Integration of OWL-S into IRS-III

Farshad Hakimpour, John Domingue, Enrico Motta,
Liliana Cabral and Yuangui Lei

Knowledge Media Institute, The Open University, Milton Keynes, UK
{f.hakimpour, y.lei, j.b.domingue, e.motta, l.s.cabral}@open.ac.uk

Abstract. IRS-III is the first WSMO compliant system for supporting the Semantic Web Services technologies and it is based on the IRS-II [3]. This paper presents how we integrated the OWL-S [5] service description ontology to IRS-III. We describe how the underlying model of IRS-III supports OWL-S.

1 Introduction

As Web Service technologies are evolving, the need for semantic description of the services has also been increasing. There is a great deal of work on developing specifications for describing Web Services, such as in OWL-S [5] and WSMO [1]. The description of Web Services facilitates automatic matching of services with a service request, automatic service composition, controlling and monitoring the execution of a Web Service. IRS-III [3] is one of the few existing systems to support the Semantic Web Services technologies. IRS-III complies with the WSMO ontology of Goal, Web Services and Mediators [1]. WSMO is a developing specification mainly supported and developed by European partners.[1] IRS-III uses a version of WSMO ontology defined in OCML (a knowledge representation language [4]) which also provides the corresponding reasoning system. OWL-S provides a specification for describing semantics of Web Services developed as part of DAML program. Supporting OWL-S will extend the potential of the IRS-III and would also help us to explore the similarities and the differences between OWL-S and WSMO ontology.

In this paper, we explain how ontologies describing a service in OWL-S specification are mapped to the WSMO ontology and translated to OCML which is in turn suitable to be used by IRS-III. The rest of the paper is organized as follows. Section 2 introduces the IRS-III, WSMO ontology and the OWL-S specification. In Section 3, we describe how major features of the OWL-S Process are translated to Web Service descriptions. Section 4 is dedicated to the technical details of OWL-S to WSMO translation, as well as OWL to OCML translation. Finally, in Section 5 we present a summary.

2 Background

IRS (Internet Reasoning Service) is a framework to support Semantic Web Services, in which services can be described by their semantics, discovered, invoked and monitored. IRS-III consists of three main components, IRS Server, IRS Publisher and IRS Client. IRS Server stores and reasons with Web Service and Goal descriptions. IRS Publisher generates wrappers for programs as Web Services. Invocation of Web

1.http://www.wsmo.org/

Services via Goal descriptions is supported by the IRS Client.

The notions of Goal and Mediator are particular characteristic of IRS-III and WSMO ontology. While a Web Service is a description of a method and concerned with the specification of mechanisms and execution, a Goal is a general description of a problem and concerned with describing a problem rather than mechanisms. As a result, Goals are suitable for describing a service for a user whose concern is not the technical details of the solution.

OWL-S [5] is based on a process ontology and benefits from developments in workflow technologies. OWL-S model is based on three major components ServiceModel, ServiceProfile and ServiceGrounding. The most important of the three for IRS-III is the ServiceModel which describes how a service works by describing its Inputs, Outputs, Preconditions and Effects (IOPEs). It also specifies the component Processes of composite services and their execution order. ServiceProfile foresees information that may be required to search for a service, such as, ContactInformation, QualityRating, ServiceCategory and other optional ServiceParameters. Furthermore, a Profile contains pointers to the IOPEs of a Process. ServiceGrounding specifies the details of accessing a service, such as communication protocol and message format. ServiceGrounding is not discussed in this paper, as we concentrate on the semantic description of services here.

3 OWL-S Process and Web Service

The core functional description of services in OWL-S appears in Process descriptions. A Process is described mainly in terms of its functional parameters: Inputs, Outputs, Preconditions and Effects (IOPEs). OWL-S also divides Processes into two types AtomicProcess and CompositeProcess in the ProcessModel. CompositeProcesses are further described by ControlStructures. We map Atomic Process and Composite Process to a Web Services description in IRS-III. In case of a composite web service the composition will be translated to the orchestration part of the Web Service description.

Functional Parameters. Web Service description in IRS-III includes: has-input-role and has-output-role that are also present in the OWL-S process description, as has-Input and hasOutput. The Capability of a Web Service in IRS-III includes: has-precondition, has-assumption, has-effect and has-postcondition. These roles are similar to hasPrecondition and hasEffect in defined in OWL-S 1.0. Since OWL-S 1.1 (Beta version) Effects are part of a new property called hasResult.

The distinction between preconditions and assumptions (as well as postconditions and effects) are result of the distinction between the state in the information space and state of the world. That is, preconditions and postcondions are conditions related to the information space while assumptions and effects are related to the state of the world. At the moment, we map the OWL-S preconditions to IRS assumptions.

On the other hand, OWL-S 1.1 provides extra parameters, namely, Local and ResVar. These parameters are used respectively in the Preconditions and Results. These parameters can be thought of as environment variables that represents states of the world. By this interpretation, OWL-S 1.1 also distinguishes between the state of the information space and the state of the world. However, this distinction is done at the level of

parameters rather than the conditions.

Goals and Web Services. The IRS Web Service is suitable for representing a service description as described by Process in OWL-S. However in IRS-III, the notion of Goal refers to a general description of a problem and can be solved by different Web Services. A Goal describes a problem to be solved and represents the knowledge required for matching the problem to a set of Web Service descriptions presented by providers. During the translation a user may decide whether a Goal based on the OWL-S description should also be generated by the translator. While generating the Goal translator will also generate the necessary mediator to mediate between the Goal and the Web Service. After the translation one can modify the generated Goal and mediator by the IRS browser. Generation of a Goal allows service discovery based on the Goal description. If a user does not require to generate a Goal she may associate a Web Service to an existing Goal, later by the IRS browser. That is, a translated OWL-S description can be associated to any existing Goal.

There is a conceptual difference between the IRS-III and OWL-S. OWL-S views a process (e.g. Amazon-book-selling-service) as an instance of class Process, while in the IRS-III the translated result is defined as a subclass of Web Service. However, this difference has side effect on translation or other IRS-III functionality.[2]

Composition. Describing Service compositions is an essential advantage of the reusability of Web Services. The IRS-III composition model is currently under development and the translator will soon be able to support translation of the composite processes. However, we developed a translator from OWL-S 1.0 to the previous version of IRS (i.e. IRS-II). OWL-S describes a composite process by means of Control Structures. OWL-S Control Constructs are built as a set of Processes, such as a Sequence of processes or an iteration of a list of processes. The same is supported in IRS-II by means of body of a PSM definition in OCML. As a result all control construct in OWL-S can be

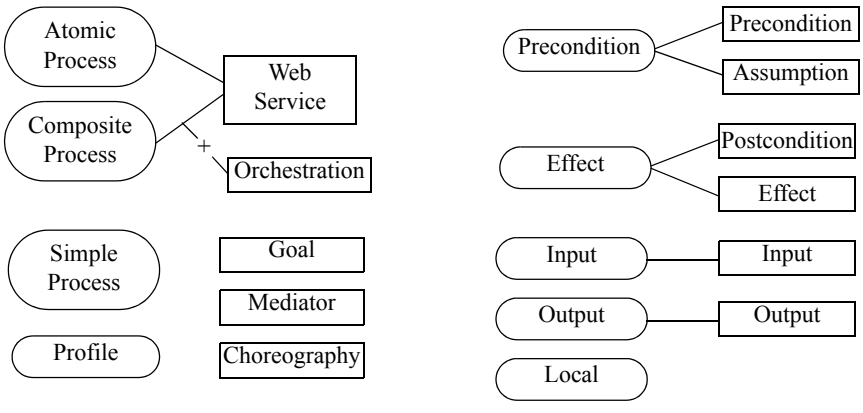

Fig. 1. Similarity of the basic elements of OWL-S (in ovals) and WSMO (in boxes).

2.This difference has started as of OWL-S version 1.0. In OWL-S 0.9 and before a Process (e.g. BravoAir_Process) was defined as a subclass of Process rather than its instance.

easily translated to OCML and used by IRS-II.

In General, IRS allows building a composition of Goals. This feature provides a certain level of dynamism in composing Web Services. Such dynamism can only be achieved by SimpleProcesses in OWL-S. That means, Web Services executed in a composition can be discovered during the execution and should not be necessarily specified during composing process.

4 Translation

The translator contains two components. Firstly, translation of OWL-S to the WSMO ontology and secondly, translation of OWL to OCML. The common concepts in both OWL-S and WSMO ontology are translated by the former component. However, OWL-S service descriptions are in OWL language and contain many definitions in OWL (e.g., concepts defining the input types of a Process). These definitions are translated by the later component. The translator also validates the consistency of the OWL-S service descriptions. For example, Profile is associated to Process descriptions by means of the has_process property. We also validate the consistency of this association with the hasProcess property of the ServiceModel for the Service. There are a number of such validation that are performed on the service descriptions.

5 Summary

This paper describes how we mapped the OWL-S descriptions to WSMO. We explained similarities and differences between OWL-S and the WSMO ontologies. OWL-S uses the ProcessModel for modelling and describing Web Services, which is similar to Web Service in the WSMO ontology. The functional parameters of OWL-S Processes (IOPE's) have similar concepts in the WSMO ontology. The paper discusses how the separation of the Goal and the Web Services can add to the flexibility in defining a composition of tasks.

Acknowledgement

This work is partly supported by and the AKT (GR/N15764/01) project sponsored by the UK Engineering and Physical Sciences Research Council and the DIP (FP6 - 507483) project funded under the European Union's IST programme.

References

1. Fensel, D. and Bussler, C.: The Web Service Modeling Framework WSMF. In Electronic Commerce: Research and Applications, Vol. 1, No. 2, (2002) 113-137
2. McGuinness D. L. and Harmelen F. V. (eds.): OWL Web Ontology Language Overview. http://www.w3.org/TR/owl-features/ (2004)
3. Motta, E., Domingue, J., Cabral, L., Gaspari, M. IRS-II: A Framework and Infra-structure for Semantic Web Services. In the Proceedings of The 2nd Int'l. Semantic Web Conf. (ISWC2003), Fensel, D. et al. eds., Springer-Verlag, LNCS Vol. 2870, (2003) pp. 306-318
4. Motta, E.: Reusable Components for Knowledge Modelling: Case Studies in Parametric Design Problem Solving, IOS Press (1999)
5. OWL Services Coalition, OWL-S: Semantic Markup for Web Services, white paper, www.daml.org/services/owl-s/1.0, (Nov. 2003)