 is the major revision of this language increasing considerably its capabilities. One of these extended capabilities is the specialisation of UML for specific domains through UML profiles.

The Eclipse platform (Eclipse Modelling Framework,2005) is an open initiative based on plug-ins implementing an essential subset of MOF called essential MOF (EMOF). This platform is also used as a java development platform but our main interest is on its capabilities to define metamodels and to model with respect to a metamodel. For example using the UML2.0 plug-in for the eclipse platform, we are able to specify models that are compliant with UML. However, this open initiative does not provide the graphical implementation of UML and its diagrams. Rational Software Modeller (RSM) and Omondo are UML tools based on the eclipse platform implementing the graphical side of these models. RSM provides facilities to represent profiles.

Models transformations are key pieces within MDA allowing traceability and checking consistency between models. The OMG MOF Query, View and Transformation (QVT) (QVT, 2002) initiative is a language to transform and to query models represented according to MOF metamodels. QVT is still under standardisation process but in the near future it will become an OMG standard. There are two first implementations: Atlas Transformation Language (ATL) (ATL, 2005) and Model Transformation Framework (MTF) (MTF,2005). Both implementations are based on rules and they are used to transform and to query models. MTF as well as ATL is compatible with eclipse platform.

3 A MODEL DRIVEN FRAMEWORK FOR ENTERPRISE MODELS

ATHENA project has developed a metamodel and a UML (Unified Modelling Language) profile called POP* to represent in a common way enterprise models. The Unified Enterprise Modelling Language (UEML,2001) is a POP* predecessor. The main intention of this paper is not to provide a huge description of both metamodels and their differences but to outline that the main difference between them is that POP* is able to represent in its metamodel and profile the following dimensions: process,

organisation, product, decision, and infrastructure. Therefore it increases the model interchange capability with respect UEML amongst commercial EM tools.

From a MDA model driven architecture point of view POP* metamodel is one of the highest architectural levels representing the business aspects that an organisation wants to model. In addition enterprises face up to interoperability issues by adopting service oriented architectures to implement and to publish their business functionality as services. Within the ATHENA project a platform independent model (PIM) metamodel is defined to describe services and their collaborations in a platform independent way. POP* metamodel as well as service oriented architectures are solutions to alleviate interoperability issues in each layer. However they do not resolve the existing gap between the business layer and the technical layer.

Our approach is focused on providing a framework to derive service oriented solutions from enterprise models and to specify a domain language for service oriented solutions. In order to bridge the gap between the business layer and the technical layer a model driven transformation framework has been defined. Figure 1 represents our approach to bridge this gap where business layer is represented separately from technical layer. Two plug-ins for Rational Software Modeller related to two UML profiles are defined in order to represent models compliant with the above metamodels. POP* plug-in is related to POP* UML profile, and PIM4SOA plug-in is related to PIM for SOA UML profile. A set of model driven transformations are also defined to maintain the consistency between UML profiles and their metamodels, and to transform POP* models to PIM for SOA models.

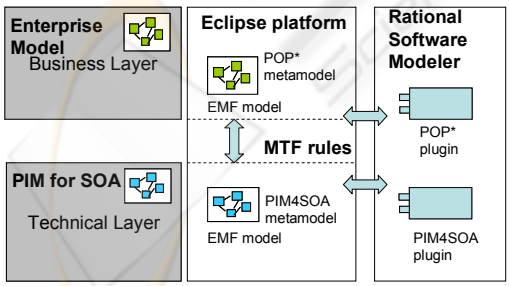

Figure 1: Model driven framework for enterprise models approach.

3.1 PIM for SOA Profile

The UML profile for SOA is based on the aspects and dimensions defined in the PIM for SOA metamodel and it represents graphically UML models depicting service oriented models in a platform independent way. Concepts as well as their relationships are defined in the metamodel but the profile and the plug-in maintain the same concepts and meanings providing a graphical editor.

The PIM4SOA profile defines for each dimension (service, process, information) a set of UML extensions.

Service extensions: “Collaboration” represents the definition of a service. Each service is viewed as collaboration amongst roles. “CollaborationUse” represents the usage of a service. “RoleBinding” is used to relate a service use to a specific role. “Role” represents a structural part in a specific collaboration. “Endpoint” represents the address for a service.

Information extensions: “Document” describes a business document information model. “BusinessTypeLibrary” describes the business logic of the information aspects. “Entity” represents elements used to describe complex types. “TypeLibrary” defines reusable type library.

Process extensions: “Process” represents the behaviour of a service provider. “StructuredTask” is composed by a set of tasks. “Task” is an activity. “Decision” is used to declare a control decision within the process.

3.2 Model Driven Transformations

One of the most important pieces in this framework is the development of the model driven transformations. In this context Model Transformation Framework (MTF,2005) is used to describe those transformations based on rules relating different elements of different metamodels.

At this level three different kinds of transformations are implemented. The first one is used to check the consistency and to transform between UML models representing POP* models and the POP* metamodel represented as an ecore model. The second transformation bridges the gap from enterprise models (POP*) to systems models (PIM for SOA). And finally the third one transforms and checks the consistency between UML models representing SOA solutions and the PIM for SOA metamodel represented as an ecore model. The transformation language is Relation Definition Language which file format is “.rdl”. This language

defines relations between metamodel elements of different metamodels and it reconciles the models involved.

4 CONCLUSIONS

In this paper a MDA approach is applied to address the gap between business models and ICT systems implementations and to build service oriented solutions from a platform independent point of view. A framework based on the eclipse platform is described. This framework contains the introduced POP* metamodel, as a business model, the PIM for SOA metamodel, a set of model driven transformations and a UML profile to describe SOA.

The separation between business models and ICT implementation models assigns flexibility to change elements within models keeping a separation of concerns between metamodels. The implementation of POP* and PIM for SOA metamodels and its model transformation in the Eclipse platform provides a higher independence from the tools and technologies used. The model transformations provide a certain level of traceability between business needs and ICT implementations. The usage of the PIM for SOA UML profile allows users to instantiate service oriented solutions and to transform them as an instance of the PIM for SOA metamodel.

In spite of these benefits this is a weak approach if one of the involved metamodels changes considerably. The existing models must be modified. This task could be resolved using a model transformation from the old metamodel to the new one. In addition this framework is independent form UML tools but the part concerning the PIM for SOA plug-in for Rational Software Modeller must be specialized for each specific UML tool.

One of the future directions is to derive from the PIM for SOA directly BPEL and WSDL code.

Acknowledgments

This work was funded by ATHENA IP (IST-2003-507849). We would like to thanks ATHENA partners for they great feedback.

REFERENCES

Allen, Linda & Rai, Anoop, 1996. "Operational efficiency in banking: An international comparison," Journal of Banking & Finance, Elsevier, vol. 20(4), pages 655-672

ATHENA, 2005. Advanced Technologies for interoperability of Heterogenous Enterprise Networks and Applications project, IST- 507849 http://www.athena-ip.org/

ATHENA DA1.1,2005. First Version of State of the Art in Enterprise Modelling Techniques and Technologies to Support Enterprise Interoperability. http://www.athena-ip.org/

ATL, 2005.Atlas Transformation Language http://www.sciences.univ-nantes.fr/lina/atl/

Eclipse Modelling Framework,2005. http://www.eclipse.org/emf

Fiorano Software, 2004. Service Oriented Architectures Implementation Frameworks. http://www.fiorano.com

HayGroup, 2005. “Annual Benchmarking Comparison of Canadian Hospitals”, http://www.haygroup.ca/services/Benchmarking%20Health/2005/CIHI%202005.pdf

INTEROP, 2005. Interoperability Research for Networked Enterprises Applications and Software project, http://www.interop-noe.org/

Meta Object Facility,2004. Object Management Group (OMG), Document ptc/03-10-04. http://www.omg.org/docs/ptc/03-10-04.pdf

MTF,2005. Model Transformation Framework, http://www.alphaworks.ibm.com/tech/mtf

QVT, 2002. "Revised submission for MOF 2.0 Query/Views/Transformations RFP (ad/2002-04-10), Version 1.8", Object Management Group (OMG), Document ad/04-10-04, 2004. http://www.omg.org/docs/ad/04-10-04.pdf

UEML, 2001. Unified Enterprise Modelling Language project, IST- 34229. http://www.ueml.org

UEML D1.1,2002. Report on the State of the art in Enterprise Modelling http://athena.troux.com/AKMii/WebComputas/TeamBlob.aspx?TeamPage=/Team/Repository/Projects/Project_249/Upload/Attachments/239_30_122__UEMLD11.pdf

UML 2.0, 2003. Infrastructure Specification, Object Management Group (OMG), Document ptc/03-09-15, December 2003. http://www.omg.org/docs/ptc/03-09-15.pdf

Vernadat,F.B. 1996. Enterprise Modeling and Integration: Principles and Applications. Chapman and Hall