XV Jornadas de Ingeniería del Software y Bases de Datos
JISBD 2006
José Riquelme - Pere Botella (Eds)
© CIMNE, Barcelona, 2006

APPLYING A MODEL DRIVEN APPROACH TO AN E-BUSINESS ENVIRONMENT

Xabier Larrucea Uriarte, Gorka Benguria Elguezabal

European Software Institute
Parque tecnológico de Zamudio #204
E-48170 Zamudio, Spain
{Xabier.Larrucea,Gorka.Benguria}@esi.es, http://www.esi.es

Palabras clave: MDA, SOA, business process, metamodels

Resumen. *Traditionally, the implementation of business processes in IT systems is based on the oral transmission of requirements between business and IT experts. This involves a high risk of misunderstanding and loss of information, which may result in the failure of the project, losing time and money. This paper presents the application of a MDA approach to bridge the gap between these domains, business and IT. This is done by applying of a set of automatic transformations, which ensure the coherence between business processes and IT systems. In addition, this paper concludes with several adoption problems and benefits of this approach.*

1. INTRODUCTION

Organisations are looking for efficiency through the usage of enterprise models; these models bring several advantages when an organisation is planning to change its actual structure to achieve a greater efficiency. Enterprise models represent in a coherent and consistent way all organisational elements at conceptual level. In addition, 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.

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

This evident gap between enterprise models and their information systems implementations causes 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.

The introduction of new standardised approaches such as service oriented architectures (SOA) to implement information systems provides many benefits [9]. 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. As the level of automation of our enterprise models increase it becomes more necessary to resolve the gap between the enterprise models and their information systems implementations.

The Model Driven Architecture (MDA) initiative is promoting the usage of models to describe, to build and to deploy systems architectures. Applying a MDA vision to enterprise architectures as well as systems architectures provides a solution to build model based systems to avoid or reduce the lost of information and to increase the separation of concerns, flexibility and traceability.

Moreover, most of the Information and Communication Technologies (ICT) systems implementations are dependent from the technology used. Many implementations exist for SOA and they 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 case study using a framework developed within the ATHENA project [3], to bridge the gap between enterprise layer and technical layer from a model driven architecture 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 briefly the framework used and its technical grounding. The third section describes the case study based on an e-business environment. The last section sums up our work, highlighting the benefits and limitations, and it outlines future directions.

2. CONTEXT AND STATE OF THE ART

Most of organisations interested in enterprise modelling take as a reference enterprise architectures [6], [7]. Enterprise models (EM) allow stakeholders to model their organisations and dimensions [5] described in terms of enterprise architectures in a coherent and consistent way. Most of these enterprise models are related to EM tools (GRAI tools [21], Metis [20], MO2GO, e-MAGIM, etc) [5], [4]. Interoperability problems arise when those organisations aim to achieve enterprise interoperability at a conceptual level. Much effort is spent in European projects [3], [2] 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) [12] and its metamodel Meta Object Facility (MOF) [13]. 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 [8] 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) [18] and Omondo [19] 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) [15] 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) [16] and Model Transformation Framework (MTF) [14]. 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*[22] to represent in a common way enterprise models. The Unified Enterprise Modelling Language (UEML) [1] 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 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. This PIM for SOA metamodel describes four important aspects: services, processes, information and non functional aspects. 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. In this figure business layer is represented separately from technical layer giving evidence of the gap. 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.

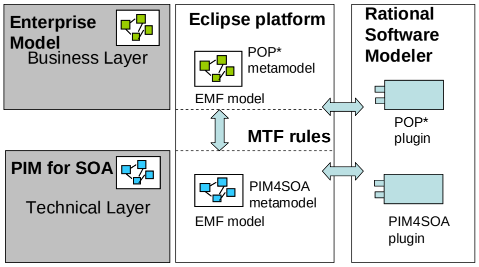


Figure 1. Model driven framework for enterprise models approach

3.1. Technical grounding

In this section we describe the technical grounding of our framework based on the Eclipse platform.

The Eclipse Modelling Framework [8] is used to define the metamodels involved in our framework: POP* metamodel and PIM for SOA metamodel. POP* metamodel is used to define enterprise models and PIM for SOA metamodel to define SOA solutions in a platform independent way.

Within Eclipse framework metamodels are described as ".ecore" models. These models are defined in terms of EMOF specification. This kind of models defines the main elements involved in a specific domain and their relationships. Therefore it allows the instantiation of models compliants with a specific metamodel. For example "pim4soa.ecore" file defines the PIM for SOA metamodel. This metamodel embraces the representation of some important technical aspects and organisational dimensions: service, process and information. One instantiation of this metamodel represents a service oriented solution in a platform independent way.

However the usage of this metamodel is an intricate task for stakeholders. The complexity of the development of models compliant with the metamodel is an arduous task using the EMF editor. This EMF editor is a simple tree representing the metamodel.

Rational Software Modeler is used to implement the UML profiles: POP* UML profile and PIM4SOA UML profile. Two plug-ins (POP* plug-in and PIM4SOA plug-in) are developed in order to customize the Rational Software Modeler environment and facilitate the usage of the profile and the description of services, processes and information aspects. Others UML tools based on Eclipse platform could be used instead of Rational Software Modeler (Omondo).

In this context Model Transformation Framework (MTF) [14] is used as a basis for checking the consistency between the different layers and the different models involved. Therefore, this framework is used to transform and to bridge the gap between POP* models and PIM for SOA models. MTF is a plug-in for the Eclipse platform and thus is integrated in the environment. Each transformation is based on a Relation Definition Language defined in a ".rdl" file. This language defines relations between metamodel elements of different metamodels and it reconciles the models involved MTF is 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.

4. CASE STUDY

This case study is based on the "change management process" presented within the ATHENA project. This process relates the European Aeronautic Defence and Space Company (EADS) and a Landing Gear Provider (LGP). The purpose of this case study is to show an example applying the concepts introduced in previous sections to proof the developed model driven approach. Therefore the model driven framework based on the Eclipse platform is used.

This case study is divided in two different sections. Firstly, a set of problems that arise when EADS and a LGP want to collaborate, are identified. Secondly we describe the application of the framework and finally we briefly discuss the results produced.

4.1. Identified problems

This scenario starts from the premise that EADS and a LGP want to collaborate. However they realise the following set of barriers and problems that they need to avoid and alleviate.

- Their business processes are not defined using the same language. This barrier makes difficult the definition of a coherent and consistent process where the stakeholders have a common and unified view of the process.
- Their systems are not interoperable. They use proprietary format for their applications and their connections are made ad-hoc.
- The functional extensibility of their applications is limited
- Their business processes and their systems supporting their business processes are not related in a systematic way.

4.2. A methodology to apply the framework

Firstly, the business process for the “change management process” is described using the POP* UML profile. To perform this task we use the POP* plugin. This plugin customizes the Rational Software Modeler to define appropriately the business model. The main intention of this model is to represent in UML-like diagrams the business information and the elements involved in the POP* metamodel. This model gets rid of technical details and it focus on business issues. In fact, the elements defining the model are business elements and the business expert can define and handle these elements instead of using technical concepts. Next picture (Figure 2) describes the process where EADS is related to the processes: “LoginToPLMServer”, “CreateChangeRequest” and “ReviewSuppliersSolutionPorposals”. LGP plays a role with the following processes: “ProposeSolution”, “ReviewSuppliersSolutionProposal” and “AcceptTechnicalSolution”.

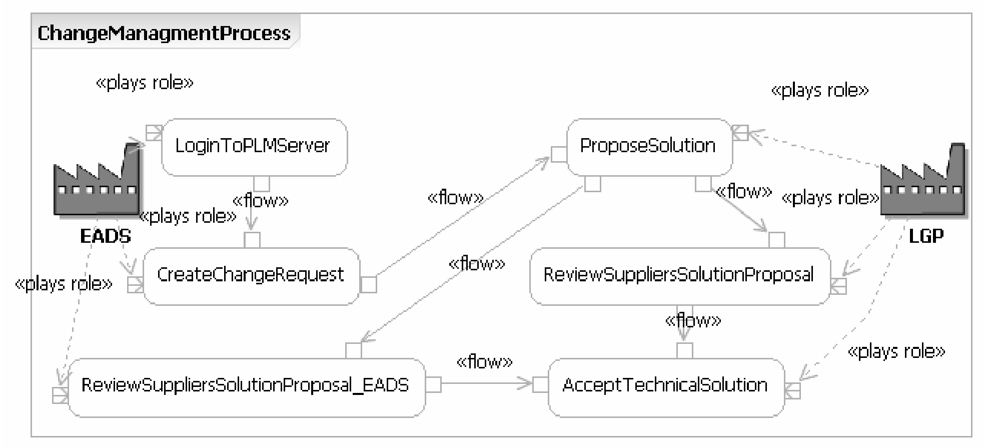


Figure 2. Change management process between EADS and Landing Gear Provider

Once the business model is described a model driven transformation is applied to generate a PIM for SOA model. This transformation only provides a PIM for SOA skeleton and we need to enrich this model with more technical details covering the four important aspects: service, information, process and quality of service. Therefore this model is more focused on the technical solution but keeping independent of the platform used. Another model transformation is used to transform this initial PIM for SOA model into a UML-like model compliant with the profile defined. The user defines as collaborations the identified services and he needs to describe the roles involved, the collaborations used and the processes. Next picture shows a chunk of “ProposeASolution” description using the PIM for SOA UML plugin

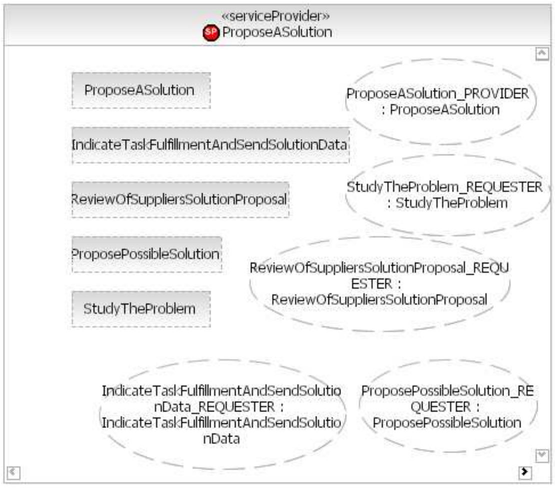
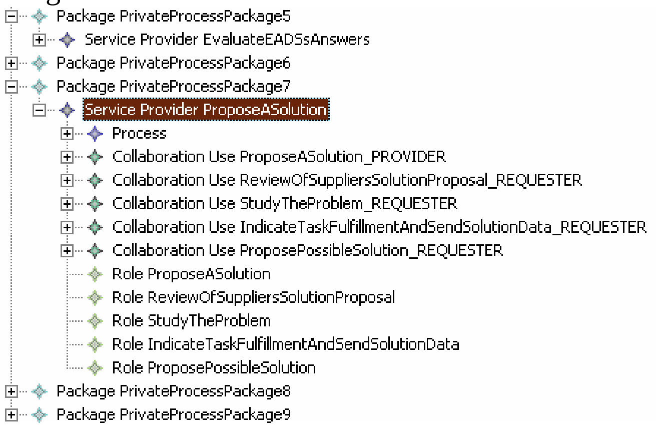

«serviceProvider»
ProposeASolution
ProposeASolution
ProposeASolution_PROVIDER : ProposeASolution
IndicateTask:FulfillmentAndSendSolutionData
ReviewOfSuppliersSolutionProposal
StudyTheProblem_REQUESTER : StudyTheProblem
ProposePossibleSolution
ReviewOfSuppliersSolutionProposal_REQUESTER : ReviewOfSuppliersSolutionProposal
StudyTheProblem
IndicateTask:FulfillmentAndSendSolutionData_REQUESTER : IndicateTask:FulfillmentAndSendSolutionData
ProposePossibleSolution_REQUESTER : ProposePossibleSolution

Figure 3. PIM for SOA UML model

The UML models provide an interface to instantiate PIM for SOA models. Therefore we transform this UML model into a PIM for SOA model. The following picture provides an overview of the resulting model.



```
Package PrivateProcessPackage5
    Service Provider EvaluateEADSsAnswers
Package PrivateProcessPackage6
Package PrivateProcessPackage7
    Service Provider ProposeASolution
        Process
        Collaboration Use ProposeASolution_PROVIDER
        Collaboration Use ReviewOfSuppliersSolutionProposal_REQUESTER
        Collaboration Use StudyTheProblem_REQUESTER
        Collaboration Use IndicateTaskFulfillmentAndSendSolutionData_REQUESTER
        Collaboration Use ProposePossibleSolution_REQUESTER
        Role ProposeASolution
        Role ReviewOfSuppliersSolutionProposal
        Role StudyTheProblem
        Role IndicateTaskFulfillmentAndSendSolutionData
        Role ProposePossibleSolution
Package PrivateProcessPackage8
Package PrivateProcessPackage9
```

Figure 4. A PIM for SOA model

Figure 4 represents Figure 3 showing only the "ProposeASolution" element. From this resulting model we generate platform specific assets as XSD (XML Schema Document).

4.3. Results

In this section we discuss the results from applying the framework to alleviate the problems identified.

- *Their business processes are not defined using the same language*. EADS and a LGP use the same language to describe and to exchange their business processes. POP* provides the way to interchange business processes without loosing information. There exist other approaches but they do not ensure a consistent format. All these approaches are explained in [3]. For example EUML [4] provides the facility to interchange business processes from one language into another through interfaces. However not all elements have a representation from one language into another and therefore this information is lost.
- *Their systems are not interoperable*. Now, their systems are transformed into SOA designs that are able to derive platform specific assets (e.g. WSDL) from PIM4SOA models.
- *The functional extensibility of their applications is limited*. The SOA design allows users to define and increase the functionality of their systems.
- *Their business processes and their systems supporting their business processes are not related in a systematic way*. The framework provides the mechanisms to derive in a systematic way PIM4SOA models from business processes represented in POP*.

5. CONCLUSIONS

In this paper a MDA approach is applied to an e-business environment. This approach addresses the gap between business models and ICT systems implementations and to build service oriented solutions from a platform independent point of view. This is one of the major benefits of this approach. This framework is based on the Eclipse platform and 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.

The presented technology infrastructure represents a first implementation of the approach. In spite of the EMF editor, this approach depends on UML tools to visualize graphically the different models. However the core of this framework, POP* metamodel and PIM for SOA metamodel, is independent from UML tools. As result of this research, an open initiative is under development on sourceforge in order to improve the mentioned PIM for SOA [23]. One of the future initiatives is to provide and to share the complete approach to the open community. Another research direction is to derive from the PIM for SOA directly BPEL and WSDL code.

REFERENCES

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

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

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

[4] UEML D1.1, Report on the State of the art in Enterprise Modelling http://www.euml.org

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

[6] Zachman framework, http://www.zifa.com

[7] TOGAF, The OPEN Group Architecture Framework, http://www.opengroup.org/

[8] Eclipse EMF, Eclipse Modelling framework,http://www.eclipse.org/emf

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

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

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

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

[13] Object Management Group, "Meta Object Facility", Object Management Group (OMG), Document ptc/03-10-04, 2003. http://www.omg.org/docs/ptc/03-10-04.pdf

[14] Model Transformation Framework, IBM. http://www.alphaworks.ibm.com/tech/mtf

[15] QVT-Merge Group, "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

[16] Atlas Transformation Language.http://www.sciences.univ-nantes.fr/lina/atl/

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

[18] Rational Software Modeler. IBM. http://www-306.ibm.com/software/awdtools/modeler/swmodeler/

[19] Omondo. http://www.omondo.com/

[20] Metis. http://www.metis.no/products/metiseditor.html

[21] Graisoft. http://www.graisoft.com/

[22] ATHENA DA1.3.1. “Report on Methodology description and guidelines definition” http://www.athena-ip.org. or http://bpmn.org/Documents/FTF/DA131-090.doc

[23] PIM4SOA sourceforge. https://sourceforge.net/projects/pim4soa