An Approach of Ontology Oriented SPEM Models Validation

Miroslav Líška

Faculty of Informatics and Information Technologies, Slovak University of Technology
Ilkovičova 3, Bratislava, Slovakia

Abstract. SPEM 2.0 is used to define software and systems development processes and their components. It separates reusable development knowledge base from its application in processes. In this work we present a knowledge oriented approach that supports design of SPEM models with respect of SPEM architecture. The main idea is based on creation three ontologies that are a SPEM ontology, a development knowledge ontology and an ontology of development knowledge application. Ontologies are generated with XSL transformations from UML models and also from SPEM CMOF specification and they are imported into the Protégé for reasoning purposes. At the conclusion paper discusses a future work to increase contributions of this approach.

1 Introduction

This paper presents our latest research in application of the Semantic Web technologies in the domain of software engineering based on the Model Driven Architecture. We have chosen the Web Ontology Language as the Semantic Web technology and the Software Process Engineering Meta-Model as the software engineering technology. OWL is a language for defining and instantiating Web ontologies [1]. SPEM 2.0 is used to define software and systems development processes and their components [2].

Our general goal was to link together these two technologies. OWL is standard of the World Wide Web Consortium that is the main international standards organization for the World Wide Web. SPEM is standard of the Object Management Group that is leading not-for-profit standards consortium that produces and maintains computer industry specifications based on the Model Driven Architecture. So it is possible to assume that linking of these technologies could be perspective.

Our concrete goal of this research was to create technique that supports validation of used company's development knowledge base in its real software processes. In order to accomplish these goals we had to ontologize SPEM, a development knowledge base and also its configuration in a process.

Líška M. (2009).
An Approach of Ontology Oriented SPEM Models Validation.
In *Proceedings of the 1st International Workshop on Future Trends of Model-Driven Development*, pages 40-43
DOI: 10.5220/0002176000400043

1.1 MDA and the Semantic Web

At present, it is missing a straightforward linking between OWL and SPEM. But this fact does not mean that OWL (Semantic Web) and SPEM (Model Driven Architecture) are not related. First they were connected through RDF and UML [3, 4]. One issue that has been addressed was the problem that RDF properties are first class entities and they are not defined relative to a class [5]. This difference has also been transferred into OWL and UML difference [6]. Moreover RDFS as a schema layer language has a non-standard and non-fixed-layer meta-modeling architecture, which makes some elements in model, have dual roles in the RDFS specification [7]. Today the main bridge that connects the Semantic Web with the Model Driven Architecture is stated in the Ontology Definition Meta-Model [8]. ODM defines the OWL Meta-Model specified in MOF (MOF – OWL mapping) and also the UML Profile for Ontology modeling (UML – OWL mapping). This architecture can be extended with additional mappings between the UML Profile for OWL and other UML Profiles for custom domains. This creates a bridge from the Semantic Web into custom domain of the Model Driven Architecture [9]. In our approach we have also used this kind of bridge to connect SPEM with the Semantic Web. We have generated a SPEM ontology from the SPEM meta-model (specified in CMOF) that has stated our required mapping between SPEM and UML.

2 Solution

The SPEM 2.0 Meta-Model is a MOF-based model that reuses UML 2 Infrastructure library (e.g. Class, Association) [10]. Its own extended elements are structured into seven main meta-model packages. These are Core, Process Structure, Process Behavior, Managed Content, Method Content, Process with Methods and Method Plugin package. In order to achieve our defined goals we have worked with Method Content package and Process with Methods package. The Method Content meta-model package provides the concepts for SPEM 2.0 users and organizations to build up a development knowledge base that is independent of any specific processes and development projects. The Process with Methods meta-model package defines new and redefines existing structures for integrating processes defined with Process Structure meta-model package concepts with instances of Method Content meta-model package concepts.

At first, we have generated (using XSL) a SPEM Ontology from SPEM CMOF specification in order to obtain all SPEM elements with their relationship. Moreover it was also necessary to extend the SPEM Ontology with SPEM UML2 profile in order to relate SPEM elements with their UML representations. Note this has created possibility to reason whether created SPEM UML models are correctly represented with the SPEM UML Profile. At next it was necessary to create UML Model that was representing a company's development knowledge. We have focused on simple Task Definition "Prioritize Use Cases" that is performed by "Project Manager", "System Analyst" and "Software Architect" Roles Definition. Mandatory output (Work Products Definition) from this task is "Use Case Model" and optional outputs from this task are “Risk List” and “Software Architecture Document”. We have created this model with Sparxsystems Enterprise Architect and transformed it (with XSL) into the Company1 Method Content ontology. This ontology has become a simple development knowledge base. If a new project is going to reuse “Prioritize Use Case” Task Definition, its use must be according to this ontology. For example, “Prioritize Use Cases” Task Use must have as a mandatory output “Use Case Model” Work Product Use. Other outputs are not required, because they are only optional. So we have created another UML model that represents use of Method Content in a process and transformed it (with XSL) into the Company1 Method Content Use ontology. To relate Method Content Use elements with Method Content elements it is necessary to use contentTrace relationship as is depicted in Fig. 1. Note in our approach we use contentTrace relationship equally as Instance relationship. For example, “Project Manager” Role Definition in Method Content is a class, where “Project Manager” Role Use is its instance.

3 Ontology Reasoning

At last we have imported the SPEM ontology, the Company1 Method Content Ontology and Company1 Method Content Use Ontology into the Protégé. The Protégé is a free, open source ontology editor and knowledge-base framework [11]. Note that it is important to share the same namespace (spem2) in every ontologies in order to the Protégé can properly recognize elements from the Company1 Method Content Use Ontology as individuals of the Company1 Method Content ontology that is specialization of the SPEM Ontology.

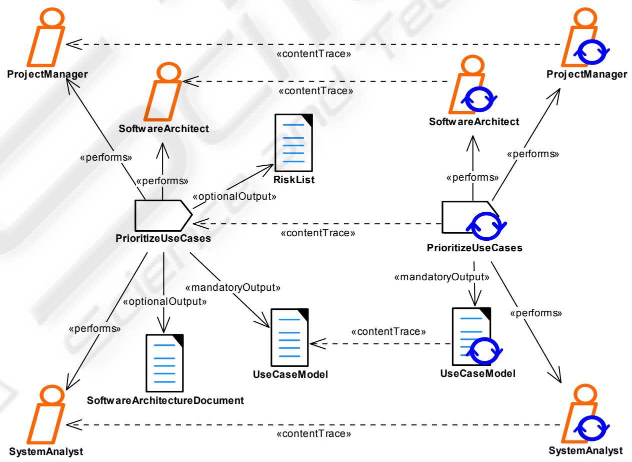

Fig. 1. Method Content (left) and Method Content Use (right).

After that we have executed reasoning to find out whether these ontologies are consistent. The Protégé resulted that ontologies are consistent what we have expected because they were not contradictional. In order to prove that our approach is able to disallow inconsistent use of the Method Content in a process, we have stated, that mandatory output from "Prioritize Use Cases" Task Use is "Risk List" Work Product Use instead of required "Use Case Model" Work Product Use. The Protégé has resulted that this was inconsistent, because the object property mandatoryOutput requires as a domain individuals of "Prioritize Use Cases" Task Definition and as range it requires individuals of "Use Case Model" Work Product Definition. Note that Work Product Definition "Risk List" and Work Product Definition "Use Case Model" must be disjointed classes.

4 Conclusions

In this paper we have presented our latest research based on knowledge engineering in MDE. A lot of work must be done and it is fair to say, that at the moment we have only started to bridge SPEM with UML. At present, our approach cannot be used without limitations. For example, necessary condition that every Method Content and Method Content Use model must have at least one responsible relationship, one performs relationship and one mandatory output relationship must be satisfied. If not, than the reasoner cannot find inconsistency what is required.

Acknowledgements

This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0508/09.

References

1. Smith, M.K., Welty, Ch., McGuinness, D.L.: OWL Web Ontology Language Guide. W3C Recommendation, (2004)
2. OMG: Software and Systems Process Engineering Metamodel. formal/2008-04-01, (2008)
3. Manola, F., Miller, E.: RDF Primer. W3C Recommendation, (2004)
4. OMG: UML Superstructure. ptc/2008-05-05, (2008)
5. Cranefield, S.: Networked Knowledge Representation and Exchange using UML and RDF. Journal of Digital Information, Volume 1 Issue 8, (2001)
6. OMG: OWL Full and UML 2.0 Compared. www.omg.org
7. Pan, J., Horrocks, I.: Metamodeling Architecture of Web Ontology Languages, In Proceedings of the First Semantic Web Working Symposium, Stanford, (2001)
8. OMG: Ontology Definition Metamodel 1.0. ptc/2008-09-07 (Beta 3), (2008)
9. Gaševic, D., Djuric, D., Devedžic, V.: MDA and Ontology Development. Springer, Berlin, (2006)
10. OMG: UML Infrastructure. ptc/2008-05-04, (2008)
11. Rector, P., Drummond, N., Horridge, M., Wang, H., Dameron, O.: Advanced Reasoning with OWL. In 9th International Protégé Conference, Stanford, (2006)