

W3C Spanish Office Prize for Web Standardization

Project Documentation

Project

As far as I am concerned, there are two important points in launching and improving the Semantic Web: **tool support** and **ontology reuse**. The aim of this project goes in these two ways, as well as in using together Web Services and Semantic Web technologies, specifically in **using Web Services to retrieve ontologies**.

The project could be briefly described as **an application that puts together a Semantic Web application development framework and a Web Service connection module for Ontology searching and retrieving**. This application can be called **SWOWS**.

If starting from scratch, the framework would be a simple text editor with syntax highlighting and support for RDF, RDFS and OWL. But programmatic and graphical visualization features could be added, converting it into a Semantic Web IDE.

The Web Service connection module for ontology searching should be there from the first version, what should suppose no major problem, as all the main programming languages offer support to deal with Web Services.

Of course, for SWOWS to be useful, it is necessary that ontologies and web services that serve them as a response to a keyword or whatever, exist. But the creation of such a tool in a way which was useful for Semantic Web developers could also encourage the development of more ontologies and such web services, as well as further integration and applications between Semantic Web and Web Services technologies, which is also an important point.

Technological description

A detailed technological description is not in the scope of this documentation, as this is only an idea, but Semantic Web language parsers and inference engines are examples of what could be needed, although some of the technological needs would depend on the features supported by the framework on each version. Of course, as mentioned earlier, support to deal with Web Services and their technologies (SOAP, WSDL...) is also required.

Due to the above, the need of a graphical user interface and the nature of the application itself, it can be said that languages like C++, Java or C# are good candidates for the implementation of SWOWS.

Functional description

The main functional features of SWOWS are described below:

- Semantic Web languages syntax highlighting and edition support.
- Verification of which OWL sublanguage (Lite, DL, Full) corresponds to an ontology.
- Graphical visualization of the relation between the elements (classes, properties, individuals...) described by the Semantic Web languages.
- Ontology search and retrieve on Web Services. This should be as easy as to give an URL. From it the tool should show the user the functions that could use and the parameters to set (words for matching in the ontology title or description, kind of OWL the ontology should belong to, version or compatibility requirements, etc.). Once the parameters are set appropriately, the query is launched and the corresponding ontologies are retrieved and shown.

As a practical case to show the functionality of the project, imagine that a Semantic Web developer is implementing a Semantic Web application for a wine seller. He has been creating a really wonderful application thanks to the support of SWOWS, when he realises that the wine ontology, surely an important part of the application, isn't done yet. He remembers that in the W3C OWL specification documents, there's a wine ontology he could reuse or at least take as a base or example for his. But for some reason (I can't imagine one) he isn't able to get it. As he falls into desperation, he starts playing with the buttons and menus of the SWOWS Web Service connection module interface and ... suddenly he finds the solution: he could use the module to access the new W3C Ontology Repository. Five minutes later he has revised the ontologies retrieved from the repository with the SWOWS graphical visualization environment and he has found one that, with some additions, he can reuse.

All the description above covers the basic idea of the project and what I think is more feasible as starting point with the present state of the technologies. But not stopping here and taking a look beyond opens many more possibilities.

Regarding extensibility and interoperability, the following possibilities can be taken into consideration:

- SWOWS could be extended with Web Service Discovery and Semantic Web Services.
- The whole SWOWS could be integrated or implemented in Amaya, the W3C web browsing and authoring environment.
- The new Semantic Web languages and standards of the upper layers (rules, logic proof and trust) could also be added in SWOWS as they are released.
- The framework could work with other technologies and standards apart from the ones of the Semantic Web, becoming a powerful integration environment.
- More features for ontology retrieving apart from the ones mentioned before could be thought and added.

The portability of all the application would depend on the implementation language, although the Web Services access should not be a problem. However, different distributions could be done if necessary, as is done in the case of Amaya.