WaSABi 2014: Breakout Brainstorming Session Summary

Sam Coppens[1], Karl Hammar[2], Magnus Knuth[3], Marco Neumann[4], Dominique Ritze[5], Miel Vander Sande[6]

[1] IBM Research - Smarter Cities Technology Center (SCTC), Ireland
[2] Information Engineering Group, Jönköping University, Sweden
[3] Hasso Plattner Institute, University of Potsdam, Germany
[4] KONA LLC, New York, USA
[5] Research Group Data and Web Science, University of Mannheim, Germany
[6] iMinds - Multimedia Lab, Ghent University, Belgium

1 Introduction

A key program point of the 2nd International Workshop on Semantic Web Enterprise Adoption and Best Practice (WaSABi 2014) was the breakout brainstorming session, at which challenges regarding enterprise adoption of Semantic Web technologies were discussed, and potential solutions to some of those challenges described.

The requested output of this session was for each group of 5 people to produce a few slides detailing such challenges and solutions, with an eye towards possible future collaboration areas between the academic and industry representatives.

Each group was provided with conversation starter sheets on enterprise adoption topics identified by the workshop chairs, and given an hour of time to discuss and summarise issues related to these topics. The resulting slides are attached to the WaSABi 2014 proceedings. Additionally, the contents of the presentations accompanying the slides are summarised in the following sections.

2 Challenges

One group discussed topics related to maintainability of systems based on Semantic Web technology. This is obviously a fundamental software quality issue that needs to be handled if enterprise is to adapt such technologies; systems must be maintainable for the duration of their lifetime, which may be significantly longer than the lifetimes that academics developing prototypes for research papers are accustomed to. Three challenges in particular were considered, though this list is far from exhaustive:

- *The longevity of URIs in a changing environment* - the distributed nature of the Semantic Web and the Linked Open Data cloud means that companies may make themselves dependant upon resources that are outside of their sphere of influence. What might happen if an entire critical namespace were to go offline due to lack of funding or interest? For instance, if xmlns.com (which hosts the FOAF namespac) were to shut down, or purl.org, or schema.org, etc?

- *Persistence of vocabularies/datasets* - Related to the above, what might happen if a resource published on a particular namespace were to be modified in a way that is incompatible with existing systems that makes use of said resource?
- *Communication of popular technologies and vocabularies* - A lot of ontologies and technology are released by academia, but do not reach sufficient degrees of adoption to warrant maintaining in the long term. How can enterprises (and academics!) measure the adoption rate and subsequent chance of long-term survivability of such technologies and ontologies?

Another group discussed the issue of hiding complexity from software developers and simply making the tech work. For Semantic Web technology to be adopted, it is crucial that the learning curve be as flat as possible. Otherwise established methods and technologies, that may be poorer in a number of perspectives than their Semantic Web counterparts but are still sufficient to solve the task satisfactorily, will win out. Consequently, the Semantic Web community needs to consider how the complexity of the Semantic Web stack can be abstracted away and hidden from software developers. Here some issues were particularly singled out:

- *What are the reference solutions / best practices concerning some common and basic developer problems?* - Some examples where there are presently no single leading Semantic Web technology include: ORM/DAO transformation, ETL, and LDP/Middleware.
- *Where to make the cut skipping Semantic Web internals?* - That is, how much Semantic Web technology do we expose to the developer users, and how much do we abstract away? If we hide too much of the complex stuff, there is a risk that developers to not make full use of the power of Semantic Web technologies, but rather consider it "yet another data storage mechanism" or "just another API".
- *Scalability: Volume, Velocity, Variety* - There is in the research community still significant discussion on how to reach scalability using Semantic Web technologies, whether general purpose SPARQL endpoints are usable at all, whether REST-ful APIs are the way to go, whether Linked Data Fragments may be a way forward, etc. Developers do not want to deal with this uncertainty, nor having to make hard judgements about tradeoffs and technology ahead of getting started with a project; they just want tech that works sufficiently well and with the promise of future scalability.

3 Maintainability Challenges Solution Suggestions

Regarding the first set of problems, the maintainability of systems based on Semantic Web technology, a number of potential solutions were suggested. Again, this list of course not exhaustive, but may be a useful starting point.

3.1 The longevity of URIs in a changing environment

Developers building vocabularies that they expect to be around for some time, or to be used in a non-trivial context, should seriously consider under which namespace they publish that vocabulary. It is likely that using the web hosting provided by their current employer is a bad choice, as a simple change of jobs might break the namespace. The suggestion of the breakout group is to make use of the W3C Vocabulary Services[7], available to any W3C Community Group (i.e. you'll need at least five people accepting to be part of the community maintaining this namespace). The breakout group has a hard time seeing any organisation more likely to maintain a consistent and stable long-term namespace than the W3C.

When it comes to existing vocabularies that are hosted on namespaces that are not guaranteed to persist in the long term, the breakout group suggests that the vocabulary authors/hosts analyse the access logs of said vocabularies to ascertain the number of users that actually make use of these namespaces, before performing any changes to them. In the case that these resources are extensively used by the community (e.g., FOAF, schema.org, etc) it may be argued that such statistics be of public interest and ought to be released to the community if possible. This might also indicate adoption rates, which can help enterprises make technology choices.

Another idea that was brought up was studying whether the traditional DNS-based URIs are in fact necessary in the future. Considering that there are technologies such as BitTorrent, which uses DNS-less anchor links for resource resolution, possible similar technologies could be repurposed for the Semantic Web? The breakout group is not very knowledgeable about the internals of BitTorrent technology, but suggests that these types of technologies and solutions be studied also.

3.2 Persistence of vocabularies/datasets

The first recommendation of the breakout group concerning this topic was vocabularies and datasets, once they have been published on a publically resolvable URI, should **NOT** be changed. This is in line with the existing W3C Best Practice Recipes for Publishing RDF Vocabularies[8].

Additionally, the breakout group suggested that easy to understand and easy to interpret change logs between different versions of a particular ontology (published on different namespaces) would be very helpful, to let developers know whether to adapt their systems and dependant ontologies to newer versions of some ontology. As far as the breakout group knows, there is presently no standardised manner of displaying such deltas in a developer-friendly manner, and this would be a valued research contribution. Related to the same issue, there are a number of annotations in OWL supporting versioning, but these are only

[7] http://www.w3.org/2013/04/vocabs/

[8] http://www.w3.org/TR/swbp-vocab-pub/

sparingly used. The group recommends that vocabulary authors learn about and use these annotations to a greater degree.

A further suggestion for future applied research (possibly a MSc project?) was the development of methods and tools to analyze a code base (primarily in Java, as this is the dominant language for Semantic Web development) and produce reports on the technology and resource dependencies of that code base. Such a tool could help developers find dependencies such as vocabularies, software libraries, etc. that they might want to back up or clone into their own version management systems.

3.3 Communication of popular technologies and vocabularies

The breakout group suggests the use of LOV[9] and LODStats[10], which are concerned with exactly this type of work for vocabularies/ontologies.

Concerning technologies and solutions, no equivalent exists. One suggestion was to consider social networking ideas; possibly setting up a community site with a high degree of interaction among users, listing technologies, and providing information about the pros and cons of each, known installations, etc. Presently semantics are discussed on many different places on the web; GitHub, SemanticWeb.org, answers.semanticweb.com, the ONTOLOG community, etc. Perhaps integrating such content into one place, and combining this with a list of well known technologies and installations, could be a useful addition to the community.

4 Complexity Challenges Solution Suggestions

The group suggests that Semantic Web researchers need to become more familiar and comfortable with composing systems using abstraction layers that hide functionality. This necessitates a degree of quality assurance not historically associated with research software, as those abstractions layers, i.e. APIs, will need remain stable and supported over time. The Semantic Web research community has a lot to learn from our practitioner partners, with regards to testing, packaging, and maintaining software!

In terms of practical development, the group suggests employing REST-style architectures, which the practitioner web developer community are already familiar with. Standardising on this type of architecture also allows our software to more easily become interoperable with non-semantics based components and systems, to the benefit of both sides.

The group notes that research in Semantic Web technology is becoming more driven by practical applicability, as exemplified by technologies such as JSON-LD (supporting the use of semantics with existing toolsets), Turtle (supporting RDF that human developers can read), Linked Data Fragments (supporting less

[9] http://lov.okfn.org/

[10] http://stats.lod2.eu/

expressive but very scalable and efficient querying), and schema.org (supporting simple data content schemas that search engines understand). These technologies, and many others, trade a little bit of technical or scientific “purity” for everyday usability by web developers: in the opinion of the group a very worthy tradeoff to make. Some of these developments are being lead exclusively by companies; others by academics with an understanding of enterprise needs. The continuation of the WaSABi workshop series seems highly important for supporting both categories of developers.