D11v0.2 WSMO-Lite

WSMO Working Draft 21 April 2007

This version:
http://www.wsmo.org/TR/d11/v0.2/20070421/
Latest version:
http://www.wsmo.org/TR/d11/v0.2
Previous version:
http://www.wsmo.org/TR/d11/v0.2/20070421/
Editors:
Jacek Kopecký
Tomas Vitvar

Table of contents

---

1. Introduction

The WSMO conceptual framework [Roman et al. 2006] defines a complete, top-down model of semantic description of Web services. It is realized in the WSML family of languages. Both the WSMO model and the WSML languages are engineered from the requirements, without significant reuse of Web standards.

The WSMO model is independent from any specific implementation and communication technologies. This independence makes it able to work with WSDL and Triple Spaces as two competing underlying systems. However, it seems likely that support for Triple Spaces can be included in WSDL (which has support for HTTP, with some very similar properties as spaces). Therefore, in this deliverable we propose an alternative to WSML which realizes the WSMO model on top of WSDL, not independent from it.

In the WSDL model [WSDL], a Web service has a single interface, a number of enpoints with one binding each. In the WSMO model, a Web service has a single capability and a number of interfaces with choreographies and orchestrations. A choreography in the WSMO model needs to be grounded in a data transport, e.g. WSDL (see WSMO D24.2).

In this very first draft of this deliverable, we focus on analyzing how we can fit the information from WSMO into the WSDL description model. The mechanism for attaching semantic information from WSMO on the WSDL components is the modelReference attribute from Semantic Annotations for WSDL and XML Schema [SAWSDL]. Future versions will propose an RDFS ontology that captures the necessary WSMO information in RDF data.

2. WSMO data in the WSDL model

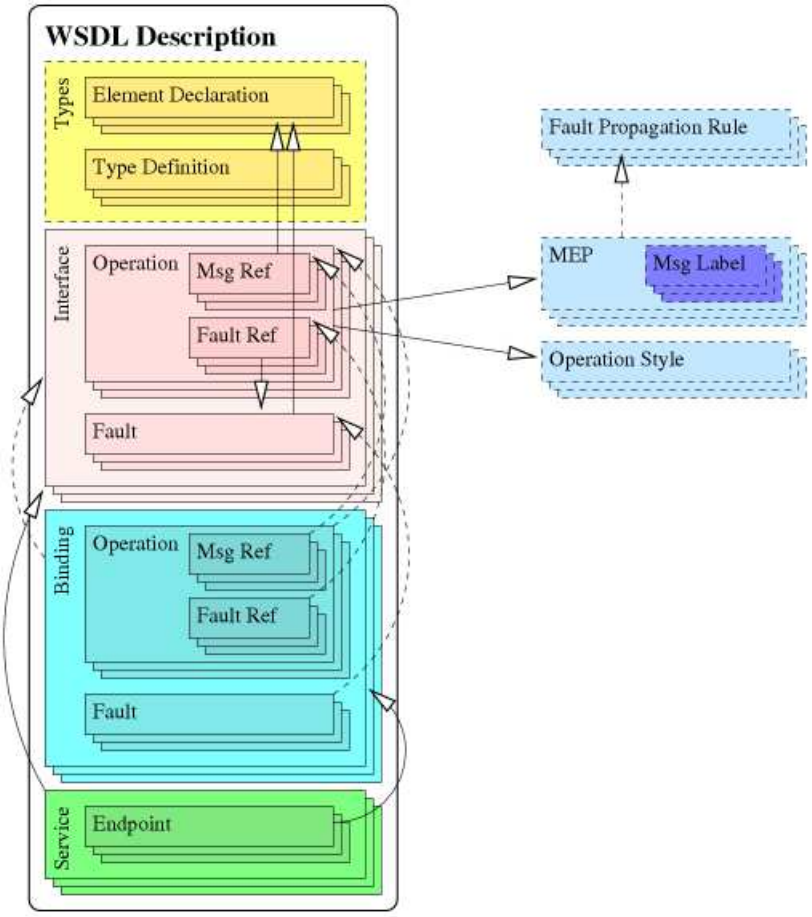
The Web Services Description Language WSDL has the following basic model of components:

1. A **Service** has a single interface and a number of **Endpoints** which do not differ in functionality, instead they may present different data transmission constraints and capabilities, which are specified as bindings.
2. An **Interface** groups the **Operations** that are available at the Web service. For WSDL, an operation is a simple message exchange, and there is a number of possible message exchange patterns in the WSDL specification. A WSDL interface may be reusable, i.e., multiple Web services may offer the same interface; such reuse may be the basis for a very simple service discovery and substitution.
3. A **Binding** describes the networking details necessary for a client to communicate with a Web service endpoint.

Figure 1 shows the structure of WSDL graphically and in some more detail. The dashed links represent implied or

optional references, and the dashed boxes represent WSDL entities that are not actual components of WSDL description, but are useful to demonstrate the structure. Each link has a 1:1 cardinality. The figure also shows the top-level components that are reusable between descriptions.

Figure 1: The structure of WSDL



The Web Service Modeling Ontology WSMO has the following basic model of components:

1. A **Web Service** has a single capability describing its functionality, and any number of interfaces, describing its interactions.
2. A **Capability** describes the functionality of a Web service by specifying its preconditions, assumptions, postconditions and effects. Capability descriptions are used for Web service discovery based on functional properties.
3. An **Interface** describes how the functionality of the Web service is achieved in terms of interactions between the client, the Web service and other Web services. The interaction between the client and the Web service is described with a **Choreography**, whereas the interaction between the Web service and other Web services that it might use is described with an **Orchestration**.

WSDL descriptions are intended to describe the Web service to a client so that the client is able to make use of the service. From this point of view, orchestration may only be important indirectly, and it is out of scope of WSDL even to reference it. Therefore we do not deal with orchestrations in this deliverable.

The two components of WSMO that deal with describing the service for the benefit of the client are therefore capability and choreography. In WSMO, both belong the the Web service top-level concept. Therefore the first set of links in WSMO-Lite are from WSDL service to a capability description, and a choreography description.

Since a WSDL interface is a top-level, reusable description, the interface itself can point to a capability description and to a choreography description, which would apply to all services that implement the interface. The capability or choreography description on a service could then be seen as restricting, extending, or overriding the descriptions from the interface; deciding which approach is best is material for further research.

As a WSDL service or interface annotation, a semantic choreography is a means of telling the client which operation it should invoke at any given time. A pre-specified choreography is not the only option, however; the client may be able to use AI Planning techniques to choose the order of operation invocations, assuming the operations are sufficiently described with their atomic capabilities. Therefore, also WSDL operations should be allowed to point to a capability description. We cannot, at this time, give a definitive statement on whether describing the choreography as a whole is more or less complex than describing the capabilities of the operations, this is also material for further research.

To summarize, we identified the following points in WSDL that should be annotated with either capabilities or choreographies (as modeled in WSMO):

WSDL component	Semantic annotation
Interface	Capability
Interface	Choreography
Interface Operation	Capability
Service	Capability
Service	Choreography

A WSDL document with these annotations should effectively be able to serve the same purpose as a WSML semantic Web service description without orchestrations. In fact, it may be possible to create a tool that reconstructs a WSML description from a WSDL document annotated with WSMO-Lite, or that annotates a WSDL document based on a WSML Web service description with grounding.

3. Conclusions and future work

In this very first draft of this document, we focus on analyzing how WSMO data may be applied to WSDL descriptions (using SAWSDL annotations), showing that the two relevant WSMO components, capability and choreography, can be applied to various WSDL components depending on how abstract or concrete they are. A further version of this deliverable will provide an RDFS ontology for WSMO capabilities and choreographies and it will describe how WSDL files can be annotated with these descriptions.

We need to further investigate the relationships between the semantic annotations on a service, on its interface and on the operations of the interface.

Additionally, we need to investigate whether describing the choreography of a WSDL interface is more or less complex than describing the capabilities of the operations and using AI Planning in lieu of following a choreography. At this time, we expect that different situations will favor one or the other technique, therefore we allow both interface choreography annotations and operation capability annotations.

Finally, we may want to investigate whether an automated tool can translate between WSMO-Lite-annotated WSDL documents and WSML Web service descriptions.

References

[Roman et al. 2006] D. Roman, H. Lausen, U. Keller (editors): *Web Service Modeling Ontology (WSMO)*, version 1.3 available at http://www.wsmo.org/TR/d2/v1.3/

[SAWSDL] J. Farrell, H. Lausen (editors): *Semantic Annotations for WSDL and XML Schema*, W3C Candidate Recommendation, 26 January 2007, available at http://www.w3.org/TR/2007/CR-sawsdl-20070126/

[WSDL] R. Chinnici, J-J. Moreau, A. Ryman, S. Weerawarana (editors): *Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language*, W3C Working Draft, 26 March 2007, available at http://www.w3.org/TR/wsdl20/

Acknowledgements

The work is funded by the European Commission under the projects ASG, EASAIER, enIRaF, Knowledge Web, Musing, Salero, Seemp, SemanticGOV, Super, SWING and TripCom; by Science Foundation Ireland under the DERI-Lion Grant No.SFI/02/CE1/I13; by the FIT-IT (Forschung, Innovation, Technologie - Informationstechnologie) under the projects Grisino, RW², SemBiz, SeNSE and TSC.

The editors would like to thank to all the members of the WSMO working group for their advice and input to this document.

$Date: 2007/04/21 18:33:14 $

webmaster