An Experience report on using DAML-S

Conference or Workshop Item

An experience report on using DAML-S

Marta Sabou [a] Debbie Richards [b] Sander van Splunter [a]

[a] Dept. of Artificial Intelligence, Vrije Universiteit Amsterdam
[b] Division of I.C.S., Macquarie University, Sydney

Abstract

Though DAML-S is growing into a *de facto* standard for semantic web-service markup, we have only found few complete service descriptions and even less papers discussing technical issues about the markup process. We addressed this lack by (1) reporting on our experiences in describing a set of services, (2) concluding several limitations of the latest DAML-S version (v0.7) and (3) making our work accessible to the research community[1].

1 Introduction

DAML-S is an initiative of the Semantic Web community to facilitate automatic discovery, invocation, composition, interoperation and monitoring of web-services (WSs) through their semantic description [1]. DAML-S is a DAML+OIL ontology conceptually divided into three sub-ontologies for specifying *what a service does?* (Profile), *how the service works?* (Process) and *how the service is implemented?* (Grounding). The existing grounding allows aligning the semantic specification with implementation details described using WSDL, the industry standard for web-service description.

DAML-S has generated a lot of interest through its promise to add semantics to web service descriptions. Despite that interest we were only able to find a small number of DAML-S descriptions of web services and most of these did not point to real services and were from the DAML-S community. Within the DAML-S coalition two complete, fictitious examples (on the DAML-S site) are provided. Several other projects use only certain parts of the DAML-S ontology, eg. match-making research tends to focus on the Profile ontology. Other researchers report on extending parts of DAML-S, e.g. by enriching the Process/Profile ontologies. Finally, some papers do mention use of complete DAML-S as is, but their purpose was to describe other research work therefore ignoring any details about their experiences with the language. Common to all the existing papers is that none of them describe the process of writing the DAML-S markup.

We sought to fill this gap by providing a set of complete, real web service descriptions and sharing our modelling experiences.

[1] All services available at http://www.cs.vu.nl/~marta/services/

2 Using DAML-S

We marked up a set of web services which are used to build web-portals from semantically annotated bibliographic information. An agent based configuration service uses these descriptions to configure the suit of services depending on the characteristics of the input data. Based on our modelling experiences we have distilled some general observations about DAML-S.

We have concluded that DAML-S is superior to existing WS languages as it allows use of formally defined domain knowledge. It goes beyond syntactic description of a service by providing a semantic description. Semantics allow reasoning about a service and move us towards the ultimate goal of dynamic service discovery and usage. The other key strength of DAML-S is that it links to an industry standard, namely WSDL. In this way, it indeed fulfills its role as a link between the Semantic Web community and industry.

However, we also encountered a set of shortcomings.

A) Imprecise conceptual model. While it is commendable that DAML-S seeks to provide flexibility and thus has not fully defined a number of its concepts, this flexibility comes at the expense of clarity. The result of this imprecision is that DAML-S has an imprecise underlying conceptual model. We base this on the following facts. First, the three parts of DAML-S employ different metaphors to describe the same service: a program metaphor, an action metaphor and a view based on network endpoints. Second, several links exist between the conceptual models however they are often unclear. Our paper demonstrates that the link between the Profile and Process ontologies leads to inconsistencies in the final descriptions. Third, there is no clear correspondence of DAML-S concepts with software engineering (SE) concepts therefore it is often ambiguous how to model well-accepted software paradigms.

B) Mapping to WSDL limits DAML-S expressivity. We have experienced that the mapping to WSDL often limits the expressivity of DAML-S. Just by modelling a simple service we conflicted with two out of three basic assumptions that underlie the mapping. This forced us to revise our descriptions so that a grounding was possible at the expense of giving up specification of parametric polymorphism or an accurate specification of a complex internal structure. We feel that DAML-S is influenced too much by the actual grounding details.

C) Difficult to learn. One of our major comments (and worries) is that it was quite difficult to get started with writing DAML-S. The previously mentioned lack of conceptual model played a fair role in this. Other inhibiting factors were the limited tool support and the low number of provided examples.

We hope that both our observations and actual examples will contribute to the development and large scale usage of DAML-S and move our community closer to realizing the Semantic Web.

References

[1] DAML Services Coalition. DAML-S: Semantic Markup for Web Services. DAML-S v. 0.7 White Paper, October 2002.