The AIDA Toolkit - a Tool for Users and Developers

Adianto Wibisono[1], Marco Roos[1,2], M. Scott Marshall[1,3,*]

[1] Informatics Institute, Faculty of Science, University of Amsterdam, P.O. Box 94323, 1090 GH Amsterdam, The Netherlands
a.wibisono@uva.nl
[2] BioSemantics Group, Department of Human and Clinical Genetics, Leiden University Medical Centre, P.O. Box 9600, 2300 RC Leiden, The Netherlands
m.roos@lumc.nl
[3] Department of Medical Statistics and Bioinformatics, Leiden University Medical Centre, P.O. Box 9600, 2300 RC Leiden, The Netherlands
* Corresponding author: mscottmarshall@gmail.com

Abstract. Vocabularies in the form of ontologies and terminologies are becoming an accepted way of incorporating both logic and linguistic resources into biomedical applications. In order to perform search and annotation with biomedical ontologies and terminologies, users and developers must be able to access and browse vocabularies. However, accessing the contents of a repository requires not only a working knowledge of the SPARQL query language, but knowledge of the repository structure and contents in order to formulate a query. Using the web services in the AIDA Toolkit, we have developed a web-based repository browser that can quickly detect the type of RDF dialect used and extract hierarchies of interest based on common patterns, such as the subclass or subsumption hierarchy, displaying the hierarchy as an interactive outline view of the labels. This functionality makes it possible to explore the basic structure of a vocabulary served by most SPARQL endpoints and find specific terms via auto-completion without any prior knowledge of either the specific vocabulary, RDF, or the SPARQL query language. The combination of browsing knowledge resources available from triplestores with the ability to create more complex queries that build upon the elements of a vocabulary makes it easier to incorporate knowledge resources into applications that are customized and scoped by well-defined vocabularies.

Keywords: SPARQL, RDF, OWL, SKOS, Vocabulary Services, Semantic Web

1 Introduction

Vocabularies in the form of ontologies and terminologies are becoming an accepted way of incorporating both logic and linguistic resources into biomedical applications. For example, the approximately 200 ontologies being offered from the

National Center for Biomedical Ontology’s Bioportal[1] are being integrated into applications via Web API's and a SPARQL endpoint. The practice of incorporating potentially distributed knowledge resources into applications via services is in stark contrast to previous practices of embedding such resources directly into the application and demonstrates an emerging trend to dynamically incorporate the most appropriate and latest vocabularies into applications. However, developers wishing to gain understanding of the vocabularies that they might like to use in their applications are faced with a bootstrapping dilemma: In order to formulate a query in SPARQL that will enable them to browse the vocabulary, they must first know something about the structure of the data.

2 The AIDA Toolkit

Using the web services in the AIDA Toolkit, we have developed a web-based repository browser that can quickly detect the type of RDF dialect used (e.g. OWL, SKOS, etc.) and extract hierarchies of interest based on common patterns, such as the subclass or subsumption hierarchy, displaying the hierarchy as an interactive outline view of the labels. The AIDA repository browser[1] is light-weight and flexible, enabling a user to explore vocabularies stored in most triplestores that support the SPARQL API, including Sesame, Virtuoso, Allegrograph, and Mulgara.

The AIDA Toolkit resulted from research done in the context of the Virtual Laboratory for e-Science project in the Netherlands[2]. The first version of the repository browser was built specifically to browse vocabularies in SKOS – the Simple Knowledge Organization System language for thesauri in RDF. The original application was based on web services that used the Sesame query language (SeRQL was used in pre-SPARQL implementations) and required conversion into SKOS in order to browse OWL. Many improvements and extensions have been added, including support for SPARQL, multiple repository types, REST, and OWL browsing, as well as auto-completion of labels, a repository configuration panel, and the ability to search the labels of several repositories at different SPARQL endpoints simultaneously. With the Direct Link functionality of the client, users can exchange URLs that enable the browsing of a vocabulary directly from the URL. Threaded updates of the plus symbol indicators (of subtree contents) in the outline view and caching have improved the user experience.

The AIDA browsing functionality has been used to browse SNOMED-CT and MeSH in SKOS. It has also been used to create a Taverna plugin, for the semantic annotation of bioinformatics workflows and workflow data using, for example, the myGrid ontology[3,4]. As well as shielding non-technical users from technical details

[1] http://ws.adaptivedisclosure.org/search/ (note: substitute “dev” for “ws” for the latest release)

while providing them access to data 'where it lives', AIDA makes it possible for developers to inspect the working SPARQL queries that have been used 'under the hood' with a ‘View Source’ feature and modify them for use on the data in the same repository. The extraction patterns used can be modified directly by the user without requiring recompilation, achieving agile interface development when unexpected patterns in the data are encountered. Another feature of the web-based AIDA application is direct access to Lucene indexes, such as a PubMed index that is refreshed nightly and accessible to our own services at http://ws.adaptivedisclosure.org. The latest build of the AIDA Toolkit can be downloaded[5] and run in one’s own instance of tomcat, for example, giving developers the ability to create their own customized Lucene indexes from their own literature corpus.

Our intention with the AIDA Repository Browser web client is to give developers some basic functionality as a starting point for exploring vocabularies in a triplestore, such as the ability to search for labels via auto-completion or the search tab in the repository browser. We also wanted to provide an example web application that makes use of the AIDA Web Services to show how vocabulary services can be embedded and applied in a web setting to personalize and customize interfaces with labels that come directly from those vocabulary services. This approach is more flexible than importing an entire vocabulary into application memory.

3 Future Work

The query editor would be more useful if users could name and save edited queries for future use. Such named queries could be programmed to automatically run and compare to previous results, for example, as unit tests[6] that will alert developers to changed results when ontologies or mappings are altered. Another application of the result comparison functionality could be for alerts to new results in literature search. For this reason and for the purposes of annotation functionality, we have begun prototyping user authentication with OpenID.

With SWObjects[7], it is possible to create *semantic views* of both triple stores and relational databases, through the use of mapping rules implemented as SPARQL Constructs. In principle, the hierarchical views that we create in the AIDA repository browser with pre-coded query patterns could be more flexibly managed with dynamic SWObjects mappings, specially generated for hierarchical extraction.

As query federation is further developed, the provision of provenance about a given repository and its contents will make it possible to automatically locate data sources that meet specific criteria. AIDA could eventually make use of repository provenance to guide the choice of both repositories and the named graphs within them.

Finally, lexical and semantic synonym expansion during auto-completion of query patterns in the query editor could be accomplished with access to a list of related knowledge resources. Semantic guidance that makes use of principles demonstrated in this workshop's "SPARQL Assist Language-Neutral Query Composer" could simplify SPARQL query composition.

4 Acknowledgements

We thank Cees de Laat, Pieter Adriaans, Bob Hertzberger, Jelle Goeman, and Theo Stijnen for their support.

5 References

1. Noy, N.F., Shah, N.H., Whetzel, P.L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D.L., Storey, M., Chute, C.G., Musen, M.A.: BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research. 37, W170-173 (2009).
2. M. Scott Marshall, Marco Roos, Edgar Meij, Sophia Katrenko, Willem Robert van Hage, Pieter W. Adriaans.: Semantic disclosure in an e-Science environment. Book Chapter in Huajun Chen, Yimin Wang and Kei Cheung (eds.) - Semantic e-Science, Springer Annals of Information Systems AoIS, Springer, April 2009.
3. Roos, M., Marshall, M.S., Gibson, A.P., Schuemie, M., Meij, E., Katrenko, S., van Hage, W.R., Krommydas, K., Adriaans, P.W.: Structuring and extracting knowledge for the support of hypothesis generation in molecular biology. BMC bioinformatics. 10 Suppl 10, S9 (2009).
4. Marco Roos, Sean Bechhofer, Jun Zhao, Paolo Missier, David Newman, Dave de Roure, M. Scott Marshall.: A Linked Data Approach to Sharing Workflows and Workflow Results, Proceedings of Tools in Scientific Workflow Composition Track of ISoLA 2010, Crete, Greece (2010).
5. http://adaptivedisclosure.org/aida/download/
6. Joanne S. Luciano, Bosse Andersson, Colin Batchelor, Olivier Bodenreider, Tim Clark, Christine Denney, Christopher Domarew, Thomas Gambet, Anja Jentzsch, Vipul Kashyap, Peter Kos, Julia Kozlovsky, M. Scott Marshall, James P. McCusker, Deborah L. McGuinness, Timothy Lebo, Chimezie Ogbuji, Elgar Pichler, Robert L.Powers, Eric Prud'hommeaux, Matthias Samwald, Lynn Schriml, Peter J. Tonellato, Patricia L. Whetzel, Jun Zhao, Susie Stephens, Michel Dumontier, The Translational Medicine Ontology and Knowledge Base: Using Semantic Web Technology in Personalized Medicine for Data Integration. 2011 AMIA Summit on Translational Bioinformatics (in press)
7. Prud'hommeaux, Eric, Deus, Helena, and Marshall, M. Scott. Tutorial: Query Federation with SWObjects. Available from Nature Precedings <http://dx.doi.org/10.1038/npre.2011.5538.1> (2011)