CREATING A META-MODEL FOR SEMANTIC WEB SERVICE STANDARDS

Florian Lautenbacher and Bernhard Bauer
Programming Distributed Systems Lab, University of Augsburg, Universitaetsstr. 14, Augsburg, Germany

Keywords: Semantic Web Services, OWL-S, WSDL-S, WSMO, SWSF, UML-profile, ODM, MDA.

Abstract: Annotating web services with semantic information is a tremendous piece of work for persons who are not familiar with the languages and underlying logic. However, a graphical model can assist users to achieve this goal easily. Therefore, we reviewed existing standards and developed a meta-model and UML profile for semantic web services based on the current W3C submissions to enable an automatic generation of code and reduce this normally time-consuming task.

1 INTRODUCTION

Service-oriented architectures (SOA) have gained a lot of attention over the last years. As the number of available web services is steadily increasing, companies realize the need for automatically discovering web services and having an automated composition. Several organizations have already proposed web service composition languages (like WSBPEL (Alves et al., 2006) or XPDL (WfMC, 2005)), but these standards lack a semantic description of services which can be automatically interpreted by machines and can not be used to make an automatic discovery or composition. But, semantic interoperability of web services is needed. To solve these issues several organizations and companies focus currently on the development of semantic-enabled web service technologies (or buy companies which have knowledge in this area). Several approaches have been submitted to the World Wide Web Consortium (W3C) – each one focusing on different aspects as the other. As the number of "standards" is getting higher, the need for an independent way of describing semantic web services increases, too, because the learning curve for the average service developer can be steep for such languages and the creation of ontologies (especially for web services) is a huge amount of work.

The Object Management Group (OMG) promotes the model-driven architecture, shortly MDA (OMG, 2003), approach towards the analysis, design and implementation of systems. One of the key aspects is the usage of the Unified Modeling Language UML (OMG, 2005) to model all kinds of aspects. UML (currently version 2.1 is under development) is established in the section of computer science, but also easily understandable for business experts (e.g. business process models as activity diagrams, see e.g. Lautenbacher, F. and Bauer, B., 2006) and supports with its well-defined meta-model (building on MOF, OMG, 2006a) model transformation and code generation.

To fulfil the need of an independent way of describing semantic web services we developed a meta-model and UML-profile for semantic web services. Additionally, we specified informal transformation rules to generate code from the meta-model and implemented these rules using the openArchitectureWare-language XPand (Efftinge, C. and Kadura, C., 2006). Our UML-profile has been implemented in a UML tool-suite (innovatorAOX 2006 from MID Enterprise Software Solutions GmbH, Nuremberg) and a well-known example has been modeled as a case study.

This paper is organized as follows: In the next section we compare already existing standards (of semantic web services and others) which we use for our meta-model and give a short explanation about each standard. We introduce our meta-model and explain each concept in section 3. Related work, other profiles and their shortcomings are described in section 4, before we conclude in the last section where we summarize the contributions of this approach and describe future work.

Lautenbacher F. and Bauer B. (2007).
CREATING A META-MODEL FOR SEMANTIC WEB SERVICE STANDARDS.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Web Interfaces and Applications*, pages 376-381
DOI: 10.5220/0001266203760381

2 A SURVEY OF EXISTING STANDARDS

This chapter gives a short introduction to the current semantic web service (SWS) submissions and other approaches and compares them.

2.1 OWL-S

OWL-S (Martin, D. et al., 2004) (currently version 1.2 is under development) enables the discovery, invocation, interoperation, composition and verification of services. It builds on the formerly developed DAML-S and was the first submission of a semantic web service to the W3C.

Each Semantic Web Service in OWL-S consists of a service profile, a service model and grounding. The service profile describes what the service does and is used to advertise the service. The service model answers the question "how is it used?" and describes how the service works internally. Finally, the service grounding specifies how to access the service.

To give a detailed perspective on how to interact with a service, it can be viewed as a process. OWL-S distinguishes three kinds of processes: *Atomic processes* are directly callable and correspond to the actions a service can perform by engaging it in a single interaction; *Composite processes* correspond to actions that require multi-step protocols and/or multiple server actions; finally, *Simple processes* are not callable and not associated with a grounding and only provide an abstraction mechanism to enable multiple views of the same process.

OWL-S is based on the Web Ontology Language OWL and supplies web service providers with a core set of markup language constructs for describing the properties and capabilities of their web services in an unambiguous, computer-interpretable form.

2.2 WSMO

The Web Service Modeling Ontology (WSMO) (Lausen, H. et al., 2005) is a formal ontology and language that consists of four different main elements for describing semantic web services:

Ontologies provide the terminology used by other elements to describe the relevant aspects of the domains of discourse.

Goals state the intentions that should be solved by web services and are representations of one or more objectives which need to be fulfilled.

Web Services: A Web Service is a computational entity which is able to achieve a part of or the complete goal a user seeks to fulfil. WSMO web service descriptions describe various aspects of a service and consist of functional, non-functional and the behavioural aspects of a web service.

Mediators resolve interoperability problems and describe elements to overcome incompatibility problems between different elements on data, process and protocol level.

2.3 WSDL-S

WSDL-S (Akkiraju, R. et al., 2005) is another semantic web service approach submitted to the W3C in November 2005 and extends the WSDL standard with semantic information. Currently it only considers single-step services and there is no specification how a composition of several web services should be handled. However, there are already efforts to annotate web service choreography languages like WS-BPEL similar to the annotations made in WSDL-S (see e.g. Pistore, M. et al., 2006). WSDL-S uses the extensibility elements of WSDL and introduces new elements to describe the semantics of a service, of its inputs and outputs and of preconditions and effects.

The advantages of this approach can be summarized as follows:

- it builds on existing web services standards
- it supports the user's choice of the semantic representation language
- it allows the association of multiple annotations
- it supports semantic annotation of web services whose data types are described in XML schema (therefore e.g. GRDDL – see Hassael-Massieux, D. and Connolly, D., 2005 – could be used)

2.4 SWSF

Another W3C submission (from September 2005) is the Semantic Web Services Framework (SWSF) (Battle, S. et al, 2005) which represents an attempt to extend the work of OWL-S and consists of two major parts: the Semantic Web Service Language (*SWSL*) and the ontology *SWSO* above.

SWSO can itself be divided in two formats: *FLOWS*, the first-order logic ontology for web services and *ROWS*, the rules ontology for web services.

SWSF emerged from the work in service composition which might require more expressivity than is available in OWL and is therefore based on logic programming, first-order logic and policy research. It builds on DAML-S, OWL-S and WSMO and provides rich semantics for greater automation of discovery, selection and invocation, content transformation, composition, monitoring and recovery and verification. Laying the focus on messages (similar to WSDL 2.0) it introduces the concepts of Channels and Messages which can be created and modified using several specialized actions.

2.5 Ontology Definition Metamodel

Based on the ideas and work of Stephen Cranefield, Dragan Gašević and others, the Object Management Group (OMG) works on a specification for the interoperability between ontologies (and its underlying description logic), UML diagrams, ER-diagrams, Topic Maps and common logic. The Ontology Definition Metamodel (ODM) (OMG, 2006b) consists of four platform independent models (PIM: Common logic, Topic Maps, RDF and OWL) and informative models like the one for Description logic. It defines a meta-model and UML-profile for RDF(S) and OWL-ontologies and mappings between the meta-model and these semantic web standards.

2.6 Comparison of SWS Approaches

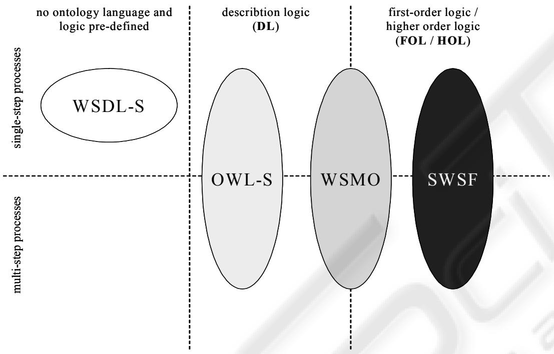

Figure 1: Categorization of Semantic Web Service submissions.

Figure 1 shows how the semantic web service submissions might be categorized. WSDL-S does not force to use a specific ontology language and therefore it does not define any underlying logic. It is only usable for single-step processes whereas the other standards can handle multi-step processes, too. OWL-S builds on OWL which is based on description logic (OWL DL). OWL-S is widely common in projects and research, because it was the first SWS submission. SWSF uses first-order logic and rules languages to model the ontology in its underlying language SWSL. WSMO offers both logic layers for the modeling of ontologies in WSML. SWSF and WSMO offer with their first-order logic and rule support a more comprehensive way of modeling, which might be problematic for reasoning.

3 A META-MODEL FOR SEMANTIC WEB SERVICES

Based on the above described standards and seeing the need for an independent way of modeling semantic web services, we designed a meta-model which can be applied and transformed into all of the mentioned W3C submissions. shows the dependencies from our meta-model to the standards introduced above.

Our semantic web service meta-model is based on the Ontology Definition Metamodel for modeling ontologies (in form of a UML-profile) and on the above mentioned semantic web service submissions. WSDL-S is independent on the underlying ontology modeling language, but OWL-S, WSMO and SWSF require a specific ontology language (OWL, WSML and SWSL).

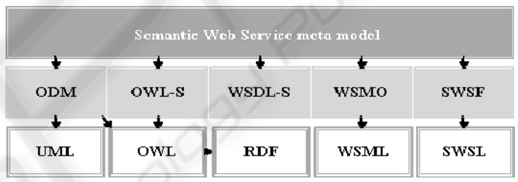

Figure 2: Profile dependencies.

It is difficult to integrate our meta-model into the layers of MDA. It is platform independent in principal, but also includes constructs for each specific platform and semantic web service language and one can directly generate code from the meta-model. But, this matches to our own experiences that business users prefer one meta-model avoiding model transformations where possible and use different views on a meta-model instead. This also conforms to current discussions about the future of the MDA at the OMG where e.g. Stan Hendryx, Chairman of the OMG Business Rules Special Interest Group, pleaded for a weakening of the current layers.

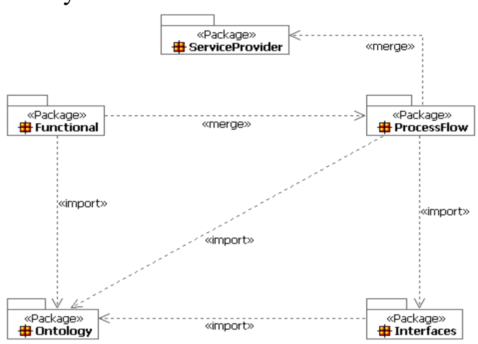

Figure 3: Package overview for our meta-model.

Our profile consists of five packages which interact with each other. The Ontology-package contains all concepts that are needed to model an ontology (similar to ODM). The Interfaces-package provides all elements to model a WSDL service and to describe it with semantics (like in WSDL-S). The ServiceProvider-package includes all aspects to model one or more semantic web services with non-functional descriptions and using the elements of the ProcessFlow-package the functional elements and composition of multi-steps can be modeled. Every single step can be annotated with functional descriptions as described in the Functional-package. The Interfaces-package, the ProcessFlow-package and the Functional-package access elements of the ontology and therefore import the Ontology-package. The ProcessFlow-package extends concepts defined in the ServiceProvider-package and therefore merges this package. The Functional-packages extends some concepts of the ProcessFlow-package and therefore merges this, too.

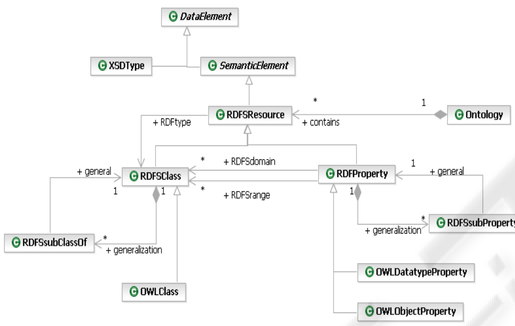

Figure 4: SWS meta-model for the ontology.

At the bottom of our meta-model are elements to model the constructs in an ontology (whereby it doesn't matter whether this ontology needs to be in OWL or in WSML). We are conform to the ODM standard in this package, but since we also need elements which are not semantically described (e.g. in a XSD-file) the *DataElement* (compare) is specialized in two ways: It can be an element specified in an XSD-file or it can be a *SemanticElement*. This might be an *RDFSResource* which itself is then part of an ontology. A resource can either be a class (or concept as it is called in WSML) or a property. *RDFSClass*es can be connected to other classed via *RDFProperties* which contain the domain and range of the property. Every class (resp. property) can be generalized and there exist specializations for OWL classes which are the mostly used presentation form of current ontologies. ODM defines many more constructs, but these are the most basic ones to model an ontology.

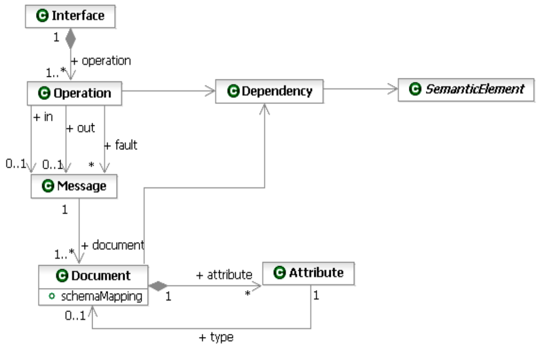

Figure 5: SWS meta-model for interfaces, operations and messages.

Having finished the ontology grounding, the next step would be to develop elements for modeling the infrastructure of the service(s) (). Every web service (as defined in WSDL) has an interface which includes a number of operations. Every operation can have one input and output message and zero or more fault messages. Each operation might be described with a semantic element from the ontology (similar to WSDL-S). A message contains one or more documents which can also be described with semantic elements and which can be structured with several attributes which might themselves be documents again or simple data types (like String, int, etc.).

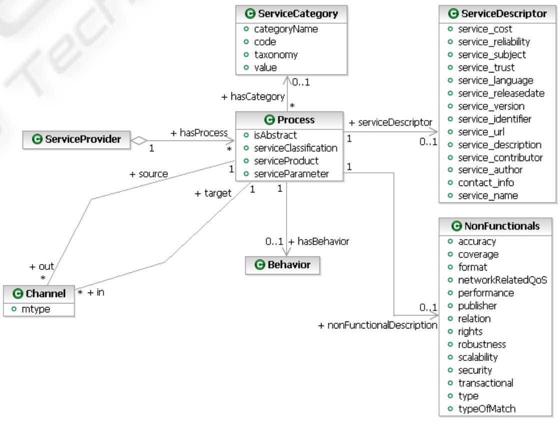

Figure 6: SWS meta-model for a service provider.

With these basic constructs for describing the interfaces, operations, messages and the underlying ontology we can now start the modeling of one or more semantic web services. A web service is a *Process* with a specified *Behavior*. A *ServiceProvider* (an organization or institution) offers a number of processes which can be executed. Every process can be categorized (according to OWL-S and SWSF) with a *categoryName*, a path to a *taxonomy*, a specific *value* in the taxonomy and the *code* associated to a taxonomy. Each process can be additionally described with a *ServiceDescriptor* and *NonFunctionals*. These contain attributes to describe the non-functional properties of a web service. Each process can communicate with other processes via *Channel*s.

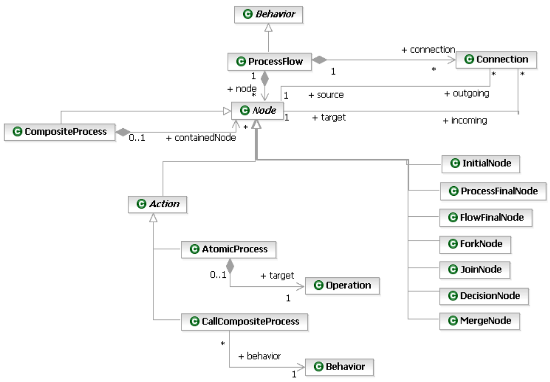

Figure 7: SWS meta-model for the process flow.

Each process has an internal behaviour which can be described using a *ProcessFlow*. A process flow contains *Node*s and *Connection*s between these nodes. A node might either be a control node like the ones known in UML activity diagrams for XOR- and AND-splits and joins and other control flow directions (InitialNode, ForkNode, JoinNode, DecisionNode, MergeNode, etc.) or an *Action* (single-step) or a *CompositeProcess* (multiple steps) which contains several other nodes. An action can be either an *AtomicProcess* which calls an operation or a *CallCompositeProcess* which can start a new behavior or a *CompositeProcess*.

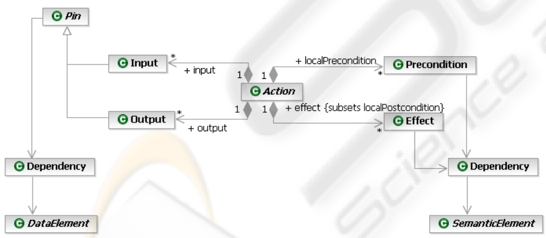

Figure 8: SWS meta-model for the functional description.

Each action has *Input*s and *Output*s which refer to simple types or semantic elements as introduced above. Every action can also be described with its *Precondition*s and *Effect*s, which itself can be described in more detail with a class or concept of the ontology.

We developed a UML-profile based on this meta-model, integrated it into a UML tool suite and modeled the CongoBuy-example which was first introduced with OWL-S. For more details see our technical report on (Lautenbacher, F., 2006).

4 RELATED WORK

There are several efforts to create a UML profile for Semantic Web Services. However, to our knowledge none of the existing approaches tries to consider every existing W3C submission of semantic web services (meaning OWL-S, WSMO, WSDL-S and SWSF).

In (Skogan, D. et al, 2004) and (Gronmo, R. et al, 2005) the authors define transformations between UML and OWL-S and a web service composition based on this information. The developed profile uses the UML Ontology Profile (defined by Duric for UML 1.5 class diagrams) to model the concepts of the ontology. They use a UML activity to describe a web service and to attach inputs and outputs which makes it difficult to use control nodes for the composition of several web services later. Their profile supports the generation of OWL-S and WSMO code (hence, there are no transformation rules for the generation of WSMO), but doesn't consider SWSF and WSDL-S.

In (Timm, J. and Gannod, G., 2005) a model-driven approach for specifying semantic web services has been developed. However, the UML-profile only considered AtomicProcesses in OWL-S, not including the collaboration of several processes. It is only applicable to OWL-S and misses transformation rules for WSMO, SWSF and WSDL-S.

(Scicluna, J. et al., 2004) describes how OWL-S services can be modeled, but in a proprietary format not using the UML-profiling mechanism. (Pondrelli, L., 2005) develops an MDD annotation methodology for semantic enhanced SOAs, but does not develop a UML profile for semantic web services in greater detail. (Acuna, C. and Marcos, E., 2006) describes a case study with a methodological framework for the development of semantic web information systems (MIDAS-S) building on WSMO. In (Kendall, E., 2006) (and other talks) the author promotes the integration of OWL-S and SWSF within the ODM. We completely agree and support this initiative, if the meta-model considers other approaches of semantic web services like WSMO and WSDL-S, too.

5 CONCLUSIONS

Using our meta-model and UML-profile one can simply model a semantic web service and then generate code in one of the currently proposed SWS-languages. Our profile provides independency from each single SWS standard and can easily be adapted in the future. We integrated our profile into a UML CASE-tool and modeled a well-known example.

Our meta-model is compatible with all of the current W3C SWS-submissions and builds on the OMG specification draft, it enables a code generation through a well-defined meta-model and it includes the modeling of ontologies.

However, the integration of semantic web rules is still missing. To make it easier to model preconditions and effects we will include rules (based on SWRL or WRL) in upcoming versions of our profile. We are also interested on semantic business process models and how to combine these business processes with the developed meta-model for semantic web services.

REFERENCES

Acuna, C. and Marcos, E., 2006. Modeling semantic web services – a case study, ICWE06, Palo Alto, CA, USA.

Akkiraju, R. et al., 2005. Web Service Semantics – WSDL-S, W3C Member Submission.

Alves, A. et al., 2006. Web Services Business Process Execution Language Version 2.0, Committee Draft at OASIS.

Battle, S. et al., 2005. Semantic Web Services Framework (SWSF) Overview, W3C Member Submission.

de Bruijn, J., Lausen, H., Polleres, A. and Fensel, D., 2005. WSML – a Language Framework for Semantic Web Services, W3C Workshop on Rule Languages for Interoperability, Washington DC, USA.

Efftinge, S. and Kadura, C., 2006: OpenArchitectureWare 4.1 Xpand Language Reference, available online at http://www.eclipse.org/gmt/oaw/doc/4.1/r20_xPandReference.pdf

Gronmo, R. and Jaeger, M. and Hoff, H., 2005. Transformations between UML and OWL-S, ECMDA-FA, Nuremberg

Hassael-Massieux, D. and Connolly, D, 2005. Gleaning Resource Descriptions from Dialects of Languages (GRDDL)", W3C Team Submission.

Kendall, E., 2006. MDA and Semantic Web Services: Integrating OWL-S & SWSF with the Ontology Definition Metamodel (ODM), SOA, MDA and Web Services Workshop, OMG.

Kopecky, J. and Parsia, B., 2006. Web Service Description Language (WSDL) Version 2.0: RDF Mapping, W3C Working Draft.

Lausen, H., Polleres, A. and Roman, D. (Eds.), 2005. Web Service Modeling Ontology (WSMO), W3C Member Submission.

Lautenbacher, F. and Bauer, B., 2006. Semantic Reference and Business Process Modeling Enables an Automatic Synthesis, Workshop SBPM at ESWC06, Budva.

Lautenbacher, F., 2006. A UML profile and transformation rules for semantic web services, Technical Report 2006-20, Augsburg, Germany, available online at http://www.ds-lab.org/publications/reports/2006-20.html

Martin, D. et al.: 2004. OWL-S: Semantic Markup for Web Services, W3C Member Submission.

Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. INSTICC Press.

OMG, 2003. Object Management Group: MDA Guide Version 1.0.1.

OMG, 2005: Object Management Group: Unified Modeling Language (UML) Specification: Superstructure, Version 2.0, Final Adopted Specification.

OMG, 2006a. Object Management Group: Meta Object Facility (MOF) Core Specification, Version 2.0.

OMG, 2006b: Object Management Group: Ontology Definition Metamodel (ODM), Fifth Revised Submission to OMG/RFP ad/2006-01-01.

Peer, J., 2005. Semantic Service Markup with SESMA, Web Service Semantics Workshop (WSS'05) at the WWW 2005.

Pistore, M., Spalazzi, L. and Traverso, P., 2006. A Minimalist Approach to Semantic Annotations for Web Processes Compositions, ESWC06, Budva.

Pondrelli L., 2005. An MDD annotation methodology for Semantic Enhanced Service Oriented Architectures, Open Interop Workshop on Enterprise Modelling and Ontologies, Porto.

Scicluna, J., Abela, C. and Montebello, M., 2004. "Visual Modeling of OWL-S Services", CSAW'04 at the University of Malta.

Skogan, D. and Gronmo, R. and Solheim, I., 2004. Web Service Composition in UML, EDOC, Monterey.

Smith, J., 1998. *The book*, The publishing company. London, 2nd edition.

Timm, J. and Gannod, G., 2005. A Model-Driven Approach for Specifying Semantic Web Services, IEEE ICWS 2005.

WfMC, 2005. XML Process Definition Language, WFMC-TC-1025, Workflow Management Coalition Specification.