ODESWS Framework

Asunción Gómez-Pérez, Rafael González-Cabero
Departamento de Inteligencia Artificial, Facultad de Informática,
Campus de Monteganςedo s/n, Universidad Politécnica de Madrid,
28660 Boadilla del Monte, Madrid, Spain.
E-mail: asun@fi.upm.es
E-mail: rgonza@fi.upm.es

Manuel Lama
Departamento de Electrónica e Computación, Facultad de Física,
Campus Universitario Sur s/n, Univesidade de Santiago de Compostela,
15782 Santiago de Compostela, A Coruña, Spain.
E-mail: lama@dec.usc.es

Abstract. Web Services are interfaces to a collection of operations that are network-accessible through standardized XML messaging, and whose features are described using standard XML-based languages. Semantic Web Services (SWS) describe semantically the internal structure and the functional/non-functional capabilities of the services, facilitating the design and evaluation of SWS based on that semantic description of the features of the services. To enable the design and composition SWS at the knowledge level, the ODESWS framework has been proposed. That framework uses problem-solving methods to describe the functional and structural features of the SWS

Introduction

Web Services (WSs) are interfaces that describe a collection of operations that are network-accessible through standardized Web protocols, and whose features are de-scribed using a standard XML-based language (Kreger, 2001; Cubera et al., 2001). Although there are other definitions of what a WS is, (refer to (Alonso, 2003) for an enumeration of them), we believe that this definition captures better what a WS is (and from where its benefits come from). In few words, "It's not the components, it's the interfaces" (Kayne,2003) .

The main web services (Kreger, 2001; Cubera et al., 2001) features are: (1) communication features that describe the protocols required to invoke the service execution; (2) descriptive features that detail the e-commerce properties; (3) functional features that specify the capabilities, enabling thus an external invoking agent to determine whether the service execution can obtain the requested results; and (4) structural features that describe the internal structure of a composite service, that is, which are its structural components and how these components are combined among them to execute the service.

In this context, the Semantic Web has risen as an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation (Berners-Lee et al., 2001). Following this approach, Web Services in the Semantic Web, the so-called Semantic Web Services (SWS), will be the mark-up of WS to make them computer-interpretable, use-apparent and agent-ready (McIlraith et al., 2001). This mark-up is the semantic description of the WS and it will facilitate external agents to understand both the functionality and the internal structure of the services.

The motivating tasks of SWS are to be able to discover, compose, and invoke automatically SWS (Hendler, 2001). Several approaches have appeared with this aim, being the more relevant the OWL-S specification, IRS-II and WSMO.

The OWL-S (OWL Services Coalition, 2004) specification (formerly DAML-S (Ankolenkar et al., 2002)) has been proposed to describe services in a semantic manner, using OWL (Dean and Schreiber, 2004) in combination with the WSDL language (Christensen et al., 2001) and SOAP (Box et al., 2000), achieving thus the desirable combination of WS standard languages and semantic annotation in order to use the current infra-structure of the WS (Sollazo et al., 2002).

The Internet Reasoning Service IRS-II (Motta et al., 2003) is a SWS framework, which allows applications to describe semantically and execute SWS. It uses problem-solving methods (PSM) to represent SWS, attaching to each WS a PSM that describes it. A PSM is an abstract implementation of a domain-independent description of reasoning processes which can be applied to solve tasks in a specific domain. It is based on the UPML (Unified Problem Solving Method description Language) frame-work (Fensel et al., 2003).

The Web Service Modelling Framework (WSMF) (Fensel and Bussler, 2002) tries to provide a model for describing the various components in an e-commerce environment. WSMF is the result of research carried out on modelling of reusable knowledge components and its core are two complementary principles: a strong de-coupling of the aforementioned components that carry out an e-commerce application; and a strong mediation between these elements. Mediation is applied at several levels: mediation of data structures; mediation of business logics; mediation of message exchange protocols; and mediation of dynamic service invocation. WSMF consists of four main elements: ontologies, goals, web services and mediators. All these elements are described using the Web Service Modelling Ontology (WSMO) (Roman et al., 2004). The underlying representation language for WSMO is F-logic (Kifer et al., 1995), a full first order logic language that provides second order syntax.

ODESWS Framework

With the aim of designing and composing SWS, the ODESWS framework (Figure 1) has been proposed for SWS (Gómez-Pérez et al., 2004). The following elements have been identified:

- ***ODESWS Ontology.*** To describe the features of a web service, we have used a stack of ontolgies since they present the features in a formal and explicit way. This stack of ontologies is composed of 4 layers: data-type ontology, knowledge representation ontology, PSM Description Ontology and SWS ontology.
- ***Instance model***. Designing SWS means to instantiate each of the ontologies of the stack that describes what a service is: the domain ontology used by the service is built on top of the data type and knowledge representation ontologies; the service features are instances of both PSM and SWS ontologies. The whole instances constitute a model that specifies the SWS at the knowledge level. The essential idea here is that the user does not handle the ontologies directly but a graphical PSM alike structure.
- ***Checking model***. Once the instance model has been created, it is necessary to guarantee that the ontology instances do not present inconsistencies among them. Design rules will be needed to check this, particularly when ontology instances have been created automatically (as in the case of (semi) automatic composition).
- ***Translate model***. Although a service is modelled at the knowledge level, it must be specified in a SWS-oriented language to enable programs and external agents to ac-cess its capabilities. Therefore, once the instance that describes the WS is created and checked, it is automatically translated into any of the existent SWS representational language.

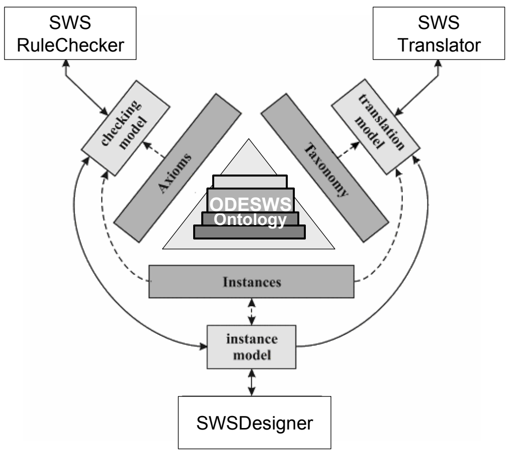

Figure 1 ODESWS Framework

This framework will enable the (semi) automatic composition of SWS using (1) PSM refiners and bridges to adapt the PSM ontology instances to the required capabilities of the new service; and (2) design rules to reject both PSM and SWS ontology instances that present errors or inconsistencies among them. Design rules are used to reduce the service candidates that are to be combined to obtain a new service.

ODESWS Ontology

The aim of designing SWS is to describe explicitly and semantically the features of a WS, as stated in the introduction. To achieve this purpose, the use of ontologies seems to be the most appropriate solution, and, in fact, this approach has been followed by other authors. In the OWL-S specification (OWL Services Coalition, 2004) the ontology is directly constructed in a semantic-enriched mark-up language (OWL), and the internal structure of the services is described with workflows. The WSMO ontology (Lara et al. 2004) uses the F-Logic language to specify the service and it introduces the concept of mediator to decouple the different elements of the service. Each of these solutions considers a particular semantic language to describe the ontology of SWS description.

The ODESWS framework, however, proposes an ontology (Figure 2), called ODESWS ontology (Gómez-Pérez et al., 2004), that describes the SWS at the knowledge and independent-language level. Thus, the ODESWS ontology is composed of a stack of ontologies developed following well-known specifications or de facto standards that cover all the features of the SWS. These ontologies are:

- ***Data Type (DT) Ontology***. It describes the types of concept attributes of the do-main ontology used to define the input/output parameters. The DT ontology is based on the XML Schema Datatypes (Biron and Malhotra, 2001), a W3C recommendation formally included into the semantic Web languages, as OWL.
- ***Knowledge Representation (KR) Ontology***. This ontology describes the knowledge primitives (concept, instance attribute, etc.) used to represent the domain ontology, which contains descriptions of the knowledge and data managed by the SWS. This ontology has been constructed on the basis of the WebODE knowledge model (Arpírez et al., 2003), which is frame-based and incorporates formulas to represent axioms.
- ***Problem-Solving Method Ontology***. It is based on the Problem-Solving Method (PSM) paradigm, where a PSM is a domain-independent and knowledge-level specification of a problem-solving behaviour which can be used to solve a class of problems (Motta, 1999). The PSM ontology is based on the Unified Problem-solving Method Language (UPML) (Fensel et al., 2003) that is a de facto standard for describing the PSM components: (1) tasks describe the operation to be solved in the execution of its solving method by specifying the input/output parameters and the pre/post-conditions required to be applicable (this description is independent of the method used for solving the task); (2) method details the control of the reasoning process to achieve a task. A method can be composite, which means that is composed of a number of subtasks whose coordinated execution indicates how the method will be carried out; and (3) adapters (Fensel, 1997) specify mappings among the knowledge components of a PSM. The adapters are used to achieve the reusability, since they bridge the gap between the general description of a PSM and the particular domain where it is applied.
  To specify the coordination of the subtasks of the composite methods, that is, the operational description of these methods, we have developed the workflow ontology. This ontology describes explicitly and semantically the workflow primitives (van der Aalst and van Kee, 2002; van der Aalst et al., 2003) that will be used to determine the coordination of the execution of the subtasks.
- **Semantic Web Service Ontology**. The SWS ontology is on top of the stack and it describes all the features of the services. This ontology replicates the upper-level concepts of the OWL-S ontology, but these concepts do not have the same attributes and relationships of the OWL-S specification. Thus, these upper-level concepts are connected with the concepts of the PSM ontology in the following way: (1) profile is a concept whose attributes specify the SWS non-functional features, and it establishes a relationship (*hasTask*) (see Figure 2) with the task concept of the PSM ontology to describe the SWS functional features; (2) model is a concept that establishes a relationship (*hasMethod*) with the method concept of the PSM ontology; it describes the components of the internal structure of the service and the control flow that indicates how those components are coordinated to solve the task related to the functional features of the service; and (3) grounding, which specifies the access protocol and the necessary message exchanges to invoke the service.

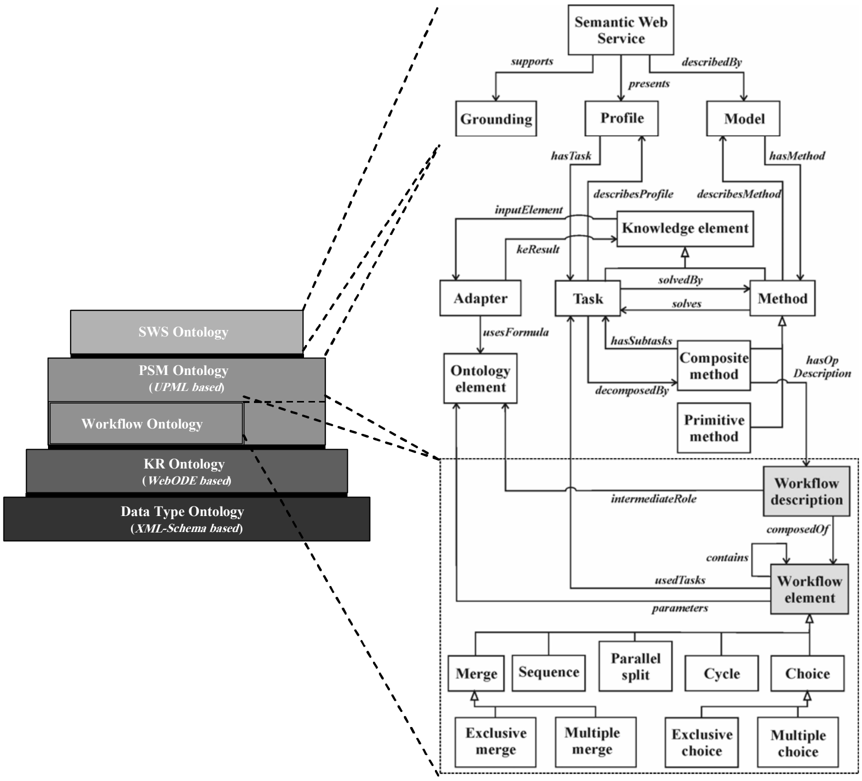

Figure 2 ODESWS Ontology

References

(Alonso, 2003) Alonso G., Casati F. Harumi Kuno, Vijay Machiraju (2003) Web Services, Concepts, Architecture and Applications. Springer.
(Ankolenkar et al., 2002) Ankolenkar, A., Burstein, M., Hobbs, J.R., Lassila, O., Martin, D.L., McIlraith, S.A., Narayanan, S., Paolucci, M., Payne, T., Sycara, K., Zeng, H.: DAML-S: Web Service Description for the Semantic Web. Proceedings of the First International Semantic Web Conference. Sardinia, Italy (2002) 348–363
(Arpírez et al., 2003) Arpírez, J.C., Corcho, O., Fernández-López, M., Gómez-Pérez, A.: WebODE in a Nutshell. AI Magazine.
24 (2003) 37–48
(Berners-Lee et al., 2001) Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American. 284 (2001) 34–43
(Biron and Malhotra, 2001) Biron P.V. and Malhotra A.. XML Schema Part 2: Datatypes. World Wide Web Consortium, May 2001. http://www.w3.org/TR/xmlschema-2/.
(Box et al., 2000) Box, D., Ehnebuske, D., Kakivaya, G., Layman, A., Mendelsohn, N., Nielsen, H.F., Thatte, S., Winer, D.: Simple Object Access Protocol (SOAP) Version 1.1. http://www.w3.org/TR/2000/NOTE-SOAP-20000508 (2000)
(Christensen et al., 2001) Christensen, E., Curbera, F., Meredith, G., Weerawarana, S.: Web Service Description Language (WSDL) 1.1. http://www.w3c.org/TR/2001/NOTE-wsdl-20010315 (2001)
11.Box, D., Ehnebuske, D., Kakivaya, G., Layman, A., Mendelsohn, N., Nielsen, H.F., Thatte, S., Winer, D.: Simple Object Access Protocol (SOAP) Version 1.1. http://www.w3.org/TR/2000/NOTE-SOAP-20000508 (2000)
(Cubera et al., 2001) Curbera, F., Nagy, W.A., Weerawana, S.: Web Service: Why and How?. Proceedings of the OOPSLA-2001 Workshop on Object-Oriented Services. Tampa, Florida (2001)
(Dean et al., 2004) Dean, M., Schreiber, G. (eds.): OWL Web Ontology Language Reference. W3C Candidate

Recommendation. http://www.w3c.org/TR/owl-ref (2004)
(Fensel et al., 2003) Fensel, D., Motta, E., van Harmelen, F., Benjamins, V.R., Crubezy, M., Decker, S., Gaspari, M., Groenboom, R., Grosso, W., Musen, M.A., Plaza, E., Schreiber, G., Studer, R., Wielinga, B.: The Unified Problem-Solving Method Development Language UPML. Knowledge and Information System. 5 (2003) 81–131
(Fensel and Bussler, 2002) Fensel D. and Bussler C. The Web Service Modeling Framework WSMF, Electronic Commerce: Research and Applications, 1 (2002) 113-137
(Fensel, 1997) Fensel, D.: The Tower-of-Adapter Method for Developing and Reusing Problem-Solving Methods. Proceedings of the 10th Knowledge, Modeling and Management Workshop. Lecture Notes in Computer Science, Vol. 1319. Springer-Verlag, Berlin Heidelberg (1997) 97–112
(Kayne, 2003) Kayne D. (2003)"Loosely Coupled, The Missing Pieces of Web Services" Rds Associates Inc
(Kreger, 2001) Kreger, H.: Web Services Conceptual Architecture (WSCA 1.0). IBM Software Group. http://www.ibm.com/software/solutions/webservices/pdf/WSCA.pdf (2001)
(Gómez-Pérez et al., 2004) Gómez-Pérez, A., González-Cabero, R., and Lama, M. 2004, A Framework for Design and Composing Semantic Web Services", IEEE Intelligent Systems, vol. 16, pp. 24–32
(Kifer et al., 1995) Kifer M., Lausen G., and Wu J.. Logical foundations of object-oriented and framebased languages. Journal of ACM (JACM), 42:741{843, July 1995.
(Hendler, 2001) Hendler, J.: Agents and the Semantic Web. IEEE Intelligent Systems. 16 (2001) 30–37
(McIlraith et al., 2001) McIlraith, S.A., Son, T.C., Zeng, H.: Semantic Web Services. IEEE Intelligent Systems. 16 (2001) 46–53
(Motta, 1999) Motta, E.: Reusable Components for Knowledge Modelling. IOS Press, Amsterdam, The Netherlands (1999)
(Motta et al., 2003) E. Motta, J. Domingue, L. Cabral M. and Gaspari (2003). IRS-II: A Framework and Infrastructure for Semantic Web Services. In proceeding of the 2nd International Semantic Web Conference (ISWC2003) 20-23 October 2003, Sundial Resort, Sanibel Island, Florida, USA
(OWL Services Coalition) OWL-S 1.0 Release: Semantic Markup for Web Services. http://www.daml.org/services/owl-s/1.0/owl-s.pdf (2004)
(Roman et al., 2004) Roman, Keller, U., Lausen, H. (eds.): Web Service Modelling Ontology (WSMO). http://www.wsmo.org/2004/d2/v01 (2004)
(Sollazo et al., 2002) Sollazo, T., Handshuch, S., Staab, S., and Frank, M.: Semantic Web Service Architecture – Evolving Web Service Standards toward the Semantic Web. Proceedings of the 15th International FLAIRS Conference. Pensacola, Florida (2002)
(van der Aalst and van Hee, 2002) van der Aalst, W.P., van Hee, K.: Workflow magament – Models, Methods, and Systems. MIT Press. Cambridge, Massachusetts (2002)
(van der Aalst et al., 2003) van der Aalst, W.P., ter Hofstede, A.H., Kiepuszewski, B., Barros, A.P.: Workflow patterns. Distributed and Parallel Databases. 14 (2003) 5–51