A user-friendly interface to browse and find DOAP projects with doap:store

Alexandre Passant

Université Paris IV Sorbonne, Laboratoire LaLICC, Paris, France
alexandre.passant@paris4.sorbonne.fr

1 Motivation

The DOAP[2] vocabulary is now widely used by people - and organizations - to describe their projects using Semantic Web standards. Yet, since files are spread around the Web, there is no easy way to find a project regarding its metadata.

Recently, Ping The Semantic Web[1] (PTSW) and Semantic Radar[2] plugin for Firefox introduced a new way to discover Semantic Web documents[1]: by browsing the Web, users ping the PTSW service so that it can maintain a contineously updated list of Semantic Web document URIs.

Thus, the idea of doap:store - http://doapstore.org is to provide a user-friendly interface, easily accessible for not RDF-aware users, to find and browse DOAP projects, using PTSW as a provider of data sources. This way, users do not have to register to promote a project as in freshmeat[3] or related services, but just need to publish some DOAP files on their websites to benefit of this distributed architecture. doap:store is the first implemented service using PTSW data sources to provide such browsing and querying features.

2 Architecture

doap:store involves 3 main components:

- A crawler: Running hourly, a tiny script parses the list of latest DOAP pings received by PTSW and then put each related RDF files into a triple-store;
- A triple-store: The core of the system, storing RDF files retrieved thanks to the crawler, and providing SPARQL capabilities to be used by the user-interface that is plugged on the endpoint;
- A user-interface: A simple interface, offering a list of latest retrieved projects, a case-insensitive tagcloud of programming languages, and a search engine to find DOAP projects regarding various criteria in a easy way.

While the crawler is written in Python, the interface is PHP5-based and the triple-store used is 3store[3], so both the crawler and the interface use its API to fetch and retrieve data. The whole application - without the API - is about 600 lines of code.

[1] http://pingthesemanticweb.com

[2] http://sioc-project.org/firefox

[3] http://freshmeat.net

3 Finding and browsing DOAP files

Apart the tagcloud used to find projects by programming language, a simple search-engine can be used to retrieve projects by (1) name (`doap:name`), (2) description (both `doap:desc` and `doap:shortdesc`), (3) name or description and (4) hostname (using the URI of the graph containing a project, since 3store is context-aware). A single SPARQL[4] query is used to find related projects, with an `FILTER REGEXP` expression added to the query depending on the search criteria.

Users can simply browse retrieved projects, ordered by name. Each project page provides a view of its available metadata, with links to the original RDF file and to a page displaying other projects from the same hostname. Since the DOAP ontology provides `rdf:label` for all its properties, not only project metadata but also property names are retrieved from the triple-store.

Another friendly way to query doap:store is to use YubNub[4], a command line service for the Web, since a `doap` command have been created for it. So, from their browser search engine or any YubNub client, users can type `doap desc=RDF` to be redirected to the doap:store results page listing projects with a description containing the `RDF` string.

Finally, for most advanced users, doap:store offers a SPARQL endpoint[5] - using a Javascript editor[6] - that can be used to query data or construct new RDF documents based on the actual content of the triple-store.

Thus, all these features provide various ways to retrieve informations about DOAP projects, from the easiest interface to the most advances SPARQL queries, in a single interface.

4 Acknowledgements

I would like to thank Uldis Bojārs and Frederick Giasson for their respective work on Semantic Radar and PTSW, that help to provide data sources for this service.

References

1. U. Bojārs, A. Passant, F. Giasson, and J. G. Breslin. An Architecture to Discover and Query Decentralized RDF Data. In *3rd Workshop on Scripting For The Semantic Web (SFSW07)*, June 2007.
2. E. Dumbill. DOAP: Description of a Project. http://usefulinc.com/doap/.
3. S. Harris. SPARQL query processing with conventional relational database systems. In *International Workshop on Scalable Semantic Web Knowledge Base System (SSWS 2005).*, 2005.
4. E. Prud'hommeaux and A. Seaborne. SPARQL Query Language for RDF. W3C Working Draft, W3C, 2006. http://www.w3.org/TR/rdf-sparql-query/.

[4] `http://yubnub.org`

[5] `http://doapstore.org/sparql.php`

[6] `http://dannyayers.com/2006/09/27/javascript-sparql-editor`