Chapter X
The Process of Semantic Annotation of Web Services

Christoph Ringelstein
University of Koblenz-Landau, Germany

Thomas Franz
University of Koblenz-Landau, Germany

Steffen Staab
University of Koblenz-Landau, Germany

ABSTRACT

Web services are software components that are—in general—distributed over multiple organizations. They provide functionality without showing implementation details for the purpose of abstracting from implementation as well as for the purpose of hiding private, that is, organization-internal, processes. Nevertheless, to use a Web service one must know some of its details, that is, what it does, what it requires, what it assumes, what it achieves, and to some extent, how it achieves its purpose. The different Web service standards, frequently summarized as WS*, allow Web services to be specified with descriptions of such details. In this chapter, we argue that one should go beyond WS* and that it is preferable to provide semantic descriptions, that is, specifications that can be understood and correctly interpreted by machines. Thereby, the particular focus of this contribution lies in analyzing the process of semantic annotation, that is, the process of deriving semantic descriptions from lower level specifications, implementations and contextual descriptions. Hence, the concern of this chapter is really orthogonal to most other work which equates Web service annotation with Web service specification. We illustrate here that this is not the case.
INTRODUCTION

Web services are software components that are accessible as Web resources in order to be reused by other Web services or software. Hence, they function as middleware connecting different parties such as companies or organizations distributed over the Web. Thereby, a party providing a service may not be interested in exhibiting their organization-internal processes to the outside world. A second party consuming such a service may not be interested in analyzing a given Web service in order to be able to use it. Therefore, an abstracting description of a Web service is necessary to allow for its effective provisioning and use.

The description of a Web service needs to include some bare technical information in order that it can be used. This includes:

1. **What it does**;
2. **How to invoke** the Web service (i.e., the used communication protocol);
3. **What parameters** to provide to the Web service (i.e., its signature);
4. **which protocol** should be followed when using the Web service (e.g., “register user; then, book journey!”).

A Web service user may have some expectations about a service’s

5. **properties**, concerning, for example, security means (e.g., “always use secure communication for my bank account information!”).

This list is by no means complete. One may require further technical descriptions (e.g., transactions) or legal aspects (e.g., contractual issues). However, it indicates that two parties that plan to cooperate via a Web service need specifications of the service allowing them to share and exploit technical and nontechnical descriptions.

Such sharing of descriptions is extremely difficult if the means of sharing are not standardized and descriptions boil down to verbose textual documents. To simplify use of Web services, several properties of Web services are described following standardized XML documents (e.g., SOAP, UDDI, WSDL, WS-Security), for further properties standardization activities for XML descriptions are underway (e.g., WS-Transaction, WS-BPEL, WS-Policy) and for yet others there is a discussion whether standardization should be initiated.¹

Exploitation of Web service descriptions may occur in various ways. Technical and nontechnical descriptions may be used (1) to select a service, (2) to compose it with other Web services, or (3) to derive relevant properties about a composition of Web services (e.g., combined cost or validity for a given specification).

In this chapter, we consider the process of provisioning data about a Web service to constitute a specification of the Web service. At this point, the question arises how a machine may attribute machine-understandable meaning to this metadata. The XML standards (WS*) listed above lack the formal semantics to achieve common interpretation and interoperability of Web service annotations. Therefore, we argue for the use of ontologies for giving a formal semantics to Web service annotations, that is, we argue in favor of semantic Web service annotations. A Web service ontology defines general concepts such as service or operation as well as relations that exist between such concepts. The metadata describing a Web service can instantiate concepts of the ontology (Patil, Oundhakar, Sheth, Verman, & Kunal, 2004). This connection supports Web service developers to understand and compare the metadata of different services described by the same or a similar ontology. Consequently, ontology-based Web service annotation leverages the use, reuse and verification of Web services.
Related Content

Pricing Utility Computing Services
www.igi-global.com/article/pricing-utility-computing-services/3101?camid=4v1a

Stakeholder Identification and Analysis for Service Lifecycle Management
www.igi-global.com/article/stakeholder-identification-and-analysis-for-service-lifecycle-management/110873?camid=4v1a

Need and Possible Criteria for Evaluating the Effectiveness of Computer-Mediated Communication
www.igi-global.com/chapter/need-possible-criteria-evaluating-effectiveness/53281?camid=4v1a

bpCMon: A Rule-Based Monitoring Framework for Business Processes Compliance
www.igi-global.com/article/bpcmon/181301?camid=4v1a