Rethinking the Semantic Annotation of Services

Nikolaos Loutas[1,2], Vassilios Peristeras[1] and Konstantinos Tarabanis[2]

[1] National University of Ireland, Galway, Digital Enterprise Research Institute
{firstname.lastname}@deri.org
[2] Information Systems Lab, University of Macedonia, Thessaloniki, Greece
{nlout, kat}@uom.gr

Abstract. This work extends and enhances existing semantic service models by involving users and by including service metadata related to the user's view of the service and their behaviour. We borrow ideas and extend the models and practices for the annotation of Web content and information resources that has recently become popular in widely-used social platforms. Users are encouraged to describe in their own terms the services they use. Our approach strengthens user participation in the Web and more generally in the service industry by providing service metadata, which are later used as a form of lightweight user-side semantic annotation of services. This annotation is provided explicitly by the users and/or implicitly by identifying patterns in the users' behaviour. This type of service annotation acts supplementary to the service descriptions provided by the service providers and is linked to the actual use of the services. Finally, we harvest the collected metadata and use it for facilitating discovery and clustering of services, as well as to enable service recommendations and matchmaking with users' profiles.

Keywords: service, SOA, semantic, social contract, social metadata

1 Introduction

The Service Oriented Architecture (SOA) paradigm is the prevalent way of developing enterprise Information Systems (IS). The main idea behind SOA is the ability to (re)use and share services from different sources. The introduction of Web Services (WS) contributed significantly to SOA's commercial uptake and helped SOA adoption by industry. Further on IBM proposed a reference architecture for SOA which identifies three basic entities: the service provider, the service requestor and the service broker. This reference architecture supports four fundamental functionalities, namely service discovery, composition, publishing and invocation.

There has been a lot of debate with respect to the advantages and disadvantages of SOA. The advantages of SOA include enhanced organizational agility and architectural flexibility, reduced costs and increased Return on Investment [18, 23]. Nevertheless, the real added value of SOA implementations is still questioned as technological, organizational and financial barriers hinder the uptake of SOA [3].

Semantics have been applied in SOA as a means to enhance the WS brokerage model. The idea is that semantically described WS could enable and facilitate the dynamic discovery, invocation, execution, composition and monitoring of services at run-time [10]. This led to the definition of various service ontologies and Semantic Web Services (SWS) models starting from DAML-S, continuing to OWL-S [11], WSMO [17], WSDL-S [13] and followed by SAWSDL [5].

In this paper, we add an additional layer to existing semantic approaches. Apart from the formal service descriptions (semantic or not) that are made available by the service providers in SOA environments, we enhance the service descriptions by "capturing" and "attaching" information, which is related to the actual usage of the services by users in the real world. Our intention is to add a social layer on top of SOA. We propose two main mechanisms for this:

- We allow users to annotate the services they use.
- We analyze the behavioural/service usage patterns of the users', i.e. how the users consume the services, by monitoring their behaviour and monitoring their actions.

Afterwards, the information that we collect is used to enrich the existing service descriptions. We call this process *social annotation of services* and the metadata that emerge bottom-up from this process *social metadata*. The social annotation may be supplementary to the semantic service descriptions that are already provided by the service providers.

In order to accommodate the social aspects and characteristics of services, existing semantic service models need to be extended. In this paper, we discuss the required extensions; we introduce an approach for adding social annotations to services and demonstrate how to utilize the extra (social) semantics that emerge in order to facilitate service search, mashing and recommendation.

Before continuing, it is necessary to clarify that our approach focuses mainly on extrovert services which have a business value and are to be used by the end-users.

The remainder of this paper is organized as follows: section 2 presents our motivation. Section 3 discusses in detail the idea of social annotation of services and introduces the notion of the Social Contract. Section 4 shows how an existing semantic service models can be extended in order to include social metadata. Section 5 describes our prototype. Finally, section 6 concludes the paper and discusses our research directions.

2. Motivation

All the semantic service frameworks proposed so far, e.g. WSMO, SAWSDL, OWL-S, both complex and lightweight, share a common principle: they assume that the (semantic) description of services comes solely from the service providers. This results into two serious limitations:

- The users are totally left outside of the service description process. These approaches do not take into account the way that the users of the service perceive it. For example, users cannot detail the reason they use a service, e.g. to book a flight, or the context in which the service is used, e.g. a service as part of a more general "travelling" context. Recently, both the research community and industry sensed this shortcoming. There are already attempts which focus on the user's perspective which was left completely out of picture in SOA e.g. [12].

- Semantic service efforts have still a low adoption rate. To come up with elaborated semantic service descriptions, the service providers have to be convinced about the additional value of these semantics in order to spend resources to annotate their services. However at the moment, it seems rather unlikely to convince service providers to use existing SWS frameworks, and as a result these efforts are not taking off.

So far, there has been no real large-scale application of SWS in industry. Among other problems in [24], the author claims that SWS have ill-defined semantics and that service ontologies usually describe the semantics of WSDL interfaces, which are different from the semantics of the WS. As such the existing SWS approaches have not managed to fully support automated discovery, matchmaking, composition, and execution. Generally, the high complexity of the SWS approaches discourages both technical and business people from adopting such solutions. These problems have created pressure to the SWS research community to come up with lightweight approaches, which may lack in expressivity but win in simplicity (e.g. SAWSDL, SA-REST [20], WSMO-Lite [22]). However, it is still early to evaluate the applicability and the adoption of such light semantic service models.

Lately, Web 2.0 is emerging as a new computing paradigm. Web 2.0 preaches for active user participation in the Web through user-centric Web portals and applications [14]. In Web 2.0 there is no clear line of separation between service providers and users as the latter interact with the Web not just as information receivers but also as content providers [4]. In Web 2.0 platforms users add data and metadata: they add content, e.g. photos, multimedia and documents, and then use tags for attaching meaning to this content. Other users also add metadata (tags), which are then used to enable better search (e.g. [2, 25]) and discovery, personalization of the user's experience etc.

Moreover, unlike SOA environments, in Web 2.0 semantics (metadata) come mainly from tags and folksonomies and as such emerge in a bottom-up fashion directly from the users (e.g. [7, 21]). We argue that as this user-defined metadata are used for creating richer descriptions for resources (e.g. photos, files etc.), they could be likewise applied to services for enhancing their descriptions. This metaphor is challenging and capitalizes on the view that Web 2.0 and SOA are two converging and complementary paradigms [19].

Within Web 2.0, new types of services appear which do not follow the SOA principles and are created in a decentralized manner. These services are usually generated by users and not by service engineers. For example, mashups, are introduced as a new simple way of composing services and combining content from different sources. Interestingly enough, Web 2.0 services lack a standardized description from their providers, as there are neither standardized ways to describe services nor public repositories to store these descriptions. The services are generated in a completely decentralized and uncontrolled way and the overall architecture lacks the core SOA idea of a WS broker that mediates between the service providers and service clients.

Thousands of mashups are available at the moment and all indications show that this number may soon scale to millions. All these mashups are capable to provide access to huge amount of distributed content and/or services. But how one can find the mashup (s)he really needs, when (s)he needs it? How can a driver that is running out of gas find if there is a mashup that displays on a map the open gas stations in the area where (s)he is driving? How can the four basic SOA functionalities, namely discovery, composition, publishing and invocation be achieved in a Web 2.0 environment? Currently the means, e.g. models, architectures and applications for answering such questions are not available.

Thus, in this work we will show how social metadata can be included in service descriptions, let them be SOA services or RESTful services, in order to facilitate service search, mashing and recommendation.

3. Social Annotation of Services

A service can be seen from different perspectives starting from a complex business process and going down to software component [1]. In [22] five complementary parts of a service's description, called service contracts, are defined, namely:

- The *Information Model* which refers to the data model that is used to semantically describe the service inputs, outputs and fault messages.
- The *Functional Descriptions* which describe the service's functionalities.
- The *Non-Functional Descriptions* which define details related to the implementation or the running environment of the service, e.g. name, author, URL, version.
- The *Behavioral Descriptions* which define the service's choreography and internal workflow.
- The *Technical Descriptions* which define details regarding the format of the messages, the communication protocols and the access points of the service.

In our previous work, we reviewed several SWS efforts. We observed that they can be grouped into two distinct categories:

- *SWS frameworks* which refer to formal and complex efforts for semantically annotating services, and
- *Semantic service models* which refer to emerging lightweight approaches for semantically annotating services.

In the first category we find OWL-S, SWSF, WSMO, IRS-III and WSMO-Lite, while WSDL-S and its successor SAWSDL, SA-REST and MicroWSMO are placed under the second category. Afterwards, we examined how the different SWS efforts address the service contracts described above.

Summarizing our comparative analysis, we found out that SWS frameworks provide the language for encoding the ontologies that form their Information Model. On the contrary, semantic service models allow the use of any ontological language. Thus, their Information Model may be comprised by a set of ontologies encoded using different languages. Moreover, both the Functional and the Behavioural Descriptions can be expressed in detail using a SWS framework. Although the semantic service models provide some means for specifying services' functionalities, these mechanisms lack in expressivity. Finally, regarding the Technical Descriptions both SWS frameworks and semantic service models rely mainly on the WSDL specification, excluding the case of SA-REST that refers to RESTful services, thus no WSDL file exists.

In all aforementioned approaches, the metadata of all the service contracts come solely from the service providers. In this work we argue that it will be highly beneficial both for users and for service providers, if the semantic descriptions of the services were enriched with information that comes from the actual usage of a service in a bottom-up fashion. The semantic description in this case emerges from the usage of the service, thus giving a social aspect to service annotation.

To achieve this, we introduced a new service contract, which we call social contract [9]. The social contract captures the way that the users' perceive the service, when and why they use the service etc. The social contract has been until now neglected when modelling and implementing services. The social metadata of services may derive explicitly or implicitly:

- *Explicitly*, where metadata is added by users who wish to describe the service in their own terms. Users can annotate services, similarly to what they currently do for products, content and multimedia in platforms like Flickr or YouTube. Users would describe why they use a service, for what reason, on which occasion etc. They could also add annotations that are related with attributes of the service like inputs or outputs. In other cases, users may express their satisfaction or dissatisfaction with regards to quality, usability, user-friendliness etc. However, some of the service attributes still remain to be solely described by the service provider, e.g. the Technical Descriptions.

  We propose the use of tags and tagging mechanisms as the means for enabling explicit service annotation. Tagging is easy and straightforward and users are already familiar with it. Tagging mechanisms can either rely completely on free text or it can be further supported by vocabularies and ontologies. Here lies a substantial difference between the annotations from the service providers and those that come from users. Service providers are more likely to use formal service models combined with structured vocabularies and ontologies, while users would rather use tags.
- *Implicitly*, where information about the service can be inferred by monitoring the user's behaviour while using the service and then enrich the service description. For example, imagine that a statistically significant number of users execute service B after service A or users with similar profiles have an interest for some particular services. In both cases, this information can provide us with interesting usage patterns that can be further exploited e.g. for a service recommendation system.

4. Including social metadata in existing semantic service models

Until know we have discussed at a conceptual level how social metadata could be included in existing semantic service models. In this section we will show how we applied this idea by extending an existing semantic service model, namely SA-REST.

SA-REST [20] introduces a lightweight approach for adding semantics to RESTful WS. It assumes that it is highly likely that when a WS is made available online, the provider will release an (X)HTML description of the service as well. Thus, SA-REST suggests to semantically annotate this (X)HTML service description using RDFa or GRDDL. SA-REST uses a set of predefined elements for annotating different attributes of a service, e.g. `input`, `output` and `sem-class`.

SA-REST is simple and easily extendable, mainly due to the fact that it is based on RDFa. Moreover, the lightweight semantic annotation of services, as suggested by SA-REST, fits very well with our line of work. Moreover, as the number of RESTful services is increasing influenced by the growth of Web 2.0, we expect lightweight efforts to semantically annotate services using RDFa to become popular. Thus, we chose SA-REST in order to semantically annotate the services of our portal and capture and include social metadata.

SA-REST like other similar approaches for semantically annotating services covers the five service contracts but does not support the social contract. This means that SA-REST assumes that the user has a passive role and that the service descriptions come only from the service provider. Summarizing, in SA-REST the information model of a service is identified by its inputs and outputs. Both the behavioural and the functional Descriptions are weakly defined in the SA-REST model due to the fact that SA-REST aims at providing a lightweight approach for service descriptions sacrificing its expressivity. Finally, technical information can be represented by the method and protocol elements, while, non-functional descriptions can be derived from the `domain-rel`, `sem-class` and `sem-rel` elements.

In order to include social metadata in SA-REST, we first express social metadata by means of a simple folksonomy and then make use of the sem-class element in order to create a link between a service and this folksonomy. In order to indicate that this `sem-class` element refers to the social metadata of this service, we make use of the `typeof` element and make a reference to the social contact concept of our model.

It is worth mentioning that the social contract, as introduced in this work, can be combined with any other SWS framework. For example, WSMO non-functional properties Web Services or a SAWSDL model references enable the inclusion of social metadata in WSMO and SAWSDL services respectively.

5. Prototype

For the needs of our prototype, we have scoped our focus on the eGovernment domain. We have selected eGovernment as our application domain due to previous work and expertise in the domain, but also because eGovernment is a challenging test-bed with thousands of services provided worldwide by public agencies to billions of clients. Another interesting characteristic of public administration is that its clientele is not restricted to a certain group of people with common needs or interests. In fact, public administration tries to cover the needs of practically each and every citizen. To achieve this, public administration tries to group the diverse needs of its clientele and translate them into services.

Social annotation of services can provide valuable input for public administration during the public service design process that could lead to higher quality services that would fit better to the clients' needs. Furthermore, public administration could predict future needs of its clientele, based on trends expressed through the social annotation of services. This will allow public administration to be more agile and proactive.

Our eGovernment portal plays the role of a national entry point to the services provided by public administration. In fact a prototype of the portal, which is available at `http://195.251.218.39/cyprus`, is currently used in a pilot study in Cyprus. Citizens use this portal to get information about public services. The descriptions of the services that are made available via the portal are semantically annotated using the extended semantic service model that is presented in section 4. Apart from the inclusion of social metadata, we have also included eGovernment domain specific semantics in the semantic descriptions of our services. Towards this direction, the GEA Public Service model [15, 16] was employed, which introduces a conceptual representation of a public service. As such, it introduces core concepts of a public service, such as service input and output, service provider, service preconditions, service domain etc. We used RDFain order to include in our semantic eGovernment service descriptions eGovernment domain specific semantics (Table 1). The semantic descriptions are initially automatically created on the service provider's side, but once they are released and used by the users, they are enriched and become more expressive as social metadata are added.

In our social eGovernment portal users can tag the services provided or use the tags already given by other users. Moreover, the users' behaviour while using the portal, e.g. when navigating from one service to another, is anonymously monitored. This allows us to extract social metadata from their behavioural patterns without violating their privacy.

In both cases mentioned above, social metadata is extracted, which is then used for enhancing the user's experience in the portal. Users can browse the underlying eGovernment service repository using different mechanisms, e.g. tags/tagclouds, keyword search. Moreover, they can get recommendations about services that are popular or beneficial or fit their profiles or are related with services that they have already used. A detailed description of the portal can be found in [8].

Table 1. Example of service description annotated with the extended SA-REST model

```
<div      xmlns:SA-REST=”http://knoesis.wright.edu/srl/SA-
REST”
       xmlns:islab=”http://islab.uom.gr”>
<div about="http://islab.uom.gr/DrivingLicense">
  The Driving License Issuance public service is provided
by the <div rel=‟islab:serviceprovider”
  resource=”http://islab.uom.gr/gea.owl#Prefecture”/>
Regional  Authorities  and  belongs  to  the  <div  rel=‟SA-
REST:domain-rel”
  resource=‟
http://islab.uom.gr/gea.owl#SubDomainCommunityAndSocialS
ervices”/> Community and Social Services domain
<div typeof=”islab:social” rel=”SA-REST:sem-class”
resource=http://islab.uom.gr/DL_social.rdf/>
</div> </div>
```

6. Conclusions and Future Work

In this paper we discussed the need for extending the existing SOA-based service descriptions by including socially derived metadata. We add a “social” layer on top of existing semantic or non-semantic SOAs. This layer enriches the service description and facilitates the discovery and automatic clustering of services.

Two mechanisms for capturing this metadata have been proposed, receiving the annotations explicitly and directly by the users through tagging and implicitly through analysis of users’ behaviour and usage patterns. In order to validate our arguments an extension of SA-REST which accommodates social metadata was proposed. In addition to that a social eGovernment portal was developed.

Harvesting the knowledge that can be extracted from social the social metadata of services is expected to benefit both users and service providers. From the user’s perspective, it allows them to:

- Express their view of the services they use by annotating these services, e.g. using tags.
- Personalize their service searches, thus improving the quality and the coherence of the result set and bridging the service discovery gap [6]. Social metadata, e.g. tags, can be added as criteria to search queries, thus narrowing down the result sets.
- Get recommendations about related services. Service platforms may suggest to their users services that share common tags.
- Form communities of interest and/or practice. As described earlier, social semantics can also be extracted by monitoring the users’ behaviour. In this case, users that tend to use similar services can be grouped in communities of interest. For example, a community of users who use online collaboration services.

Service providers can exploit the social annotation of services to:

- Improve the classification and clustering of services according to their functionality/output/behaviour etc. Since the semantics that emerge from the social annotation of services stem directly from the users, they can create a bottom-up classification of services.
- Get feedback on their services and improve their quality or design new services in order to cover emerging customers’ needs.

The benefits that social metadata are anticipated to have both to uses and service providers, set also the pillars of our future research plan. Hence, we will try draft and develop possible applications and prototypes and enhance existing systems by including and harvesting social metadata.

Acknowledgments

This work is supported by Science Foundation Ireland under grant SFI/08/CE/I1380 (Líon 2).

References

1. Baida, Z., Gordijn, J., Omelayenko, B.: A shared service terminology for online service provisioning. In: ICEC 2004, pp. 1-10 (2004)
2. Bao, S., Xue, G., Wu, X., Fei, B., Su, Z.: Optimizing web search using social annotations. In: 16th World Wide Web Conference, pp. 501-510 (2007)
3. Bloomberg, J.: Why is SOA taking so long?, ZapThink. http://searchsoa.techtarget.com/tip/0,289483,sid26_gci1132084,00.html (2005)
4. Domingue, J., Fensel D., González-Cabero, R.: SOA4All, Enabling the SOA Revolution on a World Wide Scale. In: 2nd IEEE International Conference on Semantic Computing (2008)
5. Farrell, J., Lausen, H.: Semantic Annotations for WSDL and XML Schema", W3C Recommendation, http://www.w3.org/TR/sawsdl/ (2007)
6. Fernandez, A., Hayes, C., Loutas, N., Peristeras, V., Polleres, A., Tarabanis, K.: Closing the Service Discovery Gap by Collaborative Tagging and Clustering Techniques. In: International Semantic Web Conference, Workshop on Service Discovery and Resource Retrieval in the Semantic Web (2008)
7. Gruber, T.: Ontology of Folksonomy: A Mash-Up of Apples and Oranges. International Journal of Semantic Web and Information Systems 3, 1–11 (2007)
8. Kopouki Papathanasiou, M., Loutas N., Peristeras V., Tarabanis K., Combining service models, semantic and Web 2.0 technologies to create a rich citizen experience. In: 2nd World Summit on the Knowledge Society, WSKS (1): 296-305 (2009)
9. Loutas, N., Peristeras, V., Tarabanis, K.: Extending Service Models to Include Social Metadata. In: WebSci'09: Society On-Line (2009)
10. Martin, D., Domingue, J.: Semantic Web Services, Part 2. IEEE Intelligent Systems 22(6), 8-15 (2007)
11. Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIlraith, S., Narayanan, S., Paolucci, M., Parsia, B., Payne, T., Sirin, E., Srinivasan, N., Sycara, K.: OWL-S: Semantic Markup for Web Services, W3C Member Submission, http://www.w3.org/Submission/OWL-S/ (2004)
12. Meyer, H., Weske, M.: Light-Weight Semantic Service Annotations through Tagging. In: Service-Oriented Computing 4294, pp. 465-470 (2006)
13. Miller, J., Verma, K., Rajasekaran, P., Sheth, A., Aggarwal, R., Sivashanmugam, K.: WSDL-S: Adding Semantics to WSDL - White Paper. Large Scale Distributed Information Systems, http://lsdis.cs.uga.edu/library/download/wsdl-s.pdf (2004)
14. Murugesan, S.: Understanding Web 2.0. IT Professional 9(4), 34-41 (2007)
15. Peristeras, V., Tarabanis, K.: The Governance Architecture Framework and Models. In: Saha, P. (ed.) Advances in Government Enterprise Architecture. IGI (2008)
16. Peristeras, K. Tarabanis, K.: Towards an Enterprise Architecture for Public Administration: A Top Down Approach. European Journal of Information Systems 9, 252-260 (2002)
17. Roman, D. Keller, U., Lausen, H., Bruijn, J.d., Lara, R., Stollberg, M., Polleres, A., Feier, C., Bussler, C., Fensel, D.: Web Service Modeling Ontology, Applied Ontology 1(1), 77-106 (2005)
18. Schmelzer, R.: The ROI of SOA. ZapThink, http://www.zapthink.com/report.html?id=ZAPFLASH-20050127 (2005)
19. Schroth, C., Janner, T.: Web 2.0 and SOA: Converging Concepts Enabling the Internet of Services. IEEE IT Professional 9(3), 36-41 (2007)
20. Sheth, A., Gomadam, K.. Lathem, J.: SA-REST: Semantically Interoperable and Easier-to-Use Services and Mashups. IEEE Internet Computing 11(6), 91-94 (2007)

21. Specia, L., Motta, E.: Integrating Folksonomies with the Semantic Web. In: European Semantic Web Conference 2007, pp. 624-639 (2007)
22. Vitvar, T., Kopecky, J., Fensel, D.: WSMO-Lite: Lightweight Semantic Descriptions for Services on the Web. CMS WG Working Draft (2008)
23. Windley, P.: Teaming up for SOA. Infoworld, pp 25-26 (2007)
24. Xuan S.: Semantic Web Services: An Unfulfilled Promise. IEEE IT Professional 9(4), 42-45 (2007)
25. You, G., Hwang, S.: Search structures and algorithms for personalized ranking. Journal of Information Sciences 178(20), 3925-3942 (2007)