

Autonomous Semantic Web Services

Katia P. Sycara

School of Computer Science
Carnegie Mellon University
katia@cs.cmu.edu

<http://www.cs.cmu.edu/~softagents>

Abstract. Web Services are defining a new paradigm for the Web in which a network of computer programs become the consumers of information. The growing infrastructure for Web Services is based on SOAP and WSDL and assumes XML as unifying language to guarantee Web Services interoperability. XML guarantees syntactic interoperability by providing a standard for a common syntax that is shared across the Web, with the result that Web Services can parse each other's messages, verify whether they adhere to the expected formats, and locate each piece of information within the message. Unfortunately, current Web Services do not have any means to extract the meaning of the messages exchanged. The Web Services understand the structure of each other's message, but do not understand the content of those messages. This limitation requires programmers to hard-code Web Services with information about their interaction partners, the messages that they exchange and the interpretation of those messages; moreover, current Web Services cannot reconfigure dynamically to adapt to changes without direct human intervention.

Ideally, we would like Web Services to act autonomously. Web Services should be able to register autonomously with infrastructure Registries such as UDDI, use the infrastructure Registries to locate providers of services that they need, and they should be able to transact with these Web Services sending them information formatted in a way that they can understand. Correspondingly, Web Services should be able to interpret the information that they receive. Autonomous Web Services not only minimize the human intervention by automating interaction with other Web Services, allowing programmers to concentrate on application development, but also are able to recover from failures more efficiently by automatically reconfiguring their interaction patterns. For example, if one of their partners fails, or becomes unreliable, they may be able to find other more reliable partners; similarly, if a new, cheaper, or higher quality provider comes on line, Web Services should be able to dynamically find the new partner and interact with it in peer to peer fashion. In order to find partners, Autonomous Web Services need to be able to semantically describe and automatically register their own capabilities with public registries. Semantically meaningful capability descriptions are crucial to automated partner discovery so that discovery is based on the service semantics of the partners, rather than on matching keywords, such as service name or name of the company that deploys the Service. In addition, a Web Service should have information on how to interact with another Web Service, which means that it should know the interaction protocol and binding information. Most crucially, this information should allow the re-

questing Web Services as well as the provider to decode the information exchanged, so it should specify not only the format of the messages to exchange or the remote procedures to call, but also the semantic type of the information.

This view is embraced by DAML-S (The Darpa Agent Markup Language for Services) which defines a DAML ontology for the description of Web Services. DAML-S attempts to bridge the gap between an infrastructure of Web Services based essentially on WSDL and SOAP, and the Semantic Web. In other words, DAML-S bridges the gap between the specification of the format of the information to be exchanged and the specification of its meaning. DAML-S assumes that a transaction between Web Services involves at least three parties: a provider of the service, a requester of the service, and some infrastructure component such as UDDI that facilitates the location of the provider and possibly facilitates the transaction between provider and requester. Furthermore, DAML-S allows for a flexible assignment of roles in which a Web Service can be both a provider in one transaction and a requester in another, and it also allows for a role switch within the same transaction.

DAML-S has three modules that provide a description of different aspects of Web Services. The first one, called Profile, is an abstract description of the Web Service described as a transformation from the inputs the Web Service requires to the outputs it generates. The second module is the Process Model that characterizes the Web Service, specifically, it describes the interaction flow with the Web Service. The third module, called Grounding, specifies how the inputs/outputs of each step are mapped on WSDL specifications of messages that the two Web Services exchange.

In this talk, I will describe in depth the vision of Autonomous Semantic Web Services, the requirements it imposes, and the research challenges it presents. I will also present DAML-S, contrast it with current industrial proposed standards for Web Services and present a roadmap to get us from today's Web service technology to the future of Autonomous Semantic Web Services.