Towards two-stage service representation and reasoning: from lightweight annotations to comprehensive semantics

Conference or Workshop Item

Towards two-stage service representation & reasoning: from lightweight annotations to comprehensive semantics

Stefan Dietze, Neil Benn, Hong Qing Yu, Carlos Pedrinaci
Knowledge Media Insititute
The Open University
Milton Keynes, MK76AA, UK
{s.dietze, n.j.l.benn, h.q.yu, c.pedrinaci}@open.ac.uk

Roland Siebes
Department of Computer Science
Vrije Universiteit
Amsterdam, The Netherlands
Telephone number, incl. country code
{r.m.siebes}@few.vu.nl

Dong Liu, John Domingue
Knowledge Media Insititute
The Open University
Milton Keynes, MK76AA, UK
{d.liu, j.b.domingue}@open.ac.uk

ABSTRACT
Semantics are used to mark up a wide variety of data-centric Web resources but are not used to annotate online functionality in significant numbers. That is despite considerable research dedicated to Semantic Web Services (SWS). This has led to the emergence of a new Linked Services approach with simplified and less costly to produce service models, which targets a wider audience and allows even non-SWS developers to annotate services. However, such models merely aim at enabling semantic search by humans or automated service clustering rather than automation of service tasks such as discovery or orchestration. Thus, more expressive solutions are still required to achieve automated discovery and orchestration of services. In this paper, we describe our investigation into combining the strengths of two distinct approaches to modeling semantic Web services – "lightweight" Linked Services and "heavyweight" SWS automation - into a coherent SWS framework. In our vision, such integration is achieved by means of model cross-referencing and model transformation and augmentation.

Categories and Subject Descriptors
D.3.3 [**Programming Languages**]:

Keywords
Semantic web services, semantic web, WSMO, WSMO-Lite, web services, minimal service model.

1. INTRODUCTION
The past decade has seen a range of research efforts in the area of Semantic Web Services (SWS), mainly aiming at the automation of Web service–related tasks such as discovery, orchestration or mediation via broker-based approaches. Building on formal service semantics, several frameworks, such as SAWSDL [8], OWL-S [6] and WSMO [4], have been proposed which aim at formalizing semantic service descriptions, which usually cover aspects such as service capabilities, interfaces or non-functional properties. Besides, a considerable research community evolved around these SWS frameworks, providing, for instance, annotation and execution tools based on these formal SWS frameworks [3][2].

In the Web context semantics are used to mark up a wide variety of data-centric resources but are not used to annotate online functionality in any form in significant numbers. The reasons for this are two-fold. Firstly, SWS research has for the most part targeted WSDL/SOAP-based Web services, which are not prevalent on the Web. Secondly, due to the inherent complexity required to fully capture computational functionality, creating SWS descriptions has represented an important knowledge acquisition bottleneck and has required the use of rich knowledge representation languages and complex reasoners. There exists an inherent conflict between the need to capture comprehensive and meaningful service semantics – to allow reasoning-based automation of any sort – and the requirement to keep the costs for providing services descriptions low in order to simplify the modeling process and to ensure that efficient and scalable solutions can be implemented. Hence, despite considerable amount of research dedicated to the SWS visio, so far there has been little take up of SWS technology within non-academic environments.

The prevalent lack of impact of SWS technology is particularly concerning since Web services as such are in widespread use throughout the Web nowadays, where applications use distributed HTTP requests via rather lightweight interface technologies such as RESTful services, HTTP GET-style request or XML-feeds. Hence, the SWS challenges are of increasingly crucial importance for today's highly distributed Web applications. These issues led to the emergence of more simplified SWS approaches to which we shall refer here as "lightweight", such as WSMO-Lite [9] or the Micro-WSMO/hRESTs [5] approach which replace "heavyweight" service semantics with less comprehensive and less costly to produce service models represented in RDF and hence, complying with the infrastructure of the growing *Semantic Web*. Analogous to the *Linked Data* term [1], this approach was recently dubbed as the *Linked Service* approach [7]. Due to the fact that such service annotations are much easier to produce and can be populated with references to widely established Linked Data vocabularies, they address a much wider audience and allow even non-SWS experts and lay people to describe and annotate services. However, those models merely aim at enabling structured, semantics-enabled search by humans or automated service clustering, and more expressive solutions are required to achieve greater levels of automation.

2. TWO-STAGE SERVICE ANNOTATION AND REASONING
In order to tackle the introduced challenges, we aim at combining the two distinct SWS representation approaches

(R1) lightweight Linked Services, and

(R2) heavyweight SWS descriptions.

While both approaches partially share common schema entities, e.g. both cover aspects such as *interfaces* and *non-functional properties* of services, they differ significantly in certain other aspects, for instance, the way the service models are being produced, the nature of the actual produced models or the kind of reasoning facilitated by each approach. For instance, while (R1) is being produced collaboratively as a joint effort by a potentially large group of service providers and consumers, it allows to consider a range of perspectives on one particular service and to gather annotations and RDF-model references to a wide range of existing RDF vocabularies. Hence, they can be described as *multi-faceted*, *deliberately incomplete* and *incoherent*. In contrast, the models usually subsumed under (R2), e.g. WSMO-based service specifications, reflect the perspective of one particular SWS provider and describe a service following a meta-model which aims at exhaustive modeling of a service in terms of its core identifying aspects, such as its capabilities or behavioral characteristics. Here, one strives for a much greater level of expressivity and detail and particularly takes into account execution-related aspects. Therefore, such descriptions could best be described as *comprehensive*, potentially *complex* and *coherent*.

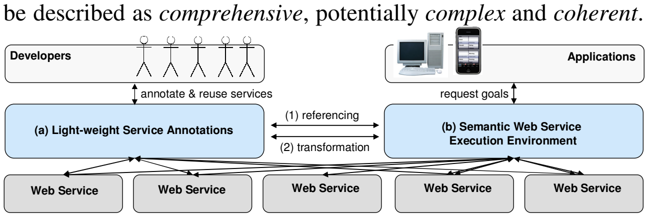

Fig. 1. From lightweight service annotations to heavyweight Semantic Web Services descriptions—the overall approach.

Depending on the quality of the produced service models, the representational approach (R2) facilitates reasoning that allows for automation of certain service-related tasks such as discovery or orchestration but are costly to produce. In contrast, models as in (R1) are less intricate, but also allow only limited reasoning, such as clustering of services or structured searches by humans.

While these approaches currently co-exist without a well-defined relationship, we propose two different bi-directional correlations, which are under investigation:

(C1) service model cross-referencing,
(C2) service model transformation and augmentation.

Under (1), we subsume all kinds of references between models across (a) and (b) as depicted in Fig 1. For instance, a lightweight service annotation could point to a heavyweight SWS description that models the same service more exhaustively or vice versa. That would allow semantics to be exploited in (a) as well as (b) for reasoning of different sorts, for instance, to perform some clustering based on (a) to reduce the amount of potentially interesting services for a given query in (b). In addition, (2) considers the transformation between models across (a) and (b), either manually or (semi-)automatically.

3. CONCLUSION

We have described a two-stage approach to semantic service representation. By integrating collaborative and user-driven Web-scale service annotations with comprehensive SWS specifications, application developers benefit from both low cost for providing annotation and a high level of automation. In that, while taking advantage of service models produced by a large non-expert audience, both structured search for service instances by humans as well as automation of service tasks is supported. In our vision, integration between lightweight service annotations and comprehensive SWS specifications is achieved by different means of (a) model cross-referencing and (b) model transformation and augmentation. While the current solution provides an overall framework for integrated service models which support different levels of automation, future work needs to address the investigation of automated model transformation mechanisms in order to support the seemless integration of instances across distinct service models schemas. Besides, future work needs to investigate the effort required to populate the introduced knowledge bases and the level of automation which is supported.

4. REFERENCES

[1] Bizer, C., T. Heath, et al. (2009). "Linked data - The Story So Far." Special Issue on Linked data, International Journal on Semantic Web and Information Systems (IJSWIS).

[2] Dietze, S., Benn, N., Domingue, J., Conconi, A., and Cattaneo, F. (2009) Two-Fold Semantic Web Service Matchmaking – Applying Ontology Mapping for Service Discovery, 4th Asian Semantic Web Conference, Shanghai, China.

[3] Domingue, J., Cabral, L., Galizia, S., Tanasescu, V., Gugliotta, A., Norton, B., Pedrinaci, C.: “IRS-III: A broker-based approach to semantic Web services”, Jounal of Web Semant, pp. 109-132. Elsevier Science Publishers B. V, 2008.

[4] Fensel, D.; Lausen, H.; Polleres, A.; de Bruijn, J.; Stollberg,, M.; Roman, D.; and Domingue, J. 2007. Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer.

[5] Kopecky, J.; Vitvar, T.; and Gomadam, K. 2008. MicroWSMO. Deliverable, Conceptual Models for Services Working Group, URL: http://cms-wg.sti2.org/TR/d12/v0.1/20090310/d12v01_20090310.pdf.

[6] Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIlraith, S., Narayanan, S., Paolucci, M., Parsia, B., Payne, T., Sirin, E., Srinivasan, N., and Sycara, K. (2004). OWL-S: Semantic Markup for Web Services. Member submission, W3C. W3C Member Submission 22 November 2004.

[7] Pedrinaci, C., Liu, D., Maleshkova, M., Lambert, D., Kopecky, J., and Domingue, J. (2010) iServe: a Linked Services Publishing Platform, Workshop: Ontology Repositories and Editors for the Semantic Web at 7th Extended Semantic Web Conference.

[8] Sheth, A. P., Gomadam, K., and Ranabahu, A. (2008). Semantics enhanced services: Meteor-s, SAWSDL and SA-REST. IEEE Data Eng. Bul l., 31(3):8–12.

[9] Vitvar, T.; Kopecky, J.; Viskova, J.; and Fensel, D. 2008. Wsmo-lite annotations for web services. In Hauswirth, M.; Koubarakis, M.; and Bechhofer, S., eds., Proceedings of the 5th European SemanticWeb Conference, LNCS. Berlin, Heidelberg: Springer Verlag.