Semantic Web Service Architecture — Evolving Web Service Standards toward the Semantic Web

Tanja Sollazzo[1], **Siegfried Handschuh**[1], **Steffen Staab**[1], **Martin Frank**[2], **Nenad Stojanovic**[1]
[1]Institute AIFB, University of Karlsruhe, 76128 Karlsruhe, Germany
tanja@sollazzo.de, {sha, sst, nst}@aifb.uni-karlsruhe.de
[2]Information Sciences Institute of the University of Southern California, Marina del Rey, California 90292
frank@isi.edu

Abstract

The importance of Web services has been recognized and widely accepted by industry and academic research. However, the two worlds have proposed solutions that progress along different dimensions. Academic research has been, mostly concerned with expressiveness of service descriptions, while industry has focused on modularization of service layers — mostly for usability in the short term. This paper is concerned with merging these two streams of progress. Our point of departure is the current proposal by IBM. Its proposal is extended by Semantic Web technologies such that a smooth evolution from Web services in the current Web to Web services in the Semantic Web appears possible and — in fact — highly desirable. As a showcase we describe SWOBIS, an ontology-compatible registry for software tools, that represents a first step towards developing a search engine for Web services based on Semantic Web technologies.

Introduction

Web services are "self-contained, self-describing modular applications" (Martin 2001). They constitute software modules that "describe a collection of operations that are network-accessible through standardized XML messaging" (Kreger 2001, p. 6). With OAP, UDDI, WSDL, and .NET industry has made a bold move and started initiatives that target the potential benefits of Web services. The focus of the initiatives was an evolutionary step from current Web technology toward a technology for Web services. Key concerns of the initiatives were, e.g., short-term applicability or scalability. This implies that the corresponding Web service architectures build on little really new technology inside, e.g. they use standardized taxonomies and vocabularies that exhibit little flexibility and expressiveness and that restrict the usability of Web services mostly to human users rather than machine agents. For the latter one would need, e.g., Web service description languages that support semi-structured data, constraints, types and inheritance.

In contrast to the industry point of view, academic research has investigated languages that fulfill exactly these needs (Horrocks *et al.* 2001; Ankolenkar *et al.* 2001; Fensel *et al.* 1999) offering extensible ontology frameworks and layering of languages in the Semantic Web. To the detriment of the latter community the adoption of their schemes into industry (quasi-)standards for Web services is far from trivial, because there is no coherent architecture of immediate practical benefit.

The core idea of this paper is to present an architecture that combines the two worlds and their potential benefits. The benefits of the integration include increased visibility of Web services because open ontology frameworks allow for semantically expressive advertising on the Web that may be found by Web crawlers. They include better usability because of more expressive Web service descriptions. They include a smooth evolution from Web services for human users such as targeted by current industry (quasi-)standards toward Web services for personalized machine agents that assist the user.

The structure of this paper is as follows. First, we sketch the general model of the Web service setting, of the architecture of IBM in particular. We consider the latter the most elaborate and the best described industry quasi-standard for Web Services so far. Second, we analyse assumptions of this (and related) architecture(s), describing several parameters which may be varied to turn the "traditional" view into a Semantic Web view. Third, we outline the current description of DAML-S (Ankolenkar *et al.* 2001). Fourth, we critically evaluate the achievements of DAML-S. Fifth, we describe the integrated architecture that we propose. Finally, we sketch SWOBIS, an ontology-based registry for software tools that we are currently developing into the direction of our proposed integrated architecture.

IBM Web Service Architecture

The typical procedure in a Web service setting is the following: A Web service provider offers services on the Web. He may choose to register her service at an online registry of a service broker (Trastour & Bartolini 2001, p.2/3). The registry publishes and locates services. To allow for service discovery, the registry also provides standardized description facilities, e.g. taxonomies that allow the description of, *(i)*, the functionality of a service, *(ii)*, its service provider, *(iii)*, how to access and interact with the service. The corresponding information about a particular service is registered by the provider at a broker.

The (human) requestor searches for a service at the reg-