

Beyond SAWSDL—a game plan for broader adoption of Semantic Web Services

[Amit Sheth](#), [Kno.e.sis Center](#), Wright State University

After a flurry of research activities led by the OWL-S, WSMO, SWSF, and WSDL-S groups, we now have taken the first concrete steps toward building a Semantic Web Services (SWS) based solution, in the form of a W3C candidate recommendation SAWSDL [<http://www.w3.org/2002/ws/sawSDL/>], associated tools and use cases, and initial applications [1]. Where do we go from here? Researchers among us may be fully convinced of the importance and benefit of adding semantics to Web services and impatient to see their research translated into technologies and adapted for real use. However, I believe we may need to be patient and do a lot more before we see substantial adoption of SWS technologies. Here are some observations about what we need to do.

- (a) First, we must **be patient**. Both Web Services and Semantic Web are prerequisites for SWS, at least in the form in which SAWSDL or its extensions may evolve. Despite a clear uptick in adoption of Web Services by industry, initial success stories, and early adoption of Semantic Web technologies [2], broad acceptance and adoption may take another three years or so. For the Web Services part, the reason is the complexity and confusion resulting from WS-* specifications, some of which (such as security and policy/agreement) are critical and not yet mature. For Semantic Web, it is convincing practitioners that ontologies can be and have been built and be managed, that much can be done with limited semantics and without becoming an expert in description logic, that we do not need a single ontology for everything and we know how to work well in a multi-ontology environment, and that technology is ready for building robust applications. We might also have to wait out the current Web 2.0 euphoria, bordering on infatuation. Clearly Web 2.0 is seductive and easy to use and adopt, but it is not a solution to all problems, and technologies such as SWS will not get attention until there is a broader realization of what problems Web 2.0 cannot solve and why Web 3.0 (the new code for Semantic Web technologies enabled Web) is needed..
- (b) Second, we must do a better job understanding, **explaining, and managing the complexity of using SWS**. Take the example of SWS Challenge [<http://sws-challenge.org/>], in which six teams of researchers have implemented or tried to implement a solution to a realistic but relatively simple application using widely different approaches, formalisms, techniques and technologies. Only one team used SAWSDL (in its defense, we can note that this challenge started well before SAWSDL became a candidate recommendation, but WSDL-S submission and W3C's work toward SAWSDL has continued at the same time). More important, the exercise, which has already involved four meetings and spanned well over a year, shows how complex the challenges are, especially when the problem involves legacy systems or interfaces for preexisting services that must be used. Lessons from this exercise lead to the next two observations.

- (c) Before practitioners (developers and real-world users) will embrace SWS, we need to develop **robust tools as well as methodologies** to streamline all aspects of managing the SWS life cycle, including annotation, publication, discovery, data mediation, composition or configuration, orchestration, and execution. In the context of SAWSDL, an initial version of most of these have been developed (some by the LSDIS lab [<http://www.iswc2006.semanticweb.org/projects/meteor-s/SAWSDL/>] and Kno.e.sis Center [<http://knoesis.wright.edu/research/webservices/>]), but these tools must be robust and in a form that developers find attractive to learn and use. An end-to-end open source toolset developed by an international collaboration of researchers could facilitate this process.
- (d) Perhaps most important step is to answer clearly a question any manager or business decision maker will ask: **What is the ROI, what is the cost of adapting SWS technology, and what are the concrete benefits?** In my view, we need to focus on two benefits we have all talked about: service reuse and mediation/interoperability/integration. In late 1990s, I had a commercial product in Workflow Management (METEOR EAppS) that was used to develop some real-world applications. In the case of existing tasks or activities, I learned that 90 percent of the time is spent developing workflows related to data mediation issues (transforming the output of one task into the input of subsequent task or tasks). A clear demonstration of how SWS can ease this or equally vexing problems faced by process developers will be important. Today's ESB, application server, and service-based middleware vendors claim to solve the data mediation problem using XSLT-based transformation. Although this addresses a number of basic problems, as my student Karthik Gomadam put it, "the idea of mediating at the level of instance or between two fixed schemas is hackneyed." To accelerate adoption of this technology, we need to focus on issues such as data and process mediation where use of semantics will very likely provide solid cases of ROI. Efforts such as [3] will need to be put side by side with current non-semantic approaches to drive the point home.
- (e) Finally, while early examples of **real-world SAWSDL-based services** already exist (e.g., http://glycomics.ccruc.uga.edu/stargate/web_services.jsp), we will need many more before potential adopters of SWS technology feel that there is critical mass.

As a follow-on to SAWSDL and with encouragements from members of our community such as Dieter Fensel, Charles Petrie and I have started a SWS Testbed Incubator (<http://www.w3.org/2005/Incubator/swsc/>) that will likely continue building additional experience in developing SWS-based solutions with the help of SWS challenge. Kno.e.sis Center, in some cases in collaboration with other researchers, is looking at three issues discussed in SAWSDL calls that we hope will be components of future solutions:

- **development of SA-REST** for semantic annotation of REST and other lightweight services—such non-SOAP/WSDL services are widely used, and I see a clear value in providing a semantics-based solution in creating powerful **smashups** (for semantic mashups, as introduced in [4]) where it is easier to do data and service integration on the client side;

- **supporting precondition and effects** in SAWSDL—OWL-S and others have noted the need for some time, and more recently we have seen clear use cases emphasizing a need for such support; and
- **semantic annotation of policy descriptions**—from a business perspective, this will be very important, especially to support non-functional or QoS requirements.

In addition, we intend to collaborate with others who are developing an ontology for SWS, which I believe will build on work being done in the WSMO and OWL-S groups. On a more strategic and longer-term path, we hope to continue to push further toward understanding and demonstrating the value of semantics in making processes more agile, adaptive, and dynamic.

Rather than look for a clear winner among various SWS approaches, I believe that in the post-SAWSDL context significant contributions by each of the major approaches will likely influence how we incrementally enhance SAWSDL. Incrementally adding features (and hence complexity) when it makes sense, by borrowing from approaches offered by various researchers, will raise the chance that SAWSDL can present itself as the primary option for using semantics for real-world and industry-strength challenges involving use of Web services.

References

- [1] K. Verma and A. Sheth, “[Semantically Annotating a Web Service](#),” IEEE Internet Computing 11 (2), March–April 2007, pp. 83–85. Also, Kunal Verma and Amit Sheth: [Using SAWSDL for Semantic Service Interoperability](#), Tutorial at Semantic Technology Conference, San Jose, CA, May 21, 2007. <http://lists.w3.org/Archives/Public/www-archive/2007May/att-0081/SAWSDL-STC07-May21.ppt>
- [2] A. Sheth and S. Stephens, “[Semantic Web: Technologies and Applications for the Real-World](#)”, 16th World Wide Web Conference (WWW2007), Banff, Canada, May 8–12, 2007. <http://www2007.org/tutorial-T11.php>
- [3] M. Nagarajan, K. Verma, A. Sheth, J. Miller, and J. Lathem, “Semantic Interoperability of Web Services—Challenges and Experiences,” in Proceedings of the 4th International Conference on Web Services, Chicago, IL, September 2006, pp. 373–382.
- [4] 72. A. Sheth, K. Verma, K. Gomadam, “Semantics to energize the full Services Spectrum,” special issue on Services Science, Communications of the ACM (CACM) 49 (7), July 2006, pp. 55–61.



Amit Sheth is the LexisNexis Ohio Eminent Scholar and the director of Kno.e.sis Center (<http://knoesis.wright.edu>) at the Wright State University. He is a fellow of the IEEE. He is well published and cited in the areas of information integration, workflow management, Semantic Web and Semantic Web Services. From 1994 to 2006 he was a Professor at University of Georgia, where he founded and directed the LSDIS lab. Earlier he worked in R&D positions at Bellcore (Telcordia), Unisys, and Honeywell. His research has led to several commercial products and two companies—Infocsm in Workflow Management and Taalee/Semagix in Semantic Web. He is the EIC of Intl. Journal on Semantic Web and Information Systems and co-EIC of Distributed and Parallel Database Systems Journal. He received his graduate education at Ohio State University and undergraduate education at BITS, Pilani, India. More information at: <http://knoesis.wright.edu/amit>