

# Accessing and Manipulating Life-Sciences Ontologies Using Web Services

Olivier Dameron, Mark A. Musen

Stanford Medical Informatics, Stanford University,  
251 Campus Drive, X-215, Stanford, CA 94305, USA  
{dameron, musen}@smi.stanford.edu

## Abstract

In this technology position paper, we propose to provide ontology access and manipulation functions as Web Services called Ontology Web Services (OWSes), and to describe their semantics, using OWL-S as an example ontology.

We present an application context that already uses Semantic Web techniques. We show that Ontology Web Services are a good solution for bringing together resources from several medical domains such as anatomy and pathology. This example illustrates that the Semantic Web provides a relevant framework for Life Sciences.

We advocate that Ontology Web Services are useful in domains other than Life Sciences, and are even necessary for Semantic Web Services clients to take full advantage of the semantic descriptions of Web Services.

## 1 Context

**Semantic needs for Life Sciences** Life Sciences need automatic tools for accessing, retrieving or processing the existing huge corpus of data and knowledge. These tools have to handle both the syntactic and semantic heterogeneity of this corpus. Therefore, they require an explicit and formalized representation of the domain's knowledge. Eventually, Life Sciences also need to have some of their applications interact. Here again, it requires to address the semantic interoperability. Moreover, having this interoperability automated as much as possible for discovery, invocation or composition requires to process some semantic information.

**Overlap with the Semantic Web approach** Some of the limitations encountered today by Life Sciences for sharing data, sharing knowledge or enhancing the interoperability of applications are common with other domains. Web technologies seem to be the the most promising solution. XML and Web Services are already mainstream techniques. A lot of efforts have been invested in

the Life Sciences community for providing ontologies such as GO<sup>1</sup>, OMIM<sup>2</sup> and MGED<sup>3</sup> for genetic knowledge, Galen<sup>4</sup> for pathologies or the FMA<sup>5</sup> for human anatomy. However, the current status of the Semantic Web doesn't allow yet to fully take advantage of the existing resources. Particularly, the previous ontologies are under-exploited and still in search for applications that would reveal their true potential.

We believe that (1) Life Sciences have the material to provide an interesting and demonstrative test case for a Semantic Web killer application, and (2) some of the outcome developed in this context would be generalizable to other domains.

## 2 Objectives

We assume that semantically enabled applications will rely on a set of generic ontology-manipulation functionalities. Therefore, we propose to implement domain-independent ontology manipulation functions as Web Services. We call such services Ontology Web Services (OWS) and propose a definition in section 3.1. In section 3.2, we present a scenario highlighting the need for Ontology Web Services in a life science context. We believe that this scenario can be implemented with the current Web technologies.

Moreover, we advocate that these ontology-manipulation functionalities are also necessary to the framework of the Semantic Web for processing semantic descriptions of Web Services so that they can be automatically retrieved and combined. In section 3.3, we explain how semantic descriptions of Ontology Web Services are necessary both for their use in a Semantic Web context and for integration of Semantic Web Services themselves.

## 3 Ontology Web Services

### 3.1 Definition

*Ontology Web Services (OWSes)* are Web Services providing generic ontology-access and ontology-manipulation functionalities.

These functionalities can be gathered in several broad categories:

- *Query access to ontologies* returns a set of variable bindings associating the variables of the query with the possible classes or relationships;
- *Generation of ontology views* returns the self-contained subset of an ontology that match the view specification;

---

<sup>1</sup><http://www.geneontology.org/>

<sup>2</sup><http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM>

<sup>3</sup><http://www.mged.org/>

<sup>4</sup><http://www.opengalen.org/>

<sup>5</sup><http://sig.biostr.washington.edu/projects/fm/>

- *Translation of ontologies* from one language to another;
- *Management of multiple ontologies* allow to establish correspondances between multiple ontologies, recognize where they are different, compose them or handle versioning issues;
- *Reasoning on ontologies* encompasses Description Logics classification or rule processing.

The list is not exhaustive and it will grow larger as use of SWS with ontologies becomes more popular. Moreover, realistic usage of Ontology Web Services will involve a combination of the categories above.

### 3.2 Ontology Web Services implementation scenario

For an example scenario illustrating OWSeS consider retrieval of the clinical trials relevant to a patient with lung tumor.

The task consists in assessing the staging of the patient’s tumor according to the TNM classification for lung tumors<sup>6</sup> and in combining the result with other information such as geographic location, age or previously prescribed drugs to query the NCI Clinical Trials<sup>7</sup> on-line database.

We will focus on the staging of the tumor according to the TNM criteria. The TNM classification consists in assessing the patient’s condition according to three axis: the primary tumor ( $T_0$  to  $T_4$ ), metastasis in the lymph nodes ( $N_0$  to  $N_3$ ) and the presence of distant metastasis ( $M_0 - M_1$ ). The tumor stage (stage 0 to stage IV) is then derived from the score along each of these axis: e.g., a  $T_3N_2M_0$  tumor is a stage IIIa tumor. Performing TNM classification is rather complex, because it relies extensively on domain-specific knowledge about the tumors and their metastasis, as well as on their anatomical location. The last point is of particular importance, because it is possible to recognize that a metastasis located in the epicardium of the left ventricle meets the criterion “metastasis located in the heart” only if one knows that (1) the epicardium of the left ventricle is a part of the heart, and (2) that a tumor located in a part of an organ is also located in this organ. The necessary knowledge for performing this classification encompasses:

- a representation of the various  $T_x$ ,  $N_y$  and  $M_z$  criteria and of the various tumor stages;
- a taxonomy of tumors;
- a taxonomy of pathologies so that an “acute obtrusive pneumonitis” will potentially match the  $T_2$  criterion, which refers to the more general concept of “obstructive pneumonitis”.

---

<sup>6</sup><http://www.bccancer.bc.ca/HPI/CancerManagementGuidelines/Lung/Staging/default.htm>

<sup>7</sup><http://www.cancer.gov/clinicaltrials>

- an ontology of anatomy representing taxonomies and partonomies, so that a tumor located in the left ventricle of the heart could indeed be considered matching the  $T_4$  criterion, or so that we know whether the pneumonitis involves the entire lung (here left lung and right lung have to be identified as kinds of lung) or only a part of it.

Most of the necessary knowledge is already formalized in existing ontologies. The Foundational Model of Anatomy<sup>8</sup> (FMA) provides the anatomical knowledge. The NCI oncology ontology<sup>9</sup> provides the knowledge about pathologies. To our knowledge, no formalized ontology of the TNM criteria exists yet, but creating one is straightforward.

The staging of tumors can be modeled as a classification problem. Therefore, OWL appears as a natural representation formalism. Moreover, generic classifiers such as Racer<sup>10</sup> can be used.

Bringing these ontologies together requires several ontology manipulations that could be accessed as remote Ontology Web Services (see Figure 1).

First, for computational efficiency considerations, we only want to merge the relevant portions of the FMA (which contains over 70,000 concepts and 1.5 million relationship instances from 168 relationship types) and of the NCI oncology ontology (which contains over 500,000 triples). Therefore, we need to extract *views* of these ontologies.

The FMA (or a view on the FMA) is not directly available in the OWL language. Therefore, we need a *translation* function.

The resulting OWL representation of the view on the FMA, we need to merge the view on the NCI oncology ontology and the TNM ontology.

We then need to map the patient's data have into the ontology before classification.

Finally, we need to query the result of classification.

### 3.3 Ontology Web Services semantic description

For the Semantic Web to take off, software programs must be able not only to manipulate data or knowledge, but also to automate the use of functions. Just as the automatic processing of data requires semantic markup, the automatic discovery, execution and composition of services relies on a formalized description of what they do and how to communicate with them.

An extensive amount of work has already been done on the description of a syntactically valid communication with a Web Service. Languages such as SOAP<sup>11</sup> and WSDL<sup>12</sup> have gained a widespread acceptance.

<sup>8</sup><http://sig.biostr.washington.edu/projects/fm/>

<sup>9</sup><http://www.mindswap.org/2003/CancerOntology/>

<sup>10</sup><http://www.cs.concordia.ca/~haarslev/racer/>

<sup>11</sup><http://www.w3.org/TR/soap12-part0/>

<sup>12</sup><http://www.w3.org/TR/wsd1>

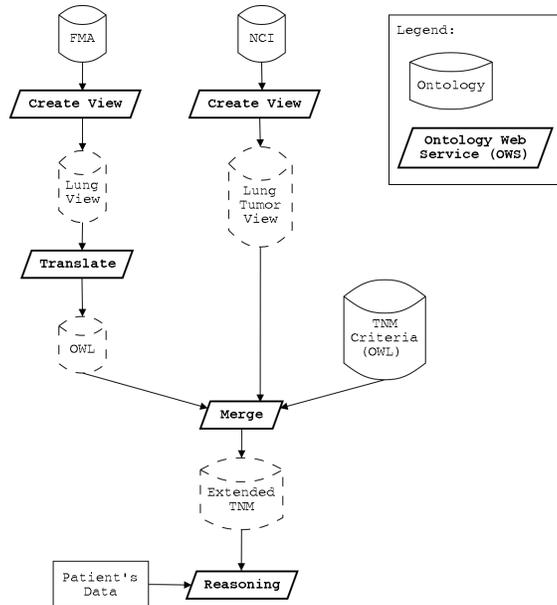


Figure 1: Ontology Web Services used for assessing the TNM classification of some lung tumors

The description of the semantic aspect of a communication with a Web Services encompasses what the service is doing, what kind of arguments it requires, which conditions have to be true. Such a description is necessary for the potential client to automatically decide if the service matches its expectations, and if so, for the client to perform the necessary steps for providing the server with the right arguments in the right form. Today, the more promising proposal is OWL-S<sup>13</sup>. As stated on the OWL-S website, it *supplies Web service providers with a core set of markup language constructs for describing the properties and capabilities of their Web services in unambiguous, computer-interpretable form. OWL-S markup of Web services will facilitate the automation of Web service tasks including automated Web service discovery, execution, inter-operation, composition and execution monitoring.*

Processing the OWL-S description of a Web Service involves some semantic access and processing operations such as mapping the client's domain ontology (or at least a view of it) to the server one (or to a view of the server's one). Therefore, OWSEs are a necessary part of the solution for applications to be able to select autonomously the Web Services they need and to interact with them. Requiring every client to implement its own version of such functions is

<sup>13</sup><http://www.daml.org/services/owl-s/>

a waste of resources, and will probably make the widespread development of Semantic Web applications too difficult.

Because OWSEs are a kind of Web Services, it is natural to provide a semantic description (e.g. in OWL-S) for them. Then we find ourselves in the situation where applications may need OWSEs to process the OWL-S description of some Web Service they are interested in, but finding the OWSEs require processing (through other OWSEs) the OWL-S description of the OWS.

We need to turn the chicken and egg problem of processing the OWL-S description of Ontology Web Services into bootstrapping. A solution could be to implement OWSEs brokers that would indicate to any applications the OWSEs it needs to start processing OWL-S.

### 3.4 Future work

We already have a simple Web Service that performs staging of lung tumors. However, we generated the “Extended TNM” ontology presented in Figure 1 manually.

First, we plan to develop the various OWSEs presented in Section 3.2 and their WSDL descriptions. As a proof of concept, we will have the tumor staging Web Service automatically generate the “Extended TNM” ontology. This automatic generation will follow a predefined *ad hoc* script, hard-coded into the tumor staging Web Service. It will require to implement the Web Services for computing the views on the FMA and on the NCI oncology ontology, the one for translating the FMA view into OWL, and the one for merging these OWL views with the TNM criteria ontology. The tumor staging Web Service will allow clients to create instances representing actual patient’s tumors and metastasis, and will classify them according to the “Extended TNM” ontology in order to infer their stage. Such a Web Service can be used as a standalone decision support application, or in combination with other services for proposing the relevant medical trials to a patient. The goal at this point is to demonstrate the use of OWSEs by Semantic Web-enabled applications in a Life Science context.

Second, we will generate the OWL-S descriptions of the OWSEs. We will then have the tumor staging Web Service rely on these descriptions in order to automatically (as much as possible) select the Web Services necessary to the completion of its goal and interact with them. The goal at this point is to demonstrate the use of OWSEs for processing the semantic descriptions of Web Services.

## 4 Expected benefits

The immediate benefits of our approach is to provide a consistent and reusable set of generic and modular semantic services for applications. In addition, it offers a complement to OWL-S by providing some of the functionalities required for processing OWL-S descriptions.

We believe that the Life Sciences community and the Semantic Web community can mutually benefit from their needs and proposals.

**Benefits for Life Sciences** We have exposed in section 1 that the Life Sciences tools need to exploit the distributed and semantically heterogeneous data currently available. These tools should rely on an explicit and formalized representation of knowledge. Moreover, a goal of the Life Sciences researchers is to make the knowledge they rely on evolve. This process is based on an interdependency between the data and the knowledge used to interpret them. So far, the applications we have presented focus on assisting the interpretation of data, but it is also an important stake to provide tools for enhancing the knowledge by a systematic exploitation of the data.

Such issues are receiving a lot of interest from the bioinformatics community, but they are not restricted to the Life Sciences domain. Therefore, reusing generic techniques that have been developed and thoroughly studied by specialists, instead of relying on *ad hoc* kludges is appealing. It is certainly helpful to “delegate” the design and implementation of those generic features to the whole Web community, which may know better than we do.

The perspectives offered by the Semantic Web are also a good opportunity for the Life Sciences community to show that the efforts and investments that were made for building shared databases and knowledge bases are beginning to be profitable. First, Semantic Web tools can potentially allow us (or at least help us) to process the huge amount of existing data that are currently underexploited because we haven’t been able to overcome their semantic heterogeneity automatically. This is not only true for specific fields such as epidemiology or neuroscience, but also for establishing connections between domains such as anatomy, physiology or psychology.

Second, by providing tools for knowledge aggregation and management, the Semantic Web approach can help to avoid the duplication of modeling efforts and thus to limit inconsistencies and to enhance interoperability. Existing ontologies such as Galen or the Digital Anatomist clearly aim at achieving such a cumulative effect, as opposed to the multiplication of efforts that resulted from the lack of general vision.

Third, the Semantic Web approach can prove helpful for promoting the (re)use of knowledge in applications, such as decision support.

Moreover, in the long term, this effort may lead to cross fertilization between the Life Science community and other communities, such as physics, chemistry, or maybe psychology. In this perspective, using the Web as a common ground is not only a convenient solution, but also a requirement.

**Benefits for the Semantic Web** Conversely, several points make Life Sciences a good test case for the Semantic Web.

First, Semantic Web provides solutions to some actual needs in the the Life Science domain; it doesn’t have to create the needs. Moreover, the problems are serious ones (i.e. not the usual travel agency example), which adds up to

the relevance of the demonstration.

Second, the Semantic Web effort benefits from a general trend to share data and knowledge.

Third, important efforts have previously been dedicated to building databases and formalized knowledge bases. Therefore, there are already some available resources, although some may need to be “upgraded” to the Semantic Web level.

Fourth, there may be opportunities for funding both research and development of real applications.

Eventually, the Semantic Web is looking for a killer application to back up its position, and the Life Sciences are eager to be part of this scheme.

One potential issue is that Life Sciences go for heavy weight ontologies and semantic services, whereas the Semantic Web in general is more directed toward “semantically lightweight”.

## **Acknowledgments**

This work has been partially supported by the French research institute INRIA. Natalya Noy provided valuable advice.