ENHANCING WEB SERVICE DESCRIPTIONS WITH CONTEXT FUNCTIONS

Georgia M. Kapitsaki
Department of Computer Science, University of Cyprus, 75 Kallipoleos Street, P.O. Box 20537, CY-1678, Nicosia,Cyprus

Keywords: Web Services, Semantic Web, Context, Context Dependencies.

Abstract: Web Service reuse is embraced both by the industry and the academia. Especially in the field of context-awareness, where Web Services are usually employed as building blocks for larger applications, the existence of adequate service descriptions is vital. The semantic Web technologies have assisted in providing richer descriptions of resources However, elaborated constructs that allow the potential user to understand the behaviour and relation of the Web Service to context information are missing. In the current paper context-related information that should form part of the service description is presented in three categories: context dependencies, context offering and context special handling. The categories are analyzed to their ingredient elements and incorporated in service descriptions as en extension to the OWL-S ontology for Web Services demonstrating the preliminary use of the proposed approach.

1 INTRODUCTION

Reuse at different levels is a main aspect of software engineering although open research issues remain (Frakes, 2005). Also in the field of services reuse is widely promoted. Especially Web Services (WSs) are often seen and exploited as building blocks for larger applications that may target desktop, Web or mobile environments depending on the purpose of use. Many large software vendors are exploiting the Web Service technology either in the form of in-house developed solutions to be integrated in the company's products or as reusable software components that can be acquired through third party providers as free and open source or purchased solutions (McKinsey, 2007).

Web Services are also widely employed in the field of context-aware services and applications (Dey and Abowd, 2000). Different frameworks that link WSs and context can be found in the literature (Keidl and Kemper, 2004). Similarly to the way it is treated in other domains, the importance of context information in Web Services is not to be neglected. Context can assist end-users and software engineers to infer on the potential use of the service, i.e., circumstances under which the service can be exploited. For instance, a service available in the United States may require the USD as currency for any payment performed and can, thus, not be used or combined with another service targeting European countries.

Nevertheless, the above is not the only case where the existence of context-related information is required in WSs to assist their proper use. It is usual that a WS be dependent on context information. For example, a WS requiring as an input parameter the value of the current temperature depends on context information related to the current weather conditions. This needs to be specified in the service description in order to make: 1) potential end-users of the WS aware that the temperature information might be requested from them and 2) potential engineers who want to use the service alone or in combination with other ones aware that it is their responsibility to inject the temperature information to the service, once they decide to use it. Similarly, a WS may be equipped with mechanisms that retrieve context information and, thus, be offering context as output to its operations instead of requiring it, or include special context handling in its functionality that should be again communicated to prospective users.

Although the Web Service Definition Language (WSDL) specification for the description of WS interfaces, has been complemented with semantics through semantic Web technologies, questions that evolve around context dependencies, offerings and

M. Kapitsaki G..
ENHANCING WEB SERVICE DESCRIPTIONS WITH CONTEXT FUNCTIONS.
DOI: 10.5220/0003961102130218
In *Proceedings of the 8th International Conference on Web Information Systems and Technologies* (WEBIST-2012), pages 213-218
ISBN: 978-989-8565-08-2
Copyright © 2012 SCITEPRESS (Science and Technology Publications, Lda.)

special handling remain unanswered in the solutions available in the bibliography. The WS description should incorporate answers to questions such as: "*What kind of context the service depends on*", "*What type of context the service offers*", "*How does the service consider context in its functionality*". The current paper presents the ongoing work towards integrating answers to the above questions in the service description. Specifically, it analyzes the elements of the answers to the three questions and proposes an extension to the OWL-S semantic markup for Web Services (Martin, et al., 2007) with context-related elements. In the proposed approach context information is regarded primarily from the software engineer's view. The main aim is to assist in reusing available services

The rest of the paper is structured as follows: section 2 presents the related literature that motivated this work. Section 3 analyzes the identified context categories and their application use, whereas section 4 introduces OWL-S-CONTX, the proposed context-related extension to OWL-S. Finally, Section 5 concludes the paper.

2 RELATED WORK AND MOTIVATION

Context-awareness is an issue that has gained tremendous interest from the research community in the latest years targeting in many cases pervasive and mobile computing systems. System architecture solutions that focus on context-awareness in Web services addressing how context information can be employed during WS execution are also available (Keidl andKemper, 2004; Kapitsaki, Kateros and Venieris, 2008).

One important issue in context-aware environments concerns the development of context-aware applications by reusing already existing components either as stand-alone and individual services or as building blocks for larger applications (Claro, Albers, and Hao, 2006). In order to assist this process the principles of the Semantic Web, where resources form part of ontologies using languages such as Resource Description Framework (RDF) and Web Ontology Language (OWL), are usually employed. The use of ontologies is common in domains, such as pervasive and ubiquitous environments (Chen et al., 2004; Reichle et al., 2008). An ontology specific to Web Services has been recommended under OWL-S (Martin, et al., 2007). OWL-S consists of three parts:

1. *Service Profile:* describes the functionality that the service performs containing information on inputs required by the service and the generated output along with information on preconditions for running the service and side effects from its execution.
2. *Service Process:* contains information about the service processes showing ways of how a client may interact with the WS.
3. *Service Grounding:* provides details of how to access the service and is related to the traditional WSDL descriptor.

Existing literature works have addressed the issue of adding context-related information in WS descriptions. Mrissa et al. (2006) have added context information as an extension to WSDL making input and output messages more specific (e.g., to state that the expected currency in an input message is the Singapore dollar). Based on this description a semantic mediator is responsible for performing any necessary transformations to ensure that composition between services that do not share the same context is feasible. In (Madnick, Zhu and Fan, 2009) context data described in an ontology are added as annotations in WSDL by extending the Semantic Annotation for WSDL and XML Schema standard.

An approach that uses OWL-S for describing context-aware services as part of a wider system, the *myCampus* Semantic Web environment, has been proposed by Sheshagiri, Sadeh and Gandon (2004), However, no context-related extension is proposed by the authors. In another work context information is added as an extension to the Service Profile of OWL-S (Suraci, Mignanti and Aiuto, 2007). A context attribute that points to a Uniform Resource Identifier (URI), where the context is stored in OWL format, is proposed. This information is maintained in the Context Manager who is responsible for administrating context information. An extension to OWL-S can also be found in (Nayar, Kirsch-Pinheiro and Souveyet, 2011). In this work the authors propose an extension consisting of three parts: one as a new parameter of the Service Profile that describes the intention associated with the service (related to user's goals), one as a new context attribute in the Profile acting as in (Suraci, Mignanti and Aiuto, 2007) and one in the Service Process that captures service variability.

A common observation is that none of the above works considers the inclusion of context-related functions in WS descriptions in the way it is proposed in the current paper. They focus either on the context of the service execution or the service requester. This information is important and should

not be neglected, but leaves out the important aspects of context dependency, offering and special handling. On the one hand, Web Services should include information on the context information they are based on for their functionality and whether they have an internal mechanism for acquiring this data; an aspect relevant to context-aware Web Services. On the other hand, if they can offer this context data also as output, they need to inform potential users. All these perspectives can assist in using the service correctly and even in reusing the service for building larger applications. The specific uses of the proposed approach are briefly discussed in a subsequent section.

3 CONTEXT ENHANCEMENT

3.1 Context Categories and Fields

Based on an extensive study of the existing literature works on context-awareness and Web Services, the context information for the enrichment of the WS descriptions should fall into the following three categories.

1. ***What context the service needs***: in the case where the WS requires context information for its functionality this needs to be specified in the corresponding description. We distinguish between two context dependency types: 1) *parameter injection* and 2) *operation selection* that correspond to the first two context adaptation cases for Web Services discussed in (Kapitsaki, Kateros and Venieris, 2008).
2. ***What context the service offers***: if the WS provides context information by acting as a wrapper for a context source of a raw nature more details about the kind of information need to be given (namely *context offering*).
3. ***What additional context-related function is required***: this category refers to the case where the WS contains context-related functionality internally or if its operations return response messages that can be context-treated. Specifically, in the first type (i.e., *internal functionality*) the WS retrieves or accesses context information as part of its operations without the requester explicitly asking for it, whereas the second type (i.e., *response manipulation*) informs about potential changes that can be performed on the service response based on context conditions and corresponds to the third adaptation case of (Kapitsaki, Kateros and Venieris, 2008).

For instance, in the case of business Web Services, which require context information for their proper functionality (e.g., a FlightReservation service requiring location information on the point of departure), the first category is utilized. This information is related to the context dependencies of the services. On the other hand, it is usual to have Web Services that have access to context information via appropriate context sources (e.g., environment sensors, database systems). These services, therefore, expose this information and can act as context retrieval points hiding any implementation details from the infrastructure they depend on for the actual retrieval process. The third category provides more details on context-related functions that should be communicated to the potential service user. Please note that a WS may be related with more than one category or may contain multiple types of the same category.

In order to demonstrate the necessity of importing the proposed aspects in the WS description, two example services are provided. The first one refers to a Web Service that provides information on theaters nearby (a similar service can be found in the MovieInformation WS offered by ignyte.com). The *TheatersFinder* WS depends on the current user location and the weather conditions by offering two operations: one for retrieving open air theaters and one for retrieving indoor theaters. Since the service delivers a list of theater options, the requester's preferences on kind of plays can be considered. It is, therefore, useful to inform the potential software engineers that the response list may be ordered according to these preferences. This additional information can be captured in three cases of the aforementioned categories: one parameter injection, one operation selection and one response manipulation respectively. The second example is taken from the *IP2Location* WS provided by ws.fraudlabs.com that offers location information by the IP address. Its detailed description can be obtained from the service provider. All context enhancements associated with the services are depicted in Figure 1.

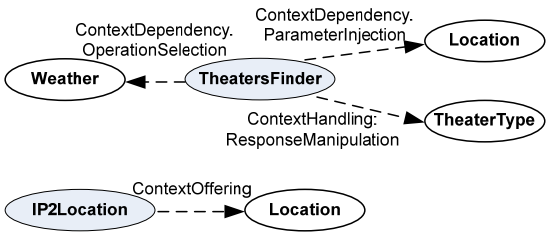

Figure 1: TheaterFinder and IP2Location examples.

Detailed fields are linked to each category depending on its specific role. These are presented in Tables 1 to 4. Table 1 refers to common fields necessary in all categories.

Table 1: Common fields.

ContextName	The context name of the dependency (as presented in Figure 1).
Aggregation Level	Specifies whether the related context is atomic or composite (Sheng and Benatallah, 2005).
Context Ingredients	Links to the atomic contexts that make up the composite information (applicable to composite context information).
Similar Concepts	Defines a list of concepts that have a similar meaning to the main context (e.g., same concept, hyponyms, meronyms etc.) as defined in lexical databases (i.e., WordNet) or categories taken from classification systems (e.g., UNSPSC).
Related Taxonomy	Specifies a URI pointing to a more detailed taxonomy depicting relations to other concepts (if available).
Related Model	Specifies a URI pointing to the context model related to the context information in question (if available).

The optional *Related Model* field refers to a URI pointing to a detailed context information model. This is proposed in (Suraci, Mignanti and Aiuto, 2007), where it relates to the context circumstances under which the service should be executed. However, the URI proposed differs in the sense that it adheres to the context category the information refers to. It may link to an ontology specifying the context information, such as the works presented in (Bettini et al., 2010), to a Unified Modelling Language (UML) model describing context, such as in approaches like (Ayed and Berbers, 2006), or to a Domain Specific Language (DSL). Placing this information in the description as an external URI adds flexibility to the proposed approach, since the context model may be updated with a newer version or by an alternate representation type without affecting the main service descriptor found in the OWL-S extension.

Table 2: Context Dependency fields and descriptions.

Types	**Context-related fields**	
Parameter injection	*Parameter Name*	Name of the context-related input parameter.
Operation selection	*Operation Names*	Specifies the list of WS operations whose selection depends on context data.

Table 3: Context Offering fields and descriptions.

Retrieval Mechanism Type	Specifies the source of the context data (e.g., Database Management System, sensor, GPS).
Context Source Locator	Gives more information on the context source depending on its type. For instance, for context data retrieved from a database this may refer to the database access information, the table and the columns where the context information can be found.
Alternate Source	If more than one context sources are considered (in case the first one is unavailable) this is specified in the same form as in the previous field.
Alternate Service	If an alternate WS offering similar context data is available it is specified through a URI pointing to its specification.

Table 4: Context Handling fields and descriptions.

Types	**Context-related fields**	
Response manipulation	*Manipulation Type*	Specifies the type of response manipulation related to context information (i.e., filtering, ordering, adding, specific).
	Related Link	Specifies an external service that may be exploited in order to perform the specified manipulation (if available).
Internal functionality	*Retrieval Mechanism Type, Context Source Locator*	<Same as in the Context Offering fields>
	Access Required	Specifies whether access to additional information is required, such as user-related data.
	AccessTo	Specifies the information the service needs access to be granted for (e.g., user email client for user-related data, mobile device sensors for environment-related information).

Table 4: Context Handling fields and descriptions (cont.).

	Alternate Mechanism	Specifies any alternate retrieval mechanisms that the service considers in case the first one is unavailable.

3.2 Application Uses

The proposed description fields can be exploited in Web Services for different purposes. Sample uses can be found in:

• ***Web service searching***: software engineers may need to discover a specific service that treats internally or offers access to context information they need in their application.

• ***Web service composition***: an application consisting of multiple WSs linked with context may be developed using existing services as building blocks. For instance, a WS may depend on context information and another WS may offer the necessary information.

• ***Web Service compatibility check***: additional applications are feasible, when additional properties, such as security and privacy, are considered. For instance, if a specific WS requires access to the user's location but the software engineer does not want to allow this access she/he is informed on this from the WS description and may decide to use an alternate choice.

4 OWL-S-CONTX

The context enhancement is included as an extension to a semantic Web Service description and specifically as an extension to OWL-S, referred to as *OWL-S-CONTX*. Each part of the OWL-S ontology has a specific role and, therefore, the information on context use has been added in different parts. Information on the input parameters and the generated output constitute subclasses of the service parameters in OWL-S. Context-related information on the input and output is specified in the categories of parameter injection, response manipulation and context offering: parameter injection has been added as a subclass to the *&process;#Input*, class (renamed to *ContextInjection* for semantic purposes), whereas response manipulation and context offering have been added as subclasses to the *&process;#Output* resource of OWL-S.

Operation selection and internal functionality are related to the service process part of OWL-S: internal functionality has been inserted as the object property *ContextFunctionality* to the *&process;#SimpleProcess* class (which provides more information on its functionality than *&process;#AtomicProcess*), whereas operation selection as a subclass to the *Choice* control structure class contained in the *CompositeProcess* class union.

The above additions are depicted in a graphical representation of the Service Process based on OWL-S version 1.2 (Figure 2). The elements in grey correspond to the additions.

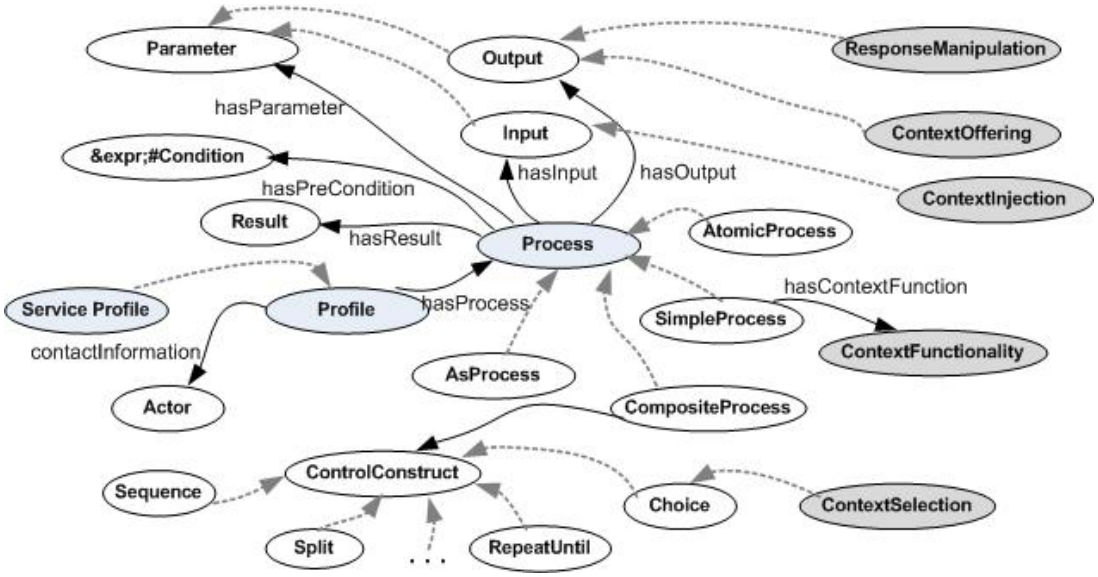

Figure 2: Context-related extension to OWL-S.

5 CONCLUSIONS

In this work the initial concept of enhancing the OWL-S ontology with information on context dependencies, context offerings and special context handling related with a Web Service has been provided. Such an addition can prove useful for searching, composing or matchmaking WSs in different use cases. The introduced categories and their respective fields have been analyzed and indicated in *OWL-S-CONTX* as extension points to the OWL-S ontology. Currently, we are in the process of investigating realistic examples by exploiting semantic Web Services test collections (semwebcentral.org/projects /owls-tc) for the evaluation of our work. As future work it will be interesting to implement a reasoner that exploits the descriptions in order to draw useful conclusions for the services and study the combination of the proposed scheme with existing works that integrate context information in the description of OWL-S.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

REFERENCES

Ayed, D., Berbers, Y., 2006. UML profile for the design of a platform-independent context-aware applications, In *MODDM'06, 1st Workshop on Model Driven Development for Middleware*, Melbourne, Australia, ACM New York, pp. 1-5.

Bettini, C., Brdiczka, O, Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D., 2010. A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, vol. 6, 2, pp. 161-180.

Chen, H., Perich, F., Finin, T., Joshi, A., 2004. SOUPA: standard ontology for ubiquitous and pervasive applications, In *MOBIQUITOUS'04, First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services*, pp. 258- 267.

Claro, D. B., Albers, P., Hao, J.-K., 2006. Web Services Composition, *Semantic Web Services, Processes and Applications, Semantic Web and Beyond*, vol. 3, Springer US, pp. 195-225.

Dey, K., Abowd, G. D. (2000) 'Towards a Better Understanding of Context and Context-Awareness'. In *CHI'00, Workshop on The What, Who, Where, When, and How of Context-Awareness, as part of the 2000 Conference on Human Factors in Computing Systems*. ACM Press.

Frakes, W. B., Kang, K., 2005. Software reuse research: status and future, *IEEE Transactions on Software Engineering*, vol.31, no.7, pp. 529-536.

Kapitsaki, G. M., Kateros, D. A., Venieris, I. S., 2008. Architecture for Provision of Context-aware Web Applications based on Web Services, In *PIMRC'08, IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1-5.

Keidl, M., Kemper, A., 2004. Towards Context-Aware Adaptable Web Services, In *WWW´04, 13th international World Wide Web conference*, ACM, New York, pp. 55-65.

Li, X., Madnick, S., Zhu, H., Fan, Y. 2009. An Approach to Composing Web Services with Context Heterogeneity, In *ICWS '09, 7th IEEE International Conference on Web Services*, pp: 695-702.

Martin, D., Burstein, M., Mcdermott, D., Mcilraith, S., Paolucci, M., Sycara, K., Mcguinness, D. L., Sirin, E., Srinivasan, N., 2007. Bringing Semantics to Web Services with OWL-S. *World Wide Web*, vol. 10, no. 3, pp. 243-277.

McKinsey Business Technology Office, Media & Entertainment Practice, 2007. How businesses are using Web 2.0: A McKinsey Global Survey, McKinsey Quarterly.

Mrissa, M., Ghedira, C., Benslimane, D., Maamar, Z., 2006. Context and Semantic Composition of Web Services, In DEXA'06, 17th International Conference on Database and Expert Systems Applications, Springer, pp. 266-275.

Nayar, S., Kirsch-Pinheiro, M., Souveyet, C., 2011. Bringing Context to Intentional Services, In *International Conferences on Advanced Service Computing*, pp. 118-123.

Reichle, R., Wagner, M., Khan, M. U., Geihs, K., Lorenzo, J., Valla, M., Fra, C., Paspallis, N., Papadopoulos. G. A., 2008. A comprehensive context modeling framework for pervasive computing systems. In *DAIS'08, 8th IFIP WG 6.1 international conference on Distributed applications and interoperable systems*, Springer-Verlag, pp. 281-295.

Sheng, Q. Z., Benatallah, B., 2005. ContextUML: a UML-based modeling language for model-driven development of context-aware Web services, In ICMB'05, International Conference on Mobile Business, pp. 206- 212.

Sheshagiri, M., Sadeh, N. M., Gandon, F., 2004. Using Semantic Web Services for Context-Aware Mobile Applications, In *MobiSys'04, Workshop on Context Awareness*.

Suraci, V., Mignanti, S., Aiuto, A., 2007. Context-aware Semantic Service Discovery, In *16th IST Mobile and Wireless Communications Summit*, pp.1-5.