Semantic Usage Policies for Web Services*

Sebastian Speiser

Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
speiser@kit.edu

Abstract. Web Services provide standardized interfaces for accessing software systems and data sources over the Internet. Semantic descriptions of Web Services help to automate the discovery and invocation of new services and their integration into existing applications. However not all services are freely available for every purpose and not all data is in the public domain. Usage policies describe the terms and conditions under which services and data can be used. Current approaches to semantic Web Service description are mostly focused on functional properties and quality attributes, and do not cover usage policies. We plan to develop a formal language for usage policies with clearly defined semantics that relies on ontologies for representing domain specific terms. We will also extend service discovery and ranking algorithms to incorporate usage policies.

1 Introduction

The classical World Wide Web (WWW) makes services and data available through Web sites targeted at human users. The Web sites are interlinked but the integration of information and services from different sources is a manual effort. The Internet of Services as well as the Semantic Web both go one step further and aim at interoperable and machine-understandable descriptions and interfaces for services and data. It will therefore be easier to combine both types of resources to a network that fulfills a user need which is not satisfiable by any single component. Such networks can be created in various ways, e.g.

- Keyword searches are automatically translated into formal queries over multiple Web resources [1].
- Mashup editors which allow even unskilled users to combine Web services and data sources with minimal time effort [2].
- Creating a business process and orchestrate it with BPEL.

All approaches have in common that suitable services have to be selected that can be part of a network fulfilling the user need.

Users select services based on the net value they expect from the service consumption [3](p. 23ff). The net value is given by subtracting the costs from

* This work was funded by the European project SOA4All and the Karlsruhe Service Research Institute (KSRI)

the experienced gross value. Both can only be determined after service consumption and therefore the selection is based on expected value and costs in which users incorporate their uncertainty about the result. In order to minimize the uncertainties and therefore increase the expected net value for users, two things have to be established: (i) services have to be described in such a way that users know which value and costs to expect, and (ii) these descriptions have to be trustworthy. This thesis deals with the first aspect.

The two most obvious factors influencing the value and cost of a service are its functionality that has to match the user's need and its price. Costs of a service are not restricted to money but also include other obligations and conditions the user has to fulfill, e.g. ensure that the output data is only stored on encrypted hard disks. Expected value is affected by uncertainties about the service experience which can be reduced by statements of the provider about the non-functional properties (NFPs) of the service which comprise the following areas:

– Quality of service (QoS) and data, e.g. availability of the service, are delivered stock prices real-time or delayed?
– Usage policy for input data. Data that is given by the user to the provider in order to consume a service can have restrictions on their usage, e.g. the user may specify that his e-mail address or credit card number are not allowed to be forwarded to other parties. This is also called privacy statement.
– Usage policy for output data, e.g. user is not allowed to display stock prices on public homepage, user has to attribute provider of weather data. We call this also data license.
– Usage policy of service, e.g. mobile data service must not be combined with VoIP service. We denote such policies also as service licenses. They also include customer obligations, e.g. pricing (every call costs 1 Euro, flat-fee for one month of 20 $).

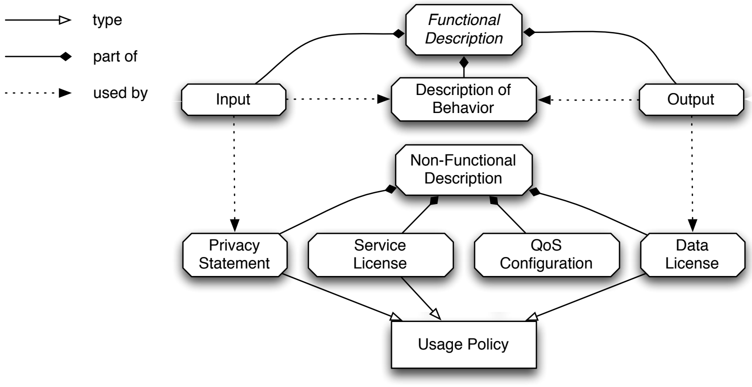

Fig. 1. Functional and Non-Functional Service Descriptions

In Figure 1 we show an overview of the service description, where we count input, output and behavioral description to the functional description and the above-mentioned aspects to the non-functional description.

As we consider a Web scenario with a large number of heterogeneous providers and an even larger number of services, we want to support the user with tools for service selection. Therefore we need formal descriptions of services. This is tackled by various semantic Web service approaches (e.g. WSMO[1], SAWSDL[2], OWL-S[3]). All approaches deal with functionality but non-functional properties are mostly restricted to quality of service and price.

This means that most usage policies are in natural language which requires a significant human effort to evaluate in terms of compatibility with user preferences. Especially in the mashup and keyword search scenario with their high user convenience this effort seems inappropriate. This makes it likely that natural language licenses are just ignored and possibly violated. Formal usage policies representing the licenses and user requirements would allow them to be integrated into service selection tools. In this way adhering to a license can be made easier for the user. As we mentioned above we consider a Web scenario with numerous, heterogeneous providers and therefore we need a license language with clearly defined formal semantics. Our goal is to create such a language and extend service selection and ranking algorithms to include usage policies.

In order to make the language applicable to a wide range of scenarios and different domains of Web resources we want to restrict it to a small core that has a well-defined meaning for permissions, prohibitions, obligations and constraints. The core elements can then be linked to domain-specific ontologies, e.g. for payments or subject and resource identification.

The rest of this paper is structured as follows. Section 2 discusses related work. In Section 3 we outline our planned solution for semantic usage policies and we describe in Section 4 how we want to achieve it. Finally we summarize in Section 5.

2 Related Work

O'Sullivan presents in his thesis a taxonomy of non-functional properties (NFPs) of services [4]. His results are based on the analysis of classic services and Web services from numerous domains. The taxonomy is very comprehensive and deals with various categories of NFPs, such as availability, price, discounts and quality. Usage policy related concepts that allow providers e.g. to forbid commercial usage of their service are not considered. The taxonomy itself is not concerned with the specification of service policies but delivers a vocabulary for common aspects of services.

The taxonomy of O'Sullivan was modeled as WSML ontologies and used to represent NFPs in WSMO [5]. Originally WSMO supported annotations like

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

[2] http://www.w3.org/TR/sawsdl

[3] http://www.w3.org/Submission/OWL-S

creator or title adapted from Dublin Core to describe non-functional properties of ontology elements. With the integration of the O'Sullivan NFP concepts and the extension of WSMO models to include rule-based definitions of NFPs for services, the approach is now able to describe expressive NFP offers in terms of the defined ontologies. Toma et al. present an algorithm that ranks WSMO services according to user specified preferences on NFPs [6]. The ranking is currently limited to NFPs that result in numerical values given the service specification and user provided context and input information.

Web service policy languages can be used to define quality guarantees and also user requirements. For example WS-Policy can specify with assertions which technologies users have to support, e.g. encryption[7]. Assertions are defined in different WS-* standards which cover technical domains but no usage rights and restrictions.

Formalized usage policies exist for digital objects such as e-books and music files. The RDF serialization of the Creative Commons licenses can be used for automatic determination of associated rights [8]. This is used by Google's advanced search[4], where users can specify their desired Creative Commons license and only documents that were annotated correspondingly by their owner are returned. However Creative Commons is mainly suitable for open contents, as the licenses are generally addressed to the public and do not specify rights for single users. Another point is that commercial usage can be allowed but no specific terms (e.g. price) can be expressed.

In the area of digital rights management (DRM), there exist several rights expression languages (REL), e.g. ODRL [9] and XrML [10]. These are XML based languages that allow more complex right specifications than Creative Commons. In theory existing RELs would be a good basis for service usage policies, however the mentioned RELs include language elements that go far beyond rights expression. They handle encryption, media encodings, user authentication and other domain-specific aspects. These are represented as syntactically defined XML elements and a meaning has to be given in the corresponding standard specifications (e.g. ODRL). Also they are lacking formal semantics which leads to ambiguities in scenarios with heterogeneous providers.

This problem is also recognized by Jamkhedkar et al. who present a DRM architecture that separates rights expression from other aspects [11]. They propose that there is a need for a core REL which is based on a mathematical foundation [12]. We share this thought and will further investigate if such a core REL could be reused for service usage policies.

Arnab and Hutchison present LiREL which is a formal REL [13]. They show what the difference to classical access control is and what the resulting requirements for an REL are. LiREL focuses on the expression of rights and requires an external standardized vocabulary for the definition of actions, constraints, etc. This is in contrast to our planned solution which relies on ontologies and ontology mappings rather than trying to build a standard vocabulary that covers all domains and has to be used by all providers and users.

[4] http://www.google.com/advanced_search

Gangadharan et al. developed an ODRL profile for service licenses (ODRL-S) [14]. As a service license they define regulations concerning the use of a service. This is a similar concept to our proposed usage policies, however their approach defines a static vocabulary of actions, including composition or attribution. This regulates general usage of the service but does not incorporate context information, e.g. this means that generally composition can be forbidden but there is no way to specify that composition is allowed with specific other services or providers. Also the policy does not refer to the output data of services. The lack of clear semantics is inherited from ODRL. The interpretation of licenses relies on information such as: composition is a special case of derivation. This hierarchy is however not modeled formally, so that it has to be implemented manually in every program that reasons about service licenses.

In general it can be said that existing rights expression approaches mostly rely on XML for syntax and specify the meaning of elements in human-readable documents. This makes it difficult to automatically match different terminologies that are likely to be used in the Web with its heterogeneous provider structure.

The thesis of Lamparter presents a policy-based approach for service offers, requests and contracts for Web service markets using ontologies to match different provider vocabularies [15]. The approach models functions ranging over non-functional properties of Web services. The functions are used to compactly represent policies that map configurations of NFPs to numerical values. Users specify their valuation of configurations and providers their prices. The policies are then used to find the optimal configuration with respect to the difference of user valuation and price. Lamparter identifies the allowed use of information returned by services as a relevant NFP and exemplifies it by showing how to restrict the disclosure of data. A general model for usage policies is however not given.

Kagal et al. developed the semantic policy language Rei[16]. It models the basic deontic concepts of permissions, prohibitions, obligations and dispensations. Concrete concepts have to be modeled in external domain-specific ontologies. Rei uses OWL as a serialization format but represents rules that have to be interpreted by an external rule engine based on logic programming. The usage of Rei was mainly for access control and we will investigate if it can also applied to usage policies.

The traditional access control model is the access control matrix, which specifies whether a specific action on a specific object by a specific subject is allowed or not [17]. Most access control approaches like role based access control do not explicitly state the matrix but can be reduced to it. Parker et al. break with this model and present the UCON_{ABC} approach for usage control [18] which not only regulates the access to resources but also their further usage [19]. Their model includes obligations on the user side and mutable subject and object attributes that have influence on user rights after the initial access to a resource was granted, e.g. an object can only be printed twice. Parker et al. observe that DRM, privacy policies and access control are developed independently despite their similarity and propose the UCON_{ABC} model as a theoretical foundation for the mentioned wide range of application areas [18]. The model has received broad attention and will be regarded in our formalization of usage policies.

3 Planned Contribution

In this section we list the components that are necessary for a solution that integrates formal usage policies in (semi-)automatic service selection.

The first required component is a language that represents the core notions of usage policies with clearly defined semantics. These notions include the ABC defined by UCON_{ABC} as Authorizations (A), oBligations (B), Conditions (C). The core REL by Jamkhedkar et al., LiREL and ODRL have all slightly different terms such as for example constraints, permissions or prohibitions. We will investigate if these notions are substitutable and which are suitable for our purposes. We envision a solution that is in a similar relationship to ODRL as SAWSDL is to WSDL for Web service description. That means that the structure of the language itself remains the same, but individual elements of a document can reference concepts of an external domain ontology.

These domain ontologies form the second required component of our solution. We plan to reuse results from O'Sullivan [4] and their formalization as ontologies by Toma et al. [5]. For the domain of pricing we will consider the approach by Lamparter [15] which provides an extensive formalization for service pricing models depending on other non-functional attributes.

Being able to describe services is only one part of tool-supported service selection. The other one are algorithms which can be classified in two categories: discovery and ranking. Discovery means that the user formulates his functional and non-functional requirements for a service and the result is a list of matching services. We plan to first extend existing service discovery algorithms to include our proposed model for usage policies. However especially for non-functional properties users often have not only hard requirements but preferences (e.g. a user prefers generally cheaper services but also considers response time). Service ranking denotes the sorting of discovered services according to a preference structure. This structure can either be global for all users and built into the ranking algorithm or specified by each user in a preference formalism. Our goal is to extend existing service ranking algorithms such as the works by Toma et al. [6] and Lamparter [15] to be able to express preferences over usage policies.

In summary our planned contribution is a formal language for usage policies and extensions to service discovery and ranking algorithms that considers usage policy requirements and preferences of users.

4 Work Plan

Until now we have completed the review of the state of the art. We have analyzed different approaches to usage policies and related problems such as access control and DRM. We identified the requirement that the core language should be separated from domain-specific descriptions. We also decided on using ontologies for these descriptions as they support interoperability between different vocabularies likely to be used in a scenario with heterogeneous providers.

As a next step we plan to develop or adapt the core language for usage policies and preferences. As described above we will investigate the possibility to align the non-domain specific parts of ODRL with one of the formal models for usage control languages. The resulting language will integrate references to external ontologies. Based on a selection of Web service and information source licenses we will identify a set of required domain ontologies.

Afterwards we will develop a possibility for users to express their policy needs and preferences. These will serve as inputs to the discovery and ranking algorithms that will be the result of the next stage.

Finally the whole system is evaluated for three aspects: (i) expressivity of the language, (ii) efficiency, and (iii) added value for users. This will be done with the following experiments:

1. Take a list of popular services e.g. from Seekda (http://seekda.com) and distribute them to different users who have to create usage policies using our formalism. By assigning the same license to different users we can afterwards check the equivalence of their results and in this way verify their work. Users have to track which licenses or terms were not expressible in our framework. This experiment evaluates expressivity of the formalism. Additionally the users have to fill out questionnaires about the policy creation process. In this way we want to find out if the approach is practically usable or too complicated.
2. Based on the annotated services we will build a list of user requests and measure the time to complete discovery and ranking with and without usage policies. This will show us if the computational overhead from introducing more complex descriptions is reasonable. The added expressivity of user requirements can lead to a smaller number of services that have to be ranked, which can have a positive effect on the performance.
3. In order to test the added value for users, we will integrate our descriptions and algorithms in a tool, such as for example the keyword-driven search by Tran et al. [1]. With such an implementation we will be able to perform user studies where we can observe if users see a benefit in having tools that are aware of usage policies.

5 Conclusions

The Internet grows more and more into a system that provides us data and services that can easily be accessed and integrated, thanks to Semantic Web and Web Service technology. In such an environment it is natural for common users to build networks of services directly or indirectly and provide them to other users. This large number of providers and services needs formal and interoperable descriptions of services in order to let service location scale. Current approaches to this problem treat functionality and quality aspects. However not all services are freely usable and not all data is in the public domain. The policies of such services also include privacy statements and are currently mostly available in natural language. In this proposal we presented our planned research that includes a formalization of usage policies and their integration into service discovery and ranking algorithms. We believe that the possibility to express such policies will be a motivation for providers to open up their services and data to the Web as it reduces their risk of policy violation.

References

1. Tran, D.T., Wang, H., Rudolph, S., Cimiano, P.: Top-k Exploration of Query Candidates for Efficient Keyword Search on Graph-Shaped (RDF) Data. In: Proceedings of the 25th International Conference on Data Engineering (ICDE). (2009)
2. Pautasso, C., Frisoni, M.: The Mashup Atelier. In: Second International Workshop on Web APIs and Services Mashups (Mashups'08) at ICSOC. (2008)
3. Ng, I.C.: The Pricing and Revenue Management of Services: a strategic approach. Routledge Advances in Management and Business Studies. (2007)
4. O'Sullivan, J.: Towards a precise understanding of service properties. PhD thesis, Queensland University of Technology (2006)
5. Toma, I., Foxvog, D.: Non-Functional Properties in Web Services. Technical Report D28.4 v0.1, WSMO Working Draft (October 2006)
6. Toma, I., Roman, D., Fensel, D., Sapkota, B., Gómez, J.M.: A Multi-criteria Service Ranking Approach Based on Non-Functional Properties Rules Evaluation. In: ICSOC. (2007) 435–441
7. Vedamuthu, A., Orchard, D., Hirsch, F., Hondo, M., Yendluri, P., Bubez, T., Yacinalp, U.: Web Services Policy 1.5 - Framework. Technical report
8. Abelson, H., Adida, B., Linksvayer, M., Yergler, N.: ccREL: The Creative Commons Rights Expression Language. Technical report, Creative Commons (2008)
9. Iannella, R.: Open Digital Rights Language (ODRL) Version 1.1. W3C Note (2002)
10. Wang, X., Lao, G., Demartini, T., Reddy, H., Nguyen, M., Valenzuela, E.: XrML - eXtensible rights Markup Language. In: ACM workshop on XML security. (2002)
11. Jamkhedkar, P.A., Heileman, G.L.: Digital rights management architectures. Comput. Electr. Eng. **35**(2) (2009) 376–394
12. Jamkhedkar, P.A., Heileman, G.L.: A formal conceptual model for rights. In: Digital Rights Management Workshop. (2008) 29–38
13. Arnab, A., Hutchison, A.: Persistent Access Control: A Formal Model for DRM. In: ACM workshop on Digital Rights Management. (2007) 41–53
14. Gangadharan, G.R., D'Andrea, V., Weiss, M.: Service Licensing Composition and Compatibility Analysis. Int. J. Cooperative Inf. Syst. **17**(3) (2008) 301–317
15. Lamparter, S.: Policy-based Contracting in Semantic Web Service Markets. PhD thesis, Universität Karlsruhe (TH), Institut AIFB (2007)
16. Kagal, L., Finin, T., Joshi, A.: A policy language for a pervasive computing environment. In: Policies for Distributed Systems and Networks. (June 2003) 63–74
17. Lampson, B.W.: Protection. In: Proc. Firth Princeton Symposium on Information Sciences and Systems, Princeton University (March 1971) pp. 437–443 reprinted in Operating Systems Review, 8,1, January 1974, pp. 18 – 24.
18. Park, J., Sandhu, R.S.: The $\mathrm{UCON}_{\mathrm{ABC}}$ usage control model. ACM Trans. Inf. Syst. Secur. **7**(1) (2004) 128–174
19. Park, J., Sandhu, R.: Towards usage control models: beyond traditional access control. In: ACM Symp. on Access Control Models and Technologies (SACMAT'02)