

W3C Workshop on Constraints and Capabilities for Web Services

Bijan Parsia

MINDSWAP Group

University of Maryland at College Park

There are several purposes for which expressive representations of various aspects of Web Services including automating, to various degrees, service discovery, composition, and invocation. Degree of automation can range from some verification of properties of a completely human generated composition or assistance in selecting from known equivalent services, to the completely automated generation and execution of complex compositions. For any degree of automation there needs to be appropriately rich, “machine processible” descriptions. For example, most planning techniques require specification of the preconditions and effects of a service (conceived as a basic action) in terms of simple assertions of what must be true of the world before the action can be taken (precondition) and what is made true by performing that action (effect). By varying the language for expressing these assertions, we can make the description of the service easier for their creators and sometimes easier for the planner to work with (other time harder, if the expressivity make reasoning especially difficult).

The use case and example constraint and capability assertions in the call for papers for this workshop are nigh uniformly concerned with the relatively low level details of the invocation of Web Services, on a par with what is already available in WSDL and XML Schema. (The possible exception is the P3P requirement.) The requirements target conditions akin to the type of a function parameter in a programming language (e.g., the message must be delivered by a reliable protocol, it must have a certain structure (e.g., a certain encrypted header), etc.). While certainly very important, and fitting in appropriately with the existing W3C Web Service Description Language, I think that such a narrow focus — well below the application or business logic level — would be disappointing. OWL-S (as described, I’m sure, in another submitted paper) already manages to integrate descriptions from the low, invocation level (the “grounding”) to the high level, action oriented descriptions used by fully automated planning systems. Our lab, in conjunction with industrial partners, have built several end to end systems that can handle discovery, composition, and execution of Web Services with a mere subset of the expressivity afforded by OWL-S. We believe that a less general variant, built on the mapping of WSDL 2.0 to RDF and OWL would provide an adequate foundation for a practical, highly expressive description language.

To be effective, a policy language has to be:

- 1) psychologically adequate, that is, people must be able to generate and understand descriptions, as well as effectively debug them, predict their effect, etc.
- 2) analyzable, that is, the language must be well specified. At the W3C, the trend for analyzability has been toward formal methods: RDF, OWL, and XQuery have separate formal semantics documents; WS-Choreography is based on the pi-calculus and could be similarly formalized; WSDL will acquire a degree of formal specification from the RDF mapping. In contrast, specifications such

as WS-Policy claim to provide a “flexible and extensible *grammar*” for policy assertions. This is, from an interoperability perspective, the starting point, not the end game.

- 3) expressive, that is, it must allow us to say what we want to say in a manner consistent with the first two requirements.
- 4) composable, that is, it must be possible to combine policy assertions expressed in the language in natural but unanticipated ways; in line with 2, systems should be able to make appropriate decision on how to resolve or report issues with composed assertions (e.g., contradictory assertions, or combinations which impose unacceptable costs).
- 5) deployable, that is, there must be sufficient implementation and use experience of systems which generate and consume policy descriptions in the language.

MINDSWAP believes, of course, that the RDF and OWL languages clearly meet, at least in principle, 1,2,4, and 5. We have found the expressivity of OWL (thus, its ability to meet 3) to be adequate for quite a lot. However, for some classes of problem, rule oriented languages often have desirable expressive capabilities, and may be more psychologically adequate.

MINDSWAP also believes that *ad hoc* reinvention of assertion languages, and, inevitable, more expressive extensions to be a very bad idea. We recognize some practical difficulties with reusing current W3C Semantic Web technology (in particular, we all know that the RDF/XML syntax is both painful for humans, and fails to interact well with DTD or Schema validation). We believe that these problems are relatively easy to overcome (e.g., there is already a Schema friendly OWL presentation syntax) and would repay both by providing sound formal properties and reuse the growing deployment of RDF and OWL databases, reasoners, vocabularies, and user communities.