

The Challenge of Semantic Web Services

I'm responding to the essays on Semantic Web services and technologies in your Sept./Oct. 2007, Nov./Dec. 2007, and Jan./Feb. 2008 Trends & Controversies department.

In "Semantic Technologies: Realizing the Services Vision" (Sept./Oct.), Michael Brodie wrote, "How will services discover the right services that meet specific requirements in such an environment? We're far from that scale at the moment." In the introduction in the same issue, David Martin and John Domingue wrote that the "central theme" of Semantic Web services (SWS) is the use of services—for example, as in the related research on OWL-S and WSMO (Web Service Modeling Ontology) and the discussions on the composite services and business processes in the articles by Frank Leymann (Sept./Oct.), Steve Battle (Nov./Dec.), and Katia Sycara (Nov./Dec.).

My question is, even if we're far from discovering the right services at the moment, how can we "use" them in OWL-S and WSMO? Ian Foster also commented in *Science* ("Service-Oriented Science," 6 May 2005) that "[Web] services have little value if others cannot discover, access, and make sense of them." For this reason, I think the "central theme" of SWS, currently, is how to discover the right services, rather than how to use them, because we are far from discovering them.

Unfortunately, SAWSDL (Semantic Annotations for Web Services Description Language) isn't the right way for SWS, given that

- the same service provider can use different service interfaces defined in WSDL documents when developing the same kind of service,
- different service providers can generate different service interfaces defined in WSDL documents when developing the same kind of service, and
- the same service interface definition can be used to implement different services and functions.

SAWSDDL annotates the service interface definition. But the question is, what are we looking for, a kind of service or a kind of service interface? In software engineering, one specification can be implemented in different ways;

in a Web service, the same service can be implemented by different service interfaces. The semantics of Web services is in the specification, not the interface, so the semantics is neutral and independent from the service interface definition.

At this moment, SOA (Service-Oriented Architecture) isn't complete because there's no service registry after the UDDI (universal description, discovery, and integration) initiatives terminated in early 2006. If we can't find the right services, we can't use them. SAWSDL might lead to chaos rather than to a solution for the SWS community because it targets the wrong object. And one last goal of SWS is to enable the dynamic service invocation, but OWL-S, WSMO, and SAWSDL can't realize such a goal.

SWS research and development hasn't incorporated classic theories such as systems theory and cybernetics. The key of systems theory for SWS research and service-oriented computing is to treat all SOA components as a whole system that has comprehensive goals (that is, the goals of SWS). Dealing with partial components or realizing partial goals of the whole system might not help fulfill that system's comprehensive goals. Cybernetics for SOA systems must focus on identifying the source of the chaos and then controlling and minimizing that chaos by examining the feedbacks between the system components.

To manage the growth of knowledge in such an open, huge, and dynamic SOA system, we must incorporate cognitive psychology (how human intelligence grows) into the design of SOA architecture (how machines' AI grows as human beings design them). Because humans inevitably experience cognitive disorder as their intelligence grows, we can also expect intelligent machines or software agents to encounter such disorder as their knowledge or AI increases.

Xuan Shi
Georgia Tech
xuan.shi@gatech.edu

We welcome your letters. Send them to intelligent@computer.org. Include your full name, title, affiliation, and email address. Letters are edited for clarity and space.