Nodalities

THE MAGAZINE OF THE SEMANTIC WEB

talis®

www.talis.com/nodalities

Issue 11

INSIDE

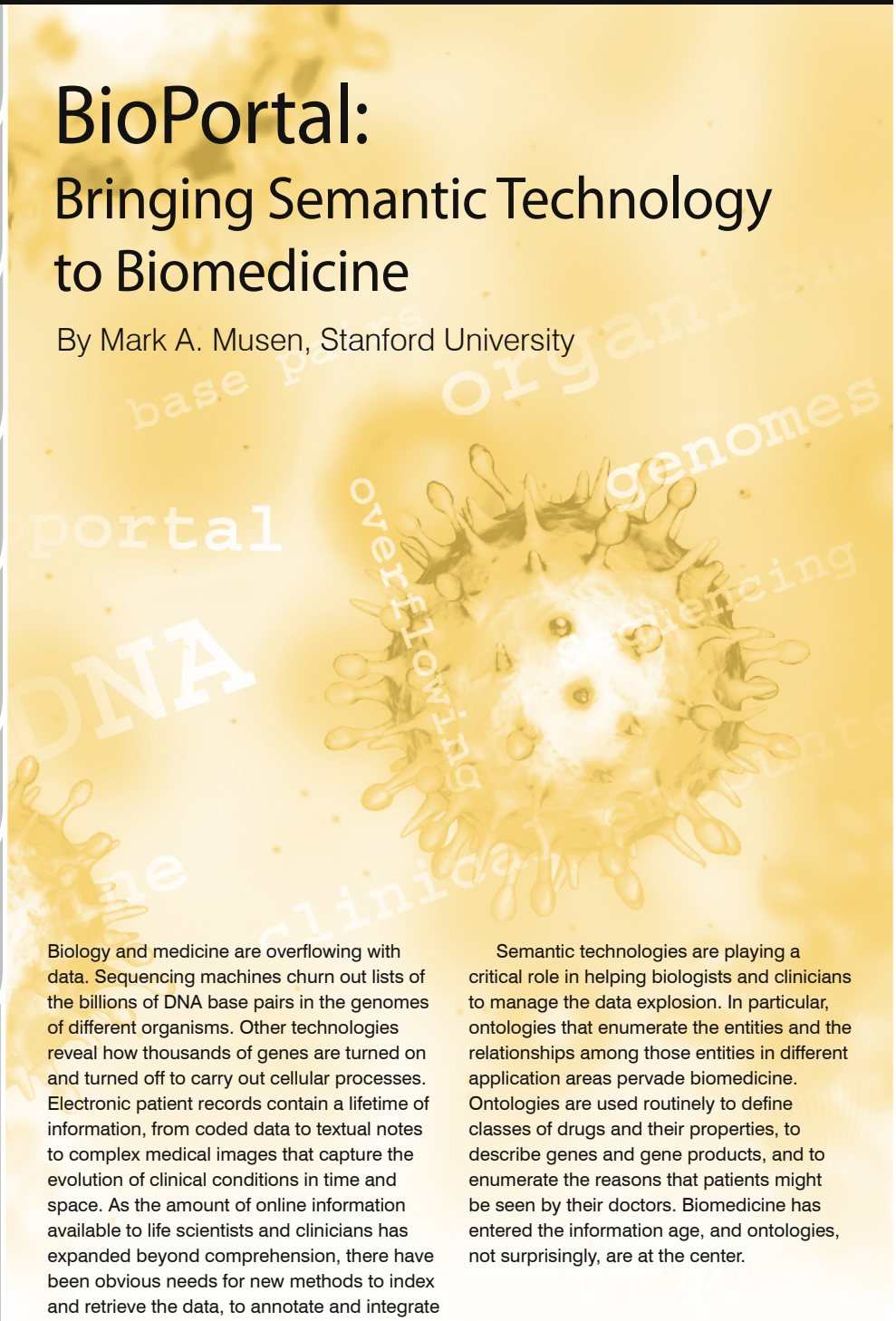
RESEARCH SPECIAL EDITION

twitter

Follow Us On Twitter
@nodalities

BioPortal:
Bringing Semantic Technology to Biomedicine

By Mark A. Musen, Stanford University



Biology and medicine are overflowing with data. Sequencing machines churn out lists of the billions of DNA base pairs in the genomes of different organisms. Other technologies reveal how thousands of genes are turned on and turned off to carry out cellular processes. Electronic patient records contain a lifetime of information, from coded data to textual notes to complex medical images that capture the evolution of clinical conditions in time and space. As the amount of online information available to life scientists and clinicians has expanded beyond comprehension, there have been obvious needs for new methods to index and retrieve the data, to annotate and integrate the data, and simply to allow workers in biomedicine to know what data are out there.

Semantic technologies are playing a critical role in helping biologists and clinicians to manage the data explosion. In particular, ontologies that enumerate the entities and the relationships among those entities in different application areas pervade biomedicine. Ontologies are used routinely to define classes of drugs and their properties, to describe genes and gene products, and to enumerate the reasons that patients might be seen by their doctors. Biomedicine has entered the information age, and ontologies, not surprisingly, are at the center.

continued on page 3

SUBSCRIBE NOW
Nodalities Magazine is made available, free of charge, in print and online. If you wish to subscribe, please visit www.talis.com/nodalities or email nodalities-magazine@talis.com.

MEET US

For further information visit: www.talis.com/platform/events

FOLLOW US

Follow Nodalities Magazine on Twitter: @nodalities

Editor's Notes

For 10 issues, Nodalities Magazine has been telling stories from across the Semantic Web. We've watched from the very first edition (cover: "Looking Ahead to Linked Data on the Web") as the field has grown from just looking forward to publishing, building, fixing, and supporting Linked Data applications, services, ideas and projects. The last editions of Nodalities have shown this, focussing on projects you can visit, services you can try and concepts which underpin something you can see. But there has always been a tremendous amount of energy from the Semantic Web's research community: pushing the boundaries of how we think (quite literally, with semantic questions) and how we apply new ideas to practice.

So, we've launched our first special edition Nodalities, complete with special edition cover stamp, with the intention of drawing attention to some of the folks who continue to push the boundaries. If there's a theme to this, it's probably something like: "Looking ahead, still."

On the cover this time is a look at how ontological thinking and linking are being applied to biomedical datasets. Elsewhere, research teams from across Europe introduce and describe some of their projects: from Søren Auer's team and Semantic Pingback to Adrian Pashke's group looking at content creation and rich tagging.

My colleague, Tom Heath also looks back and forward with his: "Without Linked Data, no Semantic Web!" Bob DuCharme from TopQuadrant also delves into the world of Publishing and Semantics.

Finally, Facebook's David Recordon talks with us about Facebook and the Open Graph—surely one of the biggest uses of graph-thinking being put into practice right now!

As always, if you have a Semantic Web story, get in touch with the editor at zb@talis.com, or follow us on twitter @nodalities.

-Zach Beauvais

Nodalities Magazine is a Talis Publication

ISSN 1757-2592

Talis Group Limited
Knights Court, Solihull Parkway
Birmingham Business Park B37 7YB
United Kingdom
Telephone: +44 (0)870 400 5000
www.talis.com/platform
nodalities-magazine@talis.com

Continued from front page.

Since the 1700s, biologists have used the taxonomy created by Linnaeus to describe each living thing's genus and species. Since the 1800s, clinicians have used the classification scheme developed by the International Statistical Institute to categorize causes of illness and death. Fast-forward to this century, and it now has become nearly impossible to do work in genomics without reference to the Gene Ontology (GO), which provides a comprehensive reference of gene products and their biological functions. For example, scientists routinely use the GO as a source of unambiguous terms for describing the results of their experiments. When experimental results are described using the names of the entities in the GO, computer programs can follow the "is-a" relationships of the ontology up the taxonomic hierarchy to identify the most abstract term that describes all the individual entities. Thus, the ontological relationships can allow biologists to discover that a set of genes are all involved in a process such as cell death, or that all the genes code for proteins that are found in the cell nucleus.

There are now scores of ontologies and controlled terminologies that are used routinely by the biomedical community on a daily basis. It seems it is impossible to do a biological experiment or to manage a clinical encounter without making reference to some ontology.

It seems that it is impossible to do a biological experiment or to manage a clinical encounter without making reference to some ontology.

The National Center for Biomedical Ontology (NCBO) was created in 2005 to help life scientists and clinicians use ontologies to manage the information overload. With support from the National Institutes of Health (NIH), some two dozen researchers, software engineers, and outreach specialists at Stanford University, the Mayo Clinic, the University at Buffalo, and the University of Victoria (in British Columbia) have teamed together to provide a broad range of semantic services to make ontologies more accessible and more useful.

The NBCO is one of several National Centers for Biomedical Computing (NCBCs) supported by the NIH. Each of the NCBCs has been created to provide critical infrastructure to aid a different aspect of biomedical computation. The NCBO is the NCBC most closely aligned with work on the Semantic Web.

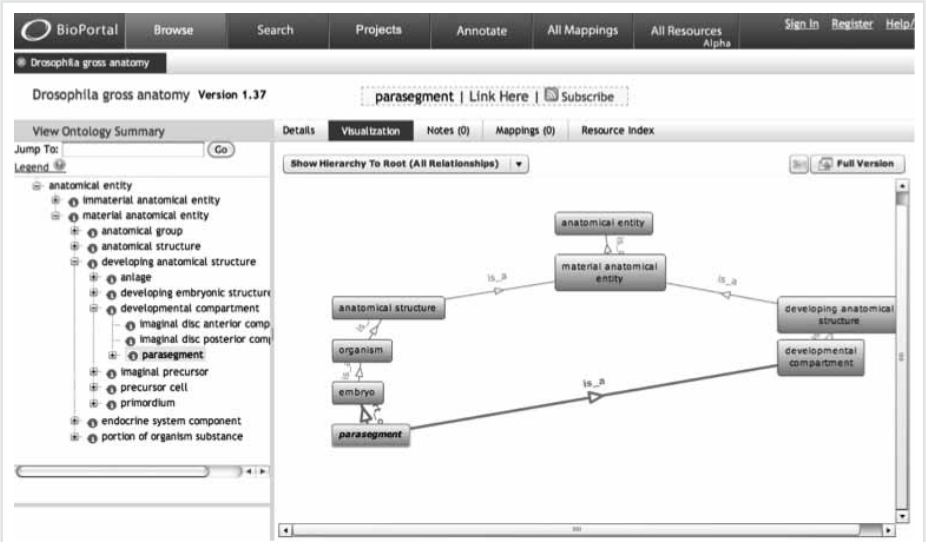


Fig. 1

The NCBO has created and maintains BioPortal, a repository of nearly all the world's biomedical ontologies. NCBO also offers tools and Web services that allow users to access those ontologies and to use them for a variety of applications. The Center additionally supports a program of collaborative investigation, where scientists and technologists from around the world team with NCBO staff to study the application and development of semantic technologies in biomedicine. The combination of research and development, education and training, and collaborative investigation makes the mission of the NCBO broad and challenging. It also makes the NCBO an exciting place to work.

The NCBO's most visible contribution to the biomedical community has been BioPortal, a website that provides access to more than 200 biomedical ontologies and controlled terminologies (Figure 1). Users come to BioPortal to browse biomedical ontologies and to search for specific ontologies that have terms that are relevant for their work. A cancer biologist may learn from BioPortal that the GO offers the best coverage for annotating her experimental data with terms related to cell division, or that more precise terms can be found in the National Cancer Institute (NCI) Thesaurus. She may discover that the Mouse Adult Gross Anatomy Ontology is ideal for describing the body parts from which her experimental specimens were obtained, or that the National Drug File–Reference Terminology provides valuable information about the properties of the drugs used in her experiments.

BioPortal enables users to navigate ontologies using a standard tree browser. Users also can visualize ontologies using special tools that employ novel layouts and animation to offer cognitive support to users trying to understand the complexities of large ontologies within the confines of the two dimensions offered by a Web browser.

When users need to understand the relationships between terms in two different ontologies, BioPortal provides mappings between the ontologies to enable direct comparisons. The mappings can inform the user that the term lung in the Mouse Adult Gross Anatomy Ontology is related to the term lung in the Foundational Model of (human) Anatomy or, for that matter, that the term limb in the NCI Thesaurus is related to the term extremity in the Mouse Adult Gross Anatomy Ontology.

BioPortal enables users to navigate ontologies using a standard tree browser.

BioPortal allows users to search biomedical data using the terms of any of its stored ontologies. Users may be interested in online data ranging from the abstracts of biomedical journal articles available via the National Center for Biotechnology Informatics' PubMed database, functional genomics experiments stored in the Gene Expression Omnibus, results of drug-development experiments stored in ClinicalTrials.gov, or any of dozens of other publicly available data sources. The NCBO has indexed the content of more than two dozen widely accessed online data sources using all the terms in the 200-odd biomedical ontologies stored in BioPortal. As a result, BioPortal users can locate online data of interest by using the

ontologies in BioPortal to search for concepts that range from the most widely known (for example, lung) to the less obvious (for example, extremity).

We have created the system as the nexus of an online community of ontology developers and ontology users who use BioPortal to view, to comment on, and to discuss the content of biomedical ontologies.

BioPortal is more than an ontology repository. We have created the system as the nexus of an online community of ontology developers and ontology users who use BioPortal to view, to comment on, and to discuss the content of biomedical ontologies. Registered users of BioPortal can not only upload new ontology content, but also annotate their content (or that of any other user) with highly granular comments about any ontology. Users can indicate where they believe ontologies may be making inappropriate modeling decisions, and other users can respond to those comments in threaded discussions that the entire BioPortal community can monitor (Figure 2). These threaded conversations allow BioPortal to behave very much like a wiki for making annotations to ontology content, and they enable new users to locate regions of BioPortal's ontologies where modeling decisions have been particularly controversial and ontology developers to identify elements of their work that may benefit from refactoring in future versions of their ontologies. BioPortal also allows users in the community to post overarching reviews of the system's ontologies—and to post online very specific proposals for changes that ontology developers might want to consider in future revisions. BioPortal thus adopts Web 2.0 conventions to allow its users to communicate with one another about the NCBO's hosted ontologies in a highly interactive manner. The outcome of these capabilities is that BioPortal offers the equivalent of online, open, community-based peer review for the BioPortal ontology content.

Although the ontologies in BioPortal mainly support work in medicine and biology, the software is totally generic. Several organizations are using the NCBO's BioPortal code base as the foundation for ontology repositories for other disciplines. BioPortal also serves as the reference implementation of the Open Ontology Repository (OOR) initiative, an effort on the part of an international group of workers in the ontology community who are promoting the idea of open-source solutions for distributed access to ontology content.

BioPortal thus adopts Web 2.0 conventions to allow its users to communicate with one another about the NCBO's hosted ontologies in a highly interactive manner.

The BioPortal is the most visible product of the NCBO, but it may not be the most widely used. Every function that BioPortal achieves in a web browser is also available to users via RESTful Web services, as documented on the Center's wiki. The NCBO Web services are allowing a huge number of developers to perform programmatic actions such as:

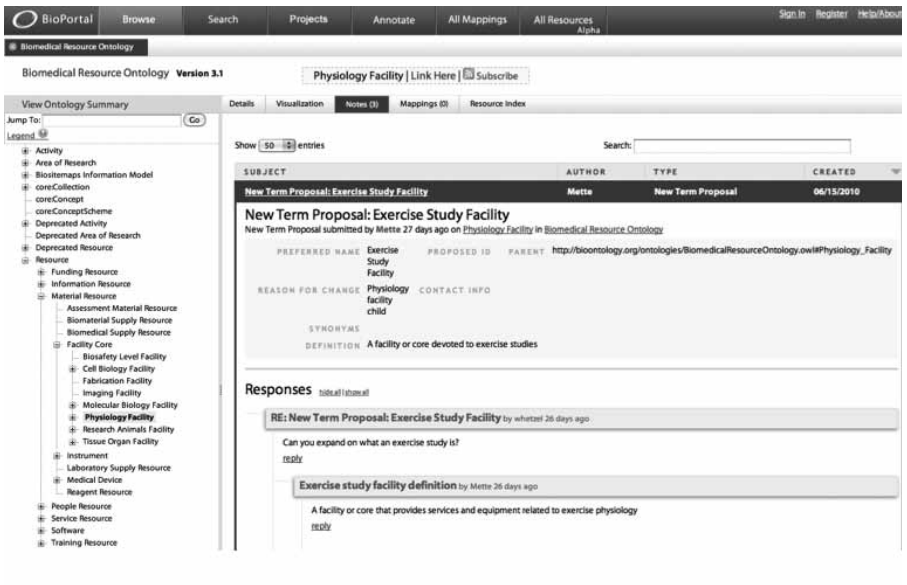


Fig. 2

ontology dissemination, and ontology improvement through an open peer-review process.

In the past century, biology and medicine have evolved from largely descriptive enterprises to scientific activities that are now overflowing with data and information. Making sense of all that information requires ontologies. The NCBO is now at the center of collating and applying ontologies in biomedicine, not only helping the life sciences and clinical medicine to deal with the pending avalanche of data and knowledge, but also providing a new model of how semantics technologies can play an increasing role in all of e-science.

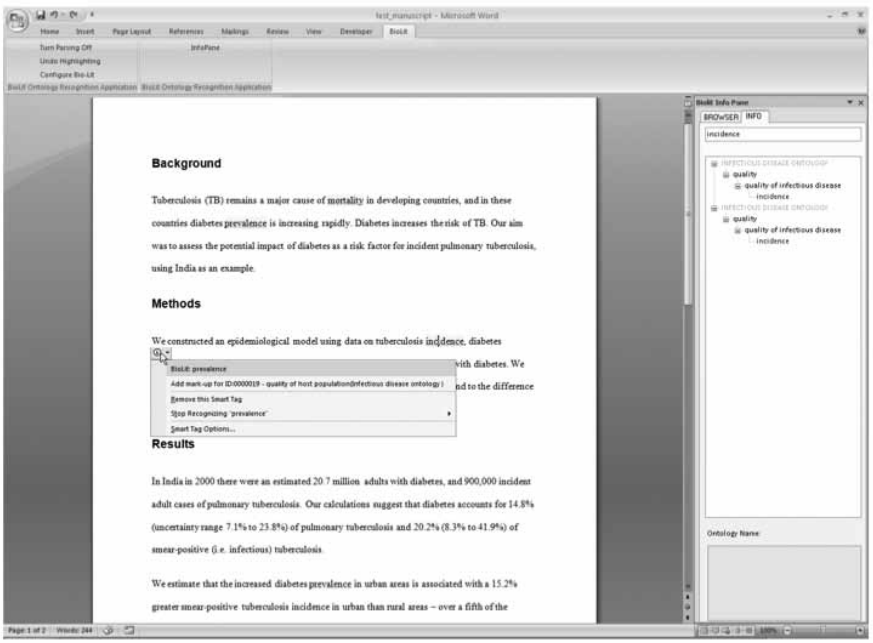
The NCBO is now at the center of collating and applying ontologies in biomedicine

1. Accessing a BioPortal ontology (or set of ontologies) to serve as a lexicon for a natural-language–processing system;
2. Querying BioPortal to generate a list of ontology terms that can form a value set (enumerated data type) to help an end user fill out a Web form
3. Using ontology terms from BioPortal to label online data sets automatically with appropriate ontology terms
4. Asking BioPortal, based on some experimental data to be annotated with ontology terms, which ontology in the repository might offer the best set of terms for performing the annotation (Figure 3).

Scientists developing ontologies such as the Neuroscience Information Framework, for example, find it very easy to use a wiki to review their evolving ontologies.

Increasingly, workers in the e-science community have turned to very generic tools such as wikis to allow online communities to view and comment on terminologies and ontologies. Scientists developing ontologies such as the Neuroscience Information Framework, for example, find it very easy to use a wiki to review their evolving ontologies. They often turn to BioPortal, however, for more advanced ontology visualization, for leaving context-specific notes and comments, and for access to NCBO's Web services by application programs. Although wikis do have the advantage of being easy to use and of being familiar to many scientists and clinicians, the Web 2.0 approach supported by BioPortal offers a wide variety of specialized services that not only facilitate user access to ontology content but also engagement in a broad user community that can take advantage of BioPortal's specialized features to drive an integrated process of ontology authoring,

Fig. 3



Mark A. Musen is Professor of Medicine (Biomedical Informatics Research) and of Computer Science at Stanford University, where he is principal investigator of the National Center for Biomedical Ontology.

Without Linked Data, no Semantic Web!

By Tom Heath

At the recent Extended Semantic Web Conference (ESWC2010), I was asked to sit on a panel and debate the question: "Linked Data: Now What?". It may seem a strange question to ask, when the world at large is just waking up to the opportunities that Linked Data has to offer, but context is key — this was a debate among a community who have been grappling with the Semantic Web concept and shaping its core infrastructure for the best part of a decade.

So where does Linked Data fit into this Semantic Web community? My opening argument at the panel was that without Linked Data there would not, and could not, be a Semantic Web. This claim has two parts. Firstly, from a technical point of view, data can be published using Semantic Web technologies such as RDF and OWL, but if it isn't linked then you don't have a Web. Therefore without Linked Data, no Semantic Web. Secondly, from a historical perspective, without the success of the Linked Data activity, the Semantic Web as a research topic and as a technology strategy for sharing data would be in the wilderness. I think it would have been hard for us to justify ESWC2010, let alone ESWC2015, without demonstrable progress in building a Semantic Web, and Linked Data has given us this.

I expected a number of people in the audience to disagree with my claims, so took the opportunity to reflect on the prevailing mood when I first attended ESWC, back in 2005. Inside the community there was passionate belief in the goals of the Semantic Web, but no one outside took us seriously. Perhaps it was because we did such a bad job of communicating these goals and were reliant on toy examples to demonstrate them. Don't get me wrong, I love pizza and wine as much as the next person, in fact probably more than the next person, but building and classifying pizza and wine ontologies didn't leave me feeling satisfied.

What did whet my appetite was FOAF, the ontology for describing people and and their social networks. This was something for which I could see a real use, and creating a FOAF file was how most budding Semantic Web geeks dipped their toes in the water. These FOAF files, and many more generated by forward-looking social networking services, were the meat of the Semantic Web in 2005. Except it wasn't really that 'Webby'.

These FOAF files weren't really Linked Data as we now know it, they were primarily a series of RDF documents connected by 'rdfs:seeAlso' links — the RDF equivalent of a simple hypertext link that tells you nothing about the connection between two things — and there was one simple reason for this — we didn't know how to do any different.

The FOAF community worked tirelessly to drive enthusiasm for and adoption of the Semantic Web, but this Web couldn't really emerge until we understood how to use URIs to identify things that weren't accessible over HTTP

The FOAF community worked tirelessly to drive enthusiasm for and adoption of the Semantic Web, but this Web couldn't really emerge until we understood how to use URIs to identify things that weren't accessible over HTTP, such as dogs, cats, people, places and abstract concepts. This issue was known as 'httpRange-14' and provoked fierce debates, both technical and philosophical, until a resolution was reached in mid-2005.

Some still dispute the resolution, but what anyone tempted to do so must consider are the relative merits of action versus inaction. The uncertainty before this resolution created a paralysis in the community. The result was inaction and no Semantic Web. What the httpRange-14 resolution provided was the degree of clarity and certainty needed to begin building.

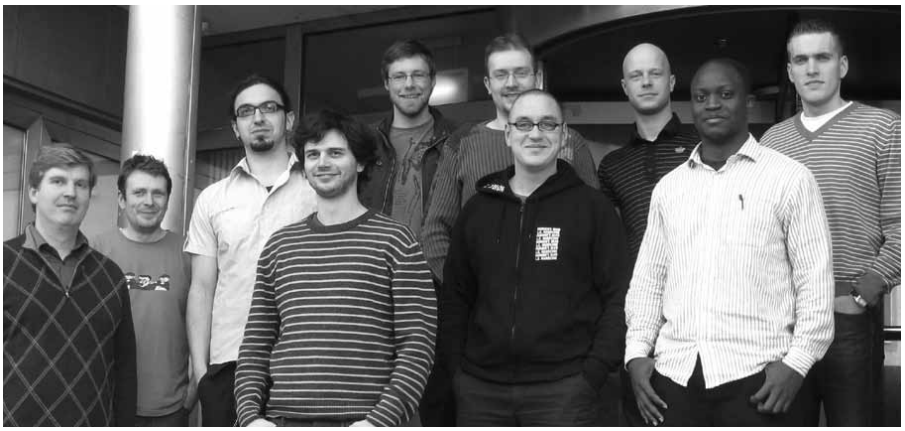
The final piece of the jigsaw came in the form of Tim Berners-Lee's seminal Design Issues document where he outlined four expectations for publishers of Linked Data. These have become known as the Linked Data principles and again provided a point of focus and agreement for people keen to get their hands dirty. The result, several years later, is that we have a Semantic Web we can point to when people ask. Yes, it's messy and inconsistent, but it's real, and it's made of Linked Data.

Dr. Tom Heath is Talis' lead researcher

Semantifying Pingback : Learning from the Blogosphere

By Sebastian Tramp, Philipp Frischmuth, Timofey Ermilov and Sören Auer

Recently, the publishing of structured, semantic information as Linked Data has gained much momentum. A number of Linked Data providers meanwhile publish more than 200 interlinked datasets amounting to 13 billion facts[1]. Despite this initial success, there are a number of substantial obstacles, which hinder the large-scale deployment and use of the Linked Data Web. These obstacles are primarily related to the quality, timeliness and coherence of Linked Data. In particular for ordinary users of the Internet, Linked Data is not yet sufficiently visible and (re-)usable. Once information is published as Linked Data, authors hardly receive feedback on its use and the opportunity of realising a network effect of mutually referring data sources is currently unused.



Once information is published as Linked Data, authors hardly receive feedback on its use and the opportunity of realising a network effect of mutually referring data sources is currently unused.

Semantic Pingback is an approach for complementing the Linked Data Web with a social dimension[2]. It is based on an extension of the well-known Pingback technology, which is one technological cornerstone of the overwhelming success of the blogosphere on the Social Web. The Pingback mechanism enables bi-directional links between blogs and websites in general as well as author/user notifications in case a link has been newly established. Since it enables authors of a blog post or article to obtain immediate feedback when other people reference their work, it facilitates reactions and social interactions. The mechanism also allows automatic publishing of backlinks from the original article to comments or references to the article elsewhere on the Web, thus facilitating timeliness and coherence of the Social Web. As a result, the distributed network of social websites using the Pingback mechanism (such as the blogosphere) is much tighter and more timely interlinked than conventional websites, thus rendering a network effect, which is one of the major success factors of the Social Web. By adding support for typed RDF links on Pingback clients, servers and in the autodiscovery process, the Pingback mechanism is extended towards a Semantic Pingback.

Semantic Pingback is an approach for complementing the Linked Data Web with a social dimension.

The basic idea

When an RDF link from a Semantic Pingback enabled Linked Data resource is established with another Semantic Pingback enabled Linked Data resource, the latter one can be automatically enriched either with the RDF link itself, with an RDF link using an inverse property or additional information. When the author of a publication, for example, adds bibliographic information including RDF links to co-authors of a publication to her semantic wiki, the co-authors' FOAF profiles can be enriched with backlinks to the bibliographic entry in an automated or moderated fashion. Semantic Pingback supports provenance through tracking the lineage of information by means of a provenance vocabulary. In addition, it allows the implementation of a variety of measures for preventing spam.

Architectural Overview

The general architecture of the Semantic Pingback approach is depicted in Figure 1. A linking resource (depicted in the upper left) links to another (Data) Web resource, here called linked resource (arrow 1). The linking resource can be either a conventional Web resource (e.g. wiki page, blog post) or a Linked Data resource. Links originating from Linked Data resources are always typed (based on the used property), links from conventional Web resources can be either untyped (i.e. plain HTML links) or typed (e.g. by means of RDFa annotations). The Pingback client (lower left) is either integrated into the data/content management system or realised as a separate service, which observes changes of the Web resource (arrow 2). Once the establishing of a link was noted, the Pingback client tries to auto-discover a Pingback server from the linked resource (arrow 3). If the auto-discovery is successful, the respective Pingback RPC server is called (arrow 4), with the parameters linking resource (i.e. source) and linked resource (i.e. target). In order to verify the retrieved request (and to obtain information about the type of the link in the semantic case), the Pingback server

1. http://esw.w3.org/topic/TaskForces/CommunityProjects/LinkingOpenData/DataSets/Statistics
2. The Semantic Pingback specification and the example implementations mentioned in this article can be found at: http://aksw.org/Projects/SemanticPingBack

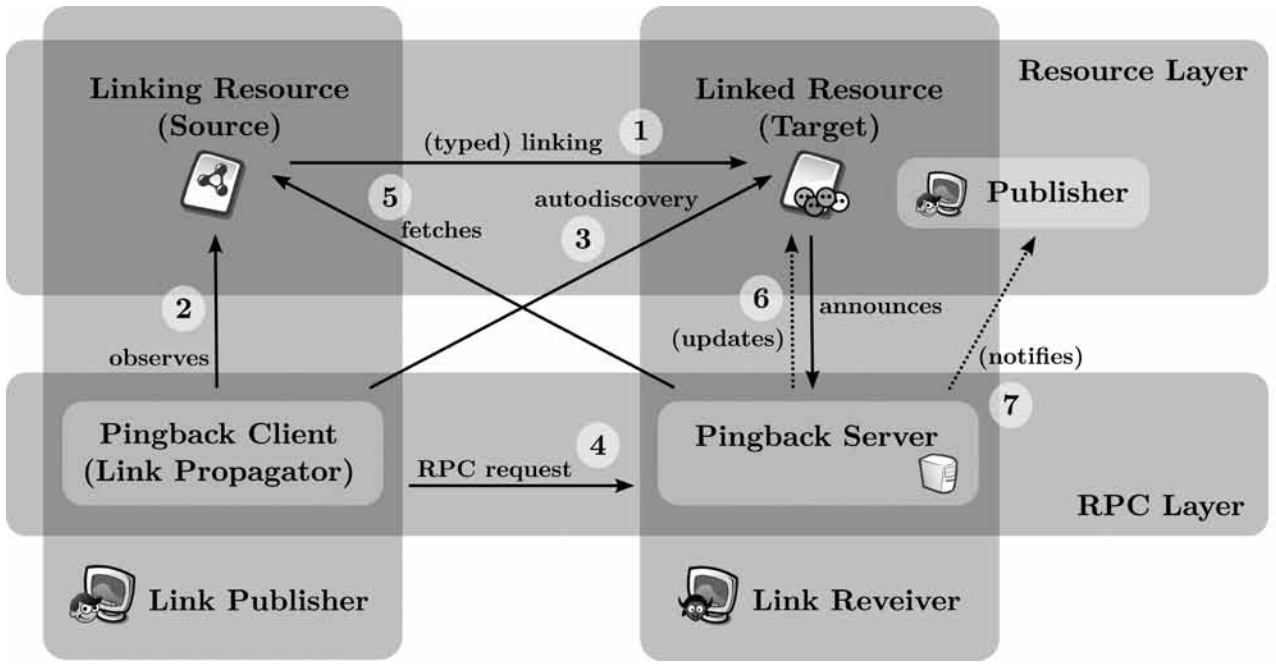


Fig. 1.

fetches (or dereferences) the linking resource (arrow 5). Subsequently, the Pingback server can perform a number of actions (arrows 6,7), such as updating the linked resource (e.g. adding inverse links) or notifying the publisher of the linked resource (e.g. via email).

Compatibility

Semantic Pingback is completely downwards compatible with the conventional Pingback implementations, thus allowing to seamlessly connect and interlink resources on the Social Web with resources on the Data Web. Semantic Pingback adds to the auto-discovery process an additional auto-discovery method (in addition to HTTP and HTML header attributes). This method allows the retrieval of the Pingback server information by means of an OWL object property service[3], which can be directly embedded into the RDF data. The downwards compatibility ensures, that a weblog author can, for example, refer to a certain Data Web resource, while the publisher of this resource can get immediately notified and rdfs:seeAlso links can be automatically added to the Data Web resource.

Integration into OntoWiki

OntoWiki is a tool for browsing and collaboratively editing RDF knowledge bases. Since OntoWiki enables users to add typed links on external resources, we integrated a Semantic Pingback client component. A recently added feature is the ability to expose the data stored in OntoWiki via the Linked Data mechanism. Based on this functionality, a Semantic Pingback server component was also integrated. Each time a statement is added or removed, the Pingback client first checks, whether the subject resource is a URI inside the namespace of the OntoWiki environment or the subject resource is (anonymously) accessible via the Linked Data mechanism[4] and the object of the statement is a resource with an dereferenceable URI outside the namespace of the OntoWiki environment. If the above steps are successfully passed, the plugin tries to auto-discover a Pingback server and sends an XML-RPC post request. In the other direction, OntoWiki announces its Pingback server functionality by exposing the X-Pingback HTTP-header in conjunction with the URL of the RPC service, as well as a providing the Pingback XML-RPC service. By using the versioning functionality of OntoWiki, provenance information of statements added via Pingback requests can be determined, thus allowing the service to delete statements that are no longer contained by the source resource. Backlinks that were established via the Pingback service are displayed in the standard OntoWiki user interface. The “Instances Linking Here” box shows all incoming links for a given resource in conjunction with the type of the link.

Semantic Pingback is completely downwards compatible with the conventional Pingback implementations, thus allowing to seamlessly connect and interlink resources on the Social Web with resources on the Data Web.

Integration into Triplify

Triplify enables the publication of Linked Data from relational databases. It utilises simple mappings to map HTTP URLs to SQL queries and transforms the relational result into RDF

3. http://purl.org/net/pingback/service
4. This step is added to the process since OntoWiki is able to handle various access control mechanisms and we thus ensure that the Pingback server of the target resource is definitely able to access either the RDF or the (X)HTML representation of the source resource.

statements. Since a large quantity of currently available web data is stored in relational databases, the number of available Linked Data resources increases. As people start to link to those resources, it becomes handy to notify the respective owner. Therefore, we integrated a Semantic Pingback server into Triplify, which exposes an X-Pingback HTTP header and handles incoming RPC requests. The RPC service creates a new database table and stores all registered Pingbacks persistently. Pingbacks are unique for a given source, target and relation and hence can be registered only once. Each time the Pingback service is executed for a given source and target, invalid Pingbacks are removed automatically. Triplify was extended to export statements for all registered Pingbacks regarding a given target resource along with the instance data.

In order to equip file-based Linked Data resources like FOAF profiles with Semantic Pingback capabilities, we developed a standalone Semantic Pingback server.

Standalone Pingback Server

In order to equip file-based Linked Data resources like FOAF profiles with Semantic Pingback capabilities, we developed a standalone Semantic Pingback server. In conjunction with the above mentioned OWL object property service, one could simply add a triple to her FOAF profile and point it to a running version of the server, such as the AKSW Semantic Pingback service[5].

Although the Data Web is currently substantially growing, it still lacks a network effect as we could observe for example with the blogosphere in the Social Web.

A brief Outlook

Although the Data Web is currently substantially growing, it still lacks a network effect as we could observe for example in the blogosphere on the Social Web. We hope Semantic Pingback will help to improve the coherence on the Data Web, since linking becomes bi-directional. With its integrated provenance and spam prevention measures, it helps to increase the information quality. Notification services based on Semantic Pingback increase the timeliness of distributed data. In addition these different benefits will mutually strengthen each other and hopefully contribute to establishing the mentioned network effect around Linked Data. Due to its complete downwards compatibility Semantic Pingback bridges the gap between the Social and the Data Web. Thus it could significantly support the transition process from data silos to flexible, decentralised structured information assets. Currently, the mechanism is applicable to more or less static resources, (i.e. RDF documents or RDFa annotated Web pages). It could be extended in such a way, as to be also usable in conjunction with dynamically generated views on the Data Web–i.e. SPARQL query results. This would allow end users as well as applications using remote end users SPARQL endpoints to get notified once results of a query change.

Semantic Pingback is a project of AKSW research group at Universität Leipzig and involves work from Sebastian Tramp, Philipp Frischmuth, Timovey Ermilov and Sören Auer.

5. http://pingback.aksw.org

Two Planes, an IT-related Sitcom, and Lessons in Platform Engineering

By Michael Fitzmaurice, Talis Platform Engineering & Operations

After setting off from Digbeth coach station some 25 hours earlier, two tired Talisians finally arrived in; Santa Clara for O'Reilly's **Velocity 2010** conference for Web Performance and Operations. We first flew from Heathrow to Dallas Fort Worth on a surprisingly cramped American Airlines flight, then onto San Francisco. By the time we hit SFO, we were looking forward to getting to the hotel to freshen up and sleep. Night had fallen as our plane approached SFO which made a nice change after nothing but daylight since waking up in Birmingham well over a day before. Not a journey for vampires, I can tell ya.

> *The Valley and Bay Area are chock full of people involved in one area or another of the tech industry.*

On the flight from Dallas to San Francisco I sat next to Patrick Wilson, CEO of Vitalsigns Technology[1] who was on his way back from a conference in Florida. The Valley and Bay Area are chock full of people involved in one area or another of the tech industry. Patrick has a pretty extensive knowledge of the area and served as a high altitude tour guide of sorts as we made our approach over the lights of San Francisco ("There's the Googleplex. That complex there is Sun," and so on) . He also revealed a recently-acquired taste for the Channel 4 sitcom "The I.T. Crowd", even going as far as to fire up his laptop and share an episode. I was more than happy to help him with his query as to the "exchange rate between pounds and quid?" given his obvious delight at one of our most surprising cultural exports.

We knew a cab from SFO to Santa Clara would run us well over $100; Irish genes kicked in and we started looking for a cheaper alternative, despite our by now zombie-like state of fatigue. At this point we discovered the magic of SFO "Shuttles". These are basically big truck-like minivans that provide a (relatively) cheap way to get from the Airport to anywhere in the general Bay Area. When enough people want to go in roughly the same direction, you jump in and off you go. Cheaper than a cab and organised to cover broad geographical regions (e.g. South Bay area) there is an element of pot luck controlling how long it will take you to reach your destination depending on the route the driver chooses in order to encompass all stops. The driver relied heavily on his sat nav, which caused some concern when it failed spectacularly. He switched it for an identical one that worked: prompting lame, tired jokes from us about redundancy and switching to a warm standby. First performance lesson of the trip: geek humour degrades dramatically after 26 hours without sleep.

A Serendipitous Meeting between Scribes

On Sunday, we had dinner with fellow Velocity attendees Patrick Debois and Torben Graversen. My colleague met Patrick at Puppet Camp a few weeks ago. Patrick is the originator of the term "Devops"; and his blog is highly recommended. As we chatted in the bar afterwards, Sean Power approached and asked if we were here for Velocity. He introduced himself and his friend Tracy Lee.

We chewed the fat for a while and talked lean start-ups, performance monitoring, Silicon Valley, and Sean's upcoming talk at Velocity. Sean mentioned that he had contributed to an O'Reilly book due out any day now; and Patrick asked which one. It turned out to be "Web Operations"; this book contains a chapter on Monitoring written by non other than one Mr. Patrick Debois! You could have knocked us over with an O'Reilly "In a Nutshell" book, such was the strength of the minor coincidence.

Lack of nerve scuppers a tour of Twitter (or possibly just a beating from security)

Since the conference didn't start until Tuesday, we hired a car on Monday and drove to San Francisco. Whilst there we met with a couple of friends of Patrick, one of whom was something of a veteran of the Bay Area Tech scene. He told us he had calculated that there were at least 400 tech companies in a 3-block area of San Francisco; pretty mind-blowing when you think about it. He also informed us the Twitter offices were just around the corner and suggested we should go in, ask at reception for his friend John Adams, tell him we were here for Velocity, ("Mike told us to ask for you") and see if he would give us a tour.

> *We were all pretty surprised to find we were able to stroll into the Twitter building unannounced, take the lift up to the 6th floor, wander into reception and hang around without anybody once challenging us*

We were all pretty surprised to find we were able to stroll into the Twitter building unannounced, take the lift up to the 6th floor, wander into reception and hang around without anybody once challenging us; I kept expecting to be thrown out. Ultimately, we were all far too European and reserved to ask for someone we didn't know, tell him we were friends of somebody else we didn't really know ("Does anybody know that guy Mike's surname?"), and cheekily ask for a tour of Twitter, so we just hung around for a bit looking goofy and then left. So much for the meek inheriting the Earth; we couldn't even blag a tour of Twitter.

> *At first glance some folks may not see the connection between these two topics, but they are increasingly intertwined as engineers seek to build highly available, scalable and fast applications that operate at Internet scale.*

Down to Business

Velocity traditionally covers two broad

1. http://www.vitalsignstechnology.com/

areas: performance of Web applications and operations.

At first glance some folks may not see the connection between these two topics, but they are increasingly intertwined as engineers seek to build highly available, scalable and fast applications that operate at Internet scale. Here at Talis, our development and operations functions work together closely in the same team, so it made perfect sense to us that these tracks had been combined into a single conference looking at performance in a holistic way. It also seemed fitting for us to send one developer and one ops person.

The conference was sold out, with over 1200 people in attendance, and up to 3 tracks at once at various times. Between the two of us, we tried to arrange our schedules to cover as many of the presentations and sessions as we could. Some of the sessions were billed as "workshops", but in reality they were way too big to be anything other than long presentations—400 people is far too many for anything "workshoppy". Nevertheless, the content was generally of a very high standard; informative and well presented.

Engineering for the win!

One of my favourite presentations came from Theo Schlossnagle. Theo's "Scalable Internet Architectures" was 90 minutes of wisdom covering a vast array of material, from analysing network packet size, to choosing between SQL and NoSQL databases, to version control, caching, monitoring, service decoupling, mastering tools and the importance of engineering maths. A truly wide-ranging and ambitious presentation, skilfully delivered. Unfortunately there appears to be no video, so you can't really appreciate the moments when Theo worked himself into a righteous engineering rage as he dismissed various bone-headed architectural decisions. However, the slides are still well worth a look.

Theo's "Scalable Internet Architectures" was 90 minutes of wisdom covering a vast array of material, from analysing network packet size, to choosing between SQL and NoSQL databases, to version control, caching, monitoring, service decoupling, mastering tools and the importance of engineering maths.

How do they do that?

Undoubtedly the most over-subscribed session of the week was "A Day in the Life of Facebook Operations" by Tom Cook. I literally had to watch this one while standing in the doorway to the lecture theatre. The room was full to bursting point and Tom did not disappoint. The sheer scale of the job at Facebook is daunting; more than 500 million users, 10s of thousands of servers, 300+ TB of data served from RAM alone via Memcached, and multiple software releases and configuration changes every single day across this gigantic stack. A great example of operating on a massive scale and yet still moving quickly and keeping risk small.

The sheer scale of the job at Facebook is daunting; more than 400 million users, 10s of thousands of servers, 300+ TB of data served from RAM alone via Memcached, and multiple software releases and configuration changes every single day across this gigantic stack.

Similarly popular and insightful were John Adams' "In the Belly of the Whale: Operations at Twitter", John Allspaw's "Ops Meta-Metrics: The Currency You Use to Pay For Change" and Paul Hammond's "Always Ship Trunk: Managing Change In Complex Websites". All of these presenters have real, in-the-trenches experience of managing development and operations in very large, very fast moving Web applications, servicing mind-boggling numbers of users via staggering amounts of code and infrastructure. Much can be learned from them.

There is no spoon

Wedged in amongst all the good stuff on performance in the browser there were a couple of sessions that took different approaches to looking at performance. Firstly, there was Yahoo Search's Stoyan Stefanov with "The Psychology of Performance", offering fascinating insights into how humans perceive the duration of various things and what that means for web applications.

Secondly, Neil Gunther and Shanti Subramanyam used performance testing analysis of Memcached in "Hidden Scalability Gotchas in Memcached and Friends" to introduce Neil's Universal Scalability Law and explain what mathematical modelling can do to help performance tuning in the Brave New World of multi core machines. This was truly eye-opening stuff; the material was accessible enough to pull you in, but deep enough that I will be digesting bits of it and delving into this further for a long time to come. It was also good to see that server-side performance was being addressed at Velocity, albeit on a much smaller scale than the browser-side.

A recurring theme for me was the additional material Velocity has pointed me towards—the performance-related blogs of Neil and Shanti being great examples.

What has all that got to do with the Talis Platform?

Common high-level threads amongst all these cool kids on the Tech block were being process-light but review-heavy and making frequent small changes with enough testing, automation and monitoring around them to keep the risk of change minimal, yet keep the pace punchy. I found it encouraging to see how much of this stuff we already do:

- version controlling everything (Subversion and Git)
- always shipping trunk
- using configuration management tools (Puppet)
- stressing peer review
- extensive automated testing (J-Unit, Grinder)
- monitoring and alerting (Ganglia, Nagios, Cacti, etc.)
- Continuous Integration (Hudson)
- dark deployment
- service decoupling
- using switches in code to enable/disable features
- frequent small releases
- appropriate use of asynchrony
- judicious use of cloud technologies (EC2, S3 and various other bits of AWS)
- having ops and devs work closely together.

We don't yet face the problems of scale that have led Facebook and Twitter to turn to BitTorrent as a means to roll out software quickly to thousands of servers. However, I left feeling confident in the way we work, primed with new areas for us to explore, and inspired at having gained an insight into how some of the leading lights of Internet-scale engineering make it all hang together.

Michael Fitzmaurice is a software developer in Talis' Platform Engineering & Operations team

We're always on the lookout for talented and pioneering people to join with us and help us push our boundaries even further.

If you're interested now, or could be in the future, please visit our website where our vacancies are advertised or to send a speculative application, please use the email address below.

talis.com/careers
catherine.mayo@talis.com

talis® Platform™
data, connected

The Data Publishing Three-Step

By Richard Wallis, Technology Evangelist at Talis

In a conversation with data owners about how they should be publishing their data, it is usually not long before the following question turns up: "So, what do I actually have to do to publish my data?" Often the conversation then wanders off into a game of buzzword bingo–RDF, RDFa, SPARQL, dereferenceable URIs, triples, content negotiation, open data, Linked Data, end-points, etc.—to be followed by a blank look and the unuttered question "Yes, but what do I actually have to do to publish my data?"

In an attempt to simplify the answer to that oft unuttered question, I break things down in to three steps.

In a conversation with data owners about how they should be publishing their data, it is usually not long before the following question turns up: "So, what do I actually have to do to publish my data?"

Step 1 Get your Data Out–for others to consume

Sounds simple. Just take the spreadsheet (or similar file) that you use to track information, post it on your web site and link to it from a description posted in an accompanying web page. It can be that simple, but there are things to consider:

- **Licensing**–will potential consumers of the data be confident on their ability to use and/or reuse it. (The UK Government are very clear on this)
- **Is it open but opaque?**–The terms, codes, identifiers etc. you use may be meaningless, or worse still ambiguous, to those outside your organisation, or even your department.
- **Could your data be made more consistent with other data you, or similar organisations, already publish?**

All things to be considered, but not to be put up as excuses for not publishing.

I've decided to go through these steps, can you remind me again why? So that your data can be linked with other data to add value to the experience of consumers of your data and services, as well as others using your data to add value elsewhere—a good example of this in action being the BIS Research Funding Explorer.

Step 2 Get your Data In–to an open linkable standard format

This is the most powerful step, which consists of identifying the elements in your data (organisations, locations, things, projects, types, etc.) and giving them unique identifiers then make these identifiers web links. Fortunately this may not be as onerous as it sounds. There are many publicly visible/usable identifiers that you can use for your data—for example:

- http://dbpedia.org/resource/Birmingham
- http://education.data.gov.uk/doc/school/514588

For this step to be effective you really need to be modelling your data: Your (first class) data elements, and the relationships between them; plus possibly relationships with external entities. The output of this step will be an RDF representation of your data to Linked Data Principles. You should also identify the process or rules to get from your source data into this new form, enabling you to repeat for later versions of your data.

Having said all that, it is not necessarily only you that will/can do step 2. It is perfectly possible for a third party, or a central organisation such as data.gov.uk, or even an enthusiast, to carry out this data modelling and transformation step with data that you have openly published.

Next you need to publish your data so that it can become part of the Web of Linked Data, which brings me, with apologies to fans of the traditional party song, to....

Step 3 Link it all about

How to publish this data is the next question, to which there are at least three equally valid answers.

Using an encoding technique called RDFa, you can embed the RDF data within the html coding of a web page so that software can obtain a more structured representation from a web page than a human, viewing it in a browser would.

You could just publish the RDF in rdf files on your web server. A good example of this is the way the BBC publish the RDF for many of their pages, such as for their Wild Life. The Lion Web page–the RDF for Lion (dependant on your browser, you may need to use it's view page source option to see the actual RDF encoded in XML)

You could store the individual RDF statement (triples) in a triple store, or SPARQL end-point. This not only publishes the RDF, but also enables the data and relationships within the data to be queried. This is how data.gov.uk publishes RDF, from Talis Platform Stores. This interface might look a bit cryptic—the results, formatted in XML in the top box, from running the SPARQL query shown in the bottom box–but this is a developers interface demonstrating the code and results an application might use, so you wouldn't expect much different.

I've decided to go through these steps, can you remind me again why? So that your data can be linked with other data to add value to the experience of consumers of your data and services, as well as others using your data to add value elsewhere—a good example of this in action being the BIS Research Funding Explorer.

Semantic technologies and the publishing world: what can they offer each other?

By Bob DuCharme, TopQuadrant

We've all seen the news stories about the publishing industry's declining revenue. If you follow the publishing or semantic technology worlds, you've probably also seen stories about how semantic technology can save the publishing industry. Look more closely at these latter stories, though, and you'll see that purveyors of the latest "semantic content solutions" often have a narrow view of the both the problems to solve and the extent of the help that semantic technologies can offer. Semantic technology can help the publishing industry much more than they realize, and as it turns out, the publishing world has a lot that can help the semantic technology world.

Semantic technology can help the publishing industry much more than they realize, and as it turns out, the publishing world has a lot that can help the semantic technology world.

By publishing "world", I mean more than companies who create books for book shops and newspapers and magazine for news stands. These same companies are typically trying to earn income in additional media such as electronic books, subscription or ad-based web sites, and even CD- and DVD-ROMs. The larger publishing eco-system includes libraries, who continue to be huge consumers of books and other media, and companies whose sole purpose is to help libraries and publishers track the metadata that helps them manage their common interests.

Semantic technology people and the publishing world are actually dealing with many common issues, but they approach it from different directions and use different terminology to describe their work. It's like the classic parable about the blind men and the elephant, in which several blind men feeling different parts of an elephant each think that they've found a different non-elephant object, but in this case, the elephant is metadata management.

Who needs help doing what?

The publishing industry needs two classes of help. Let's look at two use cases that demonstrate them.

The publishing industry needs two classes of help.

Imagine that the fictional Snee Publishing Group draws on a collection of 120,000 recipes to publish different sets of them in different media. In our first use case, customer Sam Palumbo has promised to bring a dessert to his office holiday party, where employees are encouraged to bring dishes related to their family's heritage. He goes to sneerecipes.com, where the recipes have been tagged with several keywords each, and enters the search terms "Christmas", "dessert", and "Italy". The website responds with links to recipes that have been tagged with these three keywords. With a bit more semantic technology at work, a taxonomy can help the search engine know that recipes tagged with the words "dessert", "Christmas", and "Umbria" should be included in the results, because Umbria is part of Italy.

This is the less interesting of my two use cases. It's the one that gets all the attention, though, because everyone agrees that better metadata will help customers find the content they want and help advertisers find the customers they want, resulting in more revenue. By "metadata", most discussions of these issues refer only to keywords that help to summarize content, like "Italy" and "Christmas" and "dessert". Automating the addition of these keywords is much more difficult than writing a regular expression to search for phone numbers, but there are companies out there who claim that their software will add appropriate keywords to your content so that your customers will more easily find the content that they want.

Let's look at the other use case: Snee Publishing staff member Jane Jones must prepare a hardcover book of Italian Christmas dessert recipes. She obviously needs the same help identifying candidate recipes that Sam needs, but keywords identifying content subject matter only address Jane's first of several problems.

Publishing staff and publishing problems

People buy hardcover cookbooks so that they can make the dishes described by the recipes, but they also buy them so that they can relax in a big chair and just think about making those dishes, and nice pictures are essential for this experience. The Snee Publishing Group has pictures accompanying many of their recipes, and their XML-based content management system makes it relatively easy to ask for a list of Italian Christmas dessert recipes that have accompanying image files, but Jane's task is more complicated than that. Her publishing company acquired some of these recipes and images for web or CD publication, so the images are only stored at 72 dots per inch, and images for a hard copy book should be at least 300 DPI.

Querying for Italian Christmas dessert recipes with images that have a resolution of 300 DPI or higher is not particularly difficult for an XML-based repository either—if all the images store resolution information using the same standard. Snee Publishing acquired these images from different suppliers, though, and they supplied metadata using different standards: the Japan Electronic Industries Development Association EXIF standard, Adobe's XMP, and the U.S. Library of Congress's MIX format.

An even tougher problem is digital rights management. (I mean "management" quite literally here; most discussions of DRM are really about digital rights enforcement, or preventing people from making unauthorized copies.

An even tougher problem is digital rights management. (I mean "management" quite literally here; most discussions of DRM are really about digital rights enforcement, or preventing people from making unauthorized copies. Publishers are more concerned with simply tracking which rights are associated with which content.) Jane worries that some of the candidate recipes and images for the new

cookbook may have been licensed to Snee Publishing under terms that do not allow their use in a printed book, or only allowed their use in a time period that has since expired. She definitely doesn't want to use any content in a 2010 hard copy cookbook that falls outside of the content's license agreement, because she doesn't want to jeopardize a relationship with a trusted supplier. She just wants to sort out the ones that she's allowed to use as quickly and easily as keywords let her sort out the Italian recipes from the French and German ones. As with image metadata, there are standards that can provide structure for digital rights management metadata—quite a few of them, in fact, such as PRISM, ONIX, XACML, ODRL, ccREL, and ACAP.

Publishers typically keep this kind of content in Content Management Systems, with images often stored in a more specialized Digital Asset Management system. These are all metadata-driven, often with a bit of flexibility, but not not nearly enough to to support Jane's needs

For most such repositories, standards support means support for one particular standard, and the "flexibility" of the metadata structures in a given CMS or DAM could simply be a few extra fields that you get to define yourself. This is not nearly enough to accommodate the use of multiple standards at once.

For most such repositories, standards support means support for one particular standard, and the "flexibility" of the metadata structures in a given CMS or DAM could simply be a few extra fields that you get to define yourself. This is not nearly enough to accommodate the use of multiple standards at once.

Semantic technology: store all the data, track multiple standards

Before putting a customisable CMS into production—or a relational database, or a set of XML documents belonging to a particular document type—step one is defining the structure of the data and metadata to store. If the Snee Publishing Group uses a typical CMS and has an automated process pulling eight particular pieces of metadata from the images supplied by a given vendor, and that vendor starts adding in two more pieces of metadata for each image, Snee Publishing will probably ignore these new fields at first. If several other suppliers start populating the same two fields in their images, the Snee staff might want to start using this new metadata. They'll have to redefine their schema and their data extraction routines, though, and maybe do a data reload after implementing the schema redefinition. They can then start storing the values from these two new fields, but all the metadata that arrived in these fields up to this point has been ignored and therefore lost.

Storing metadata using a triplestore offers much greater flexibility. These data stores are built around the RDF data model, which lets you store any data expressed using the three-part {subject, predicate, object} model that makes up a single RDF statement. (Those with a more traditional database background can think of the three parts of a triple as a resource identifier, an attribute name, and an attribute value.) Using one of the growing number of commercial and open source triplestores, a publisher can store all the metadata from every image or document as it comes along, whether your system knew about the properties defined for each one beforehand or not.

When an RDF-based system does know in advance about all or some of the metadata properties being used, it can do even more. An RDF Schema or OWL ontology (both of which are simply additional triples to store) adds further metadata about the content and image metadata that lets you track their relationships and use them together more easily.

Before putting a customisable CMS into production—or a relational database, or a set of XML documents belonging to a particular document type—step one is defining the structure of the data and metadata to store.

For example, image resolution is typically tracked with two properties: X resolution and Y resolution. Adobe's XMP specification refers to the concept of X resolution with the URI http://ns.adobe.com/tiff/1.0/XResolution/XResolution, and EXIF and the Library of Congress MIX standard refer to it as http://www.w3.org/2003/12/exif/ns#xResolution and http://www.loc.gov/mix/v20/xOpticalResolution, respectively. Using the OWL standard, you can assert that these three properties mean the same thing, which would let a single search look for images with an X resolution value above 299, even if different images stored this value using terminology from different standards. (In practice, defining these properties as equivalent makes assumptions about the standards that could complicate things later on; a better approach for Snee Publishing would be to define their own http://www.snee.com/ns/imagemetadata#xResolution property, declare the other three as subproperties of that for the purposes of their system, and then search for all images whose http://www.snee.com/ns/imagemetadata#xResolution value was greater than 299. This would have the same effect as declaring them equal, but in a local, more manageable context.).

You could employ a similar strategy with other image metadata properties and with the properties tracked by digital rights management standards. Most other property relationships will not be as simple to manage as an image's X resolution, but thinking through their relationships in the context of your system and then defining them using this standardized metadata can have an even greater payoff for the more complex standards.

What can the semantic web world learn from the publishing world?

Library science professionals have been refining systems for managing the metadata associated with published works since the early thirteenth century, when location lists to help find books within a library started appearing. In modern times, they've worked with the publishing industry to define and use the appropriate metadata so that people can efficiently find the information they need regardless of the storage media. (Keep in mind that in addition to the staff at your local public library, this includes the professionals at technical and legal libraries in private and government organizations where meaningful metadata has a direct effect on cost management.)

When semantic web technology developers create tools and schemas for metadata management, the best use of those tools is often an afterthought. While someone designing a schema for a relational database or an XML document type wouldn't dream of publishing a schema without sample data to demonstrate how it is used, this is quite common in the OWL world. A surprising number of ontology developers don't realize that sample data makes the use of the metadata structures they've defined much easier to understand.

Working with the publishing world, ontology developers can find real data and associated metadata to test their theories, their ontologies, and their tools. For domains such as medical and legal research that seem separate from the publishing world, remember that there are

specialized publishers for these domains, and there are projects that seek to provide free and low-cost alternatives to the often expensive electronically published products offered by the leading publishers. Semantic web developers who seek out these projects can help them and help their own work by using real-world data to test their ontology and software development work.

The semantic web world can also learn a lot from people who manage keyword vocabularies for published metadata. These taxonomists, who typically have degrees in library science (now known as "information science" at many schools) come from a long tradition of defining terminology and terminology relationships to make the associated content easier to use. If we look at an ontology as a taxonomy with additional, domain-specific properties and relationships defined between the stored terms, we can see that taxonomy development best practices can teach a lot to ontology developers. Two great books are Elaine Svenonius's "The Intellectual Foundation of Information Organization" and Patrick Lambe's "Organising Knowledge: Taxonomies, Knowledge, and Organisational Effectiveness", and discussions on the TaxoCoP (Taxonomy Community of Practice) mailing list are open to anyone. When a posting there requests advice on the implications of splitting, joining, or categorizing a particular term, the answers from some of the leading names in the field can be fascinating.

If we look at an ontology as a taxonomy with additional, domain-specific properties and relationships defined between the stored terms, we can see that taxonomy development best practices can teach a lot to ontology developers.

Getting back to the analogy of the blind men and the elephant, we have to remember that the semantic technology and library/publishing communities often have their own terminology for discussing metadata management. For example, I recently saw the phrase "Automated capture, crosswalk, and enhancement of publisher ONIX metadata" in a slide in a presentation by Renee Register, Global Product Manager of Online Computer Library Center (a.k.a. the OCLC, the group responsible for the Dublin Core vocabulary). I knew that ONIX was an international metadata standard for book publishers that covers typical book metadata such as the title and author as well as things like book jacket blurbs and cover art, but I didn't know what a crosswalk was. Wikipedia describes it as something that "maps the elements in one metadata scheme to the equivalent elements in another scheme.".

As I explained to one taxonomist, semantic web technology people may not know the term "crosswalk", but if you explain it to them, they will understand the concept very quickly. They would call it a collection of owl:sameAs relationships, although OWL offers additional properties to account for more nuanced relationships between terms that aren't exact equivalents. This provides a great example of the payoff of learning taxonomist vocabulary, because the ability to discuss metadata management using their own terminology lets semantic web developers explain the extra capabilities that they can bring to metadata management tasks that taxonomists had been working on.

Baselines for looking at the elephant

In addition to seeking out common vocabulary to discuss the issues, people in the semantic technology world will find that certain aspects of this technology have already made a good impression on the publishing world and can provide a foundation for building more advanced applications together:

RDFa Typical publishers are interested in helpful new technology, but RDF and OWL discussions of blank nodes, striping, and open world assumptions scare then off. Because XML flows through the plumbing that makes up publishing production systems, RDFa provides a simple way to add RDF triples to the formats that they and their tools already know about, and they can build from there to take more advantage of semantic technology. (Remember, RDFa isn't just about embedding triples in HTML; you can use it with just about any XML DTD or schema.)

- SKOS If building an OWL ontology from scratch sounds like a complex, time-consuming venture into the unknown for a publisher, then the W3C standard SKOS ontology for taxonomy and thesaurus management can look like an old friend to them. Properties such as skos:broader, skos:narrower, and skos:historyNote represent basic taxonomy management concepts that they will recognize, and the prospect of customizing SKOS with a few new properties specific to their enterprise for their particular added value is much less intimidating than building a brand new ontology.
- SPARQL Query Results XML Format Semantic technology discussions of SPARQL focus mostly on the query language, with occasional references to the protocol. The Query Results XML Format, however, is a separate W3C Recommendation that defines an XML-based representation of a SPARQL query's results. I mentioned above that XML flows through the plumbing of a publisher's production system; the SPARQL Query Results XML Format is a simple, straightforward XML format that publishers can easily process with their existing tools, such as XSLT and XQuery, so that they have a much easier time incorporating the results of semantic technology tools into their own publishing systems.

RDFa provides a simple way to add RDF triples to the formats that they and their tools already know about, and they can build from there to take more advantage of semantic technology.

Looking ahead

The two worlds I've described here are becoming less and less separate. An impressive number of people with a strong information science background attended the 2010 Semantic Technologies conference, and the new W3C Library Linked Data Incubator Group announced shortly before that conference aims "to help increase global interoperability of library data on the Web, by bringing together people involved in Semantic Web activities—focusing on Linked Data—in the library community and beyond, building on existing initiatives, and identifying collaboration tracks in the future". I look forward to seeing the products of these collaborations and other efforts of publishing and semantic web technologists to get more out of their data and metadata by learning from each other's tools, standards, and best practices.

Bob DuCharme is a solutions architect at TopQuadrant, a provider of software for modeling, developing and deploying semantic web applications.

Future Content Authoring

By Markus Luczak-Rösch, Ralf Hesse, and Adrian Pashke



In times when famous publishers claim that "the people who simply just pick up everything and run with it–steal our stories…"[1] it is more clear than ever that the publishing sector is in a crisis due to the paradigm shift in the way people create and consume content. The Web offers a revolutionary space for everyone to aggregate and access information that fits individual needs. For sure, the success of publishing houses depends on the number of consumers that are accessing their free and paid content offers directly. But protecting this success by repelling external Web applications and search engines cannot be the answer.

The Semantic Web offers promising and standardized technologies to enrich content with meaningful metadata that can be processed by machines. As a lightweight intermediate step, the Linked Data initiative gained momentum not only in the scientific community. It has proven how publishing and interlinking of RDF on the Web can obtain new beneficial models of consumer-oriented content publication and consumption.

Even though these chances would inspire the stakeholders in the publishing sector to improve their content quality, the tool support for intelligent content authoring is limited. Since the users of such tools are non-experts w.r.t. semantic technologies, the most important requirements are:

- to provide a clear and intuitive user interface,
- to present only simple and well-described ontologies/vocabularies
- to let the user focus on the task of writing text

One Click Annotation

Following the above requirements we designed the so-called loomp One Click Annotator (OCA). It aims at the most intuitive workflow for authors to add semantics to their contents. The key idea is that every computer user is familiar with word processors and is able to select some text and to click on a button to format it italic. Thus, we transferred the concept of assigning style sheets to texts as known from modern word processors to the OCA for semantic annotation. In a word processor a user can choose between different sets of style sheets. After having chosen a set, he can select some text and assign a style sheet to it, (e.g., heading 1). In the OCA, a user can similarly choose between different vocabularies, (e.g., personal information and geography), and assign an annotation to some selected text, (e.g., foaf:firstName). The dashed box in Figure 1 shows the ribbon toolbar for selecting an annotation. To provide an intuitive and user-friendly interface the toolbar only displays human understandable labels but not namespace prefixed labels of properties–in general, we try to avoid showing URIs of the background semantic data on the human user interface.

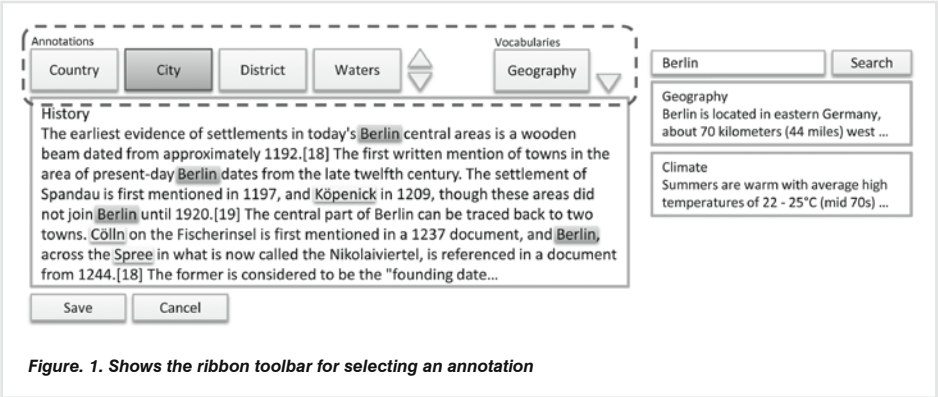


Figure. 1. Shows the ribbon toolbar for selecting an annotation

In our opinion, a non-expert user is hardly aware of the difference between a concept and the label of a concept. For example, if a user reads the phrase George Bush in some text, he probably wants to state that George Bush is a person not that it is the name of a person. We consider this behavior in the OCA by using the labels of properties in the toolbar which look like names of entities. Moreover, in many cases the system is able to add further semantic information to the text due to background knowledge contained in the vocabularies. For example, it can derive the type of the generated resource, because a property has a specific domain defined in its RDF schema (e.g., foaf:Person is the domain of foaf:surname). If users annotate the same resource in different texts, it is important to reference the same resource in the generated RDF statements. Otherwise, we get many resources that are not interlinked and the statements in the repository are not very useful and meaningful. After a user has selected some text and clicked on an annotation in the toolbar, the system presents a list with information of existing resources that are likely to be chosen by him (see Figure 2). The resources may originate from a local or an external repository (e.g., DBpedia). To enable a user to choose an appropriate resource, the list of resources has not only to contain the label of a resource but also some properties of it. If he does not find the resource in the list then he can manually search for a resource or create a new one.

Annotating text is a time-consuming task that, in most cases, users do not like to perform even if they have to add only a few metadata about the text.

Annotating text is a time-consuming task that, in most cases, users do not like to perform even if they have to add only a few metadata about the text. According to the principle "don't repeat yourself" a user can invoke annotation services which automatically analyze the text and propose annotations. Besides external annotation services such as OpenCalais we developed a client-side component that creates further annotations while the user is typing. This component keeps track of the annotations created by the user and adds the

1. Rupert Murdoch in an interview about search engines behavior in 2009– http://www.guardian.co.uk/media/2009/nov/09/murdoch-google

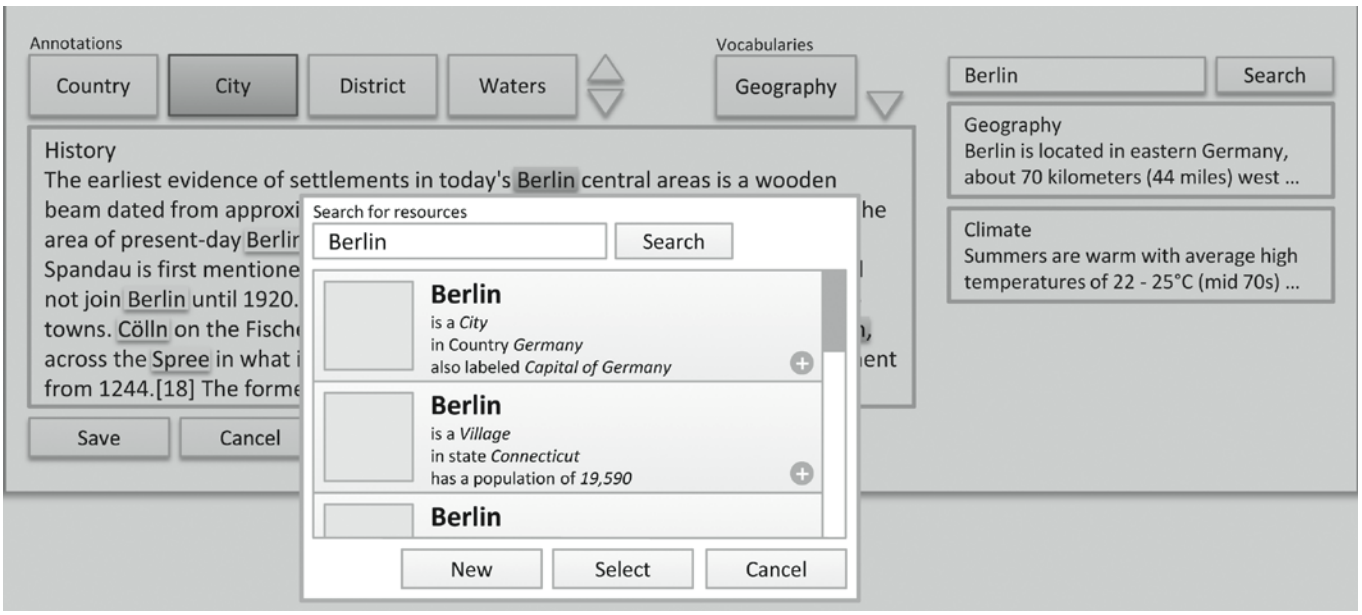


Figure. 2

same annotations to other parts of the text. It can be also used for pro-active suggestions and auto-completions. Independent of the used annotation service, a curation-style icon is placed behind any automatically generated annotation allowing to accept, to reject, to choose an alternative annotation, or to disable automatic annotation at all.

loomp Architecture

The OCA is the client part of the loomp architecture[2] for semantic content authoring (Figure 3). It is responsible for displaying the user's content as well as the annotation toolbar and for modifying the content to include semantics. The server is only contacted if the OCA needs information about resources or has to invoke external annotation services. In this section we describe details about processing of the user's content and the creation of semantic annotations by the OCA.

The server and the OCA exchange content in XHTML+RDFa format and data about resources in JSON format. Using the loomp API the OCA has access to the semantic content, information about resources, vocabularies for annotating content, and additional services.

The content API comprises functionality that is related to manag content. Its foundation is a simple and easy-to-understand domain model: we distinguish between content elements and documents. A content element can consist of text, multimedia objects[3], or embedded SPARQL queries. A document consists of a sequence of content elements. We believe that this domain model is advantageous, because it conforms to user's mindset, already used in modern content management systems, and allows an easy reuse of content. The data API allows for accessing the RDF data directly, (e.g., to retrieve all statements of a given resource). In an OCA, the functionality of this API is used to display all known information about a resource or to create links between resources. The vocabulary API provides access to sets of annotation. Finally, the service API provides access to (external) text processing methods, (e.g., to annotate content automatically using OpenCalais).

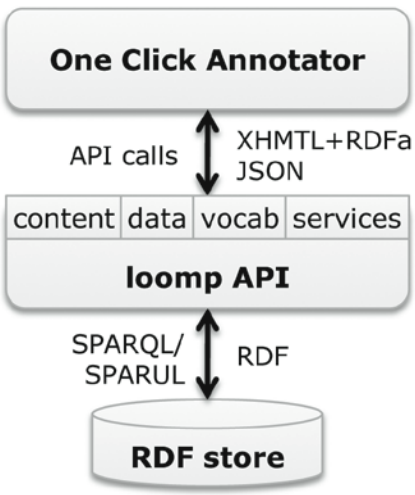


Figure. 3

Summary

A key prerequisite in the vision of a Semantic Web is to enable non-experts to create and consume semantic content on the Web easily. Currently, the barrier for them is still too high, because there are no easy-to-use Semantic Web authoring tools available. We introduced a simple-to-use loomp "One Click Annotator" for enriching text content semantically. The key design goals are a clear and intuitive user interface hiding the complexity of creating semantic data and simple access to vocabularies. To our best knowledge we are not aware of any tool that is focused on the target group non-experts as much as the OCA.

Funded by the German Federal Ministry of Education and Research (BMBF) and the BMBF Innovation Initiative for the New German Laender - Entrepreneurial Regions InnoProfile–the Corporate Semantic Web project at the Free University of Berlin aims to establish economically beneficial adoption of Semantic Web technologies in corporate environments. *Ralf Heese (hesse@inf.fu-berlin.de) and Markus Luczak-Roesch (markus.luczak-roesch@fu-berlin.de) work as a doctoral researcher in the Corporate Semantic Web project which is headed by Prof. Adrian Paschke (pashke@inf.fu-berlin.de).*

2. http://loomp.org
3. loomp supports only text and embedded SPARQL queries at the moment.

Facebook: David Recordon talks with Talis about the Social Graph

David Recordon, Senior Open Programs Manager at Facebook

David Recordon: I joined Facebook last fall. I'm really focused on open source and web standards, and going and building out Facebook's team around that. My background is working on technologies like OpenID and OAuth in the past few years. So I've always sort of had an interest in both the social side of the web as well as identity. I guess probably what's prompted this conversation is what we announced at F8, our annual developer conference, which is the Open Graph Protocol. That really goes and looks at how we allow any page on the Internet to have some more rich semantic markup to really represent itself well. It was a node inside of a graph.

Zach: So how does that thinking differ from what goes before, the idea of seeing the web as a graph as opposed to however else it's been seen?

> *The Open Graph Protocol is what really allows us, or allows anyone, to go and take a page and find information*

David: Well, I think Facebook's really always focused on this idea of the social graph and using the social graph to inform the majority of things that we do. In this case, it's really going and taking this notion and applying it to the web in a social manner as well. So you see it in a few different forms. One is the l*ike* buttons and other forms of social plugins that we have: allowing people to go and very easily provide input into a graph about content that they're interacting with. And then being able to go and use this from a recommendation perspective: going to a website like CNN and seeing on the front page the content that your friends are recommending you read, what's interesting right now. Then, the Open Graph Protocol is what really allows us, or allows anyone, to go and take a page and find information in an easy way such as a good canonical title, what type of node it should be in the graph, is it a movie, is it article, is it a business, a canonical URL, an image and some other basic information like that.

Zach: OK, so you're literally linking together the very bits and pieces of people's social lives together so that they can see a better and more full picture of it?

David: Yeah, I think that's a reasonable summary.

Zach: So, through the Open Graph Protocol, if I were a developer and I wanted to start using some of this, what's the idea for my site? If I've got a page which I'm interested in people being able to link to, what does it take for me to become part of that graph?

David: Well, we focused on a few different design decisions. Really, the one of them that was driving it the most was simplicity. So this was something where we knew that we wanted a technology that was no more complex than copy and paste. If we were going to go and get developers all over the web, some are used to running large websites and some which just have small websites, it needed to be something that was just a few different HTML tags that had a consistent schema throughout it. So then they could just easily put it into the head of the page.

So there are four required properties: the title, the type, the image and the URL. This is where we went and are making use of RDFa because it gives us this meta tag with the property attribute. It gives us consistency in terms of just having one type of HTML tag being able to have the same sort of markup in the header and the body when we get there.

So going back to the idea: what does it take for a developer to participate? Adding in this basic markup to their website then, going and, at least for us placing a social plugin onto their website which is that interaction that causes Facebook to go and fetch your web page, find the Open Graph Protocol markup and start doing interesting things with it.

But obviously, since it's a set of meta tags that's based on RDFa, it's something that anyone can go and make use of when they're looking at the content of a website.

Zach: OK. So the first step is to say: "This is who I am and this is the kind of things I'm doing," using fairly straightforward RDFa markup to identify the things you're talking about. Give it a title and a little bit of meta data about your subject. The second part, from Facebook's perspective, is to identify to Facebook who you are and what you do. Is that the idea of the social plugin?

David: So you don't need to give Facebook any additional information. The most common case is going and placing a *like* button onto a website. Literally, you go and include an iframe and the first time somebody clicks like with your URL, we'll go and fetch your web page, parse it for the Open Graph Protocol and just use that information.

Zach: OK, and what does that create within Facebook? So if I put the like button on zachbeauvais.com or on talis.com, and say, somebody likes one of the articles I wrote—in the long distance hope somebody might do—they click "like," what goes on behind the scenes there?

> *The product experience that we were really building was allowing any web page to have the same sorts of social properties and social interactions and distribution as a page on Facebook.*

David: The product experience that we were really building was allowing any web page to have the same sorts of social properties and social interactions and distribution as a page on Facebook. So the past few years, we've had a product, Facebook Pages, which lots of businesses and public figures and all sorts of different people have been using. It allows people to go and connect with brands, have those owners of pages create updates, which go into their fans' news feeds and all sorts of different things like that. So that's what we wanted to allow for any web page, not just pages on Facebook. As the admin of a page, this is actually an area where we've gone and extended the Open Graph Protocol a little bit. We've defined two tags, which are specific to Facebook. One which allows you to list the user IDs for admins of your page or just list an application ID, which represents the admins for

your page. So what we do is once someone goes and clicks a like button or we find out about one of these URLs, we'll go and fetch it, we'll go and extract Open Graph Protocol meta data, and we'll create a Facebook page. When you go to it from an admin perspective, you're able to go and update your fans and do things like that. But when anyone else goes and sees this link on Facebook, they'll actually be taken directly to your website.

Zach: OK, so if I create this social plugin for my site and somebody sees it within Facebook itself and they click on to get more information about it, instead of seeing a Facebook page with the information brought through, they're actually going to my site?

David: Correct. And we actually do a little bit more for some of the content types: things like movies and music. Let's say that you like Avatar on IMDB or on Rotten Tomatoes. Avatar's a movie and there's a section on your profile for movies that you like. We'll go and actually add Avatar to that section of your profile. Once again, when somebody goes and clicks on Avatar from your profile, they'll go off to IMDB or they'll go off to Rotten Tomatoes.

Zach: So you're being a bit more clever with the actual semantics of the content type there and trying to get some more sort of social use out of just a little piece of information?

David: Yeah.

Zach: Can you tell us a little bit more about the Open Graph Protocol itself, and how that differs from what Facebook's doing with it?

What we're doing is really built on this foundation of the Open Graph Protocol. It requires four pieces of basic metadata. Title, type, URL, and image.

David: What we're doing is really built on this foundation of the Open Graph Protocol. It requires four pieces of basic metadata. Title, type, URL, and image. We then ask for you to list what admins on Facebook should own your page, so that we can go and create that. From a metadata perspective, that's all that we're doing. We're really going and making use of the Open Graph Protocol. It's actually been pretty interesting. Since F8, there's been a pretty reasonable community starting to form around it. There's almost a dozen different implementations of it in a variety of languages like Perl, PHP, Python and Java. A few different web services that you can go and put in an URL and it returns the Open Graph Protocol markup. Things like WordPress plugins to make it easy to add to your blog. I think we're really focused on making use of the Open Graph Protocol, and having it solve a variety of problems there, and then being able to go in and to use that information to build a really interesting product experience around it.

Zach: But the Open Graph protocol isn't part of Facebook itself, right?

David: No. That was one of our other design goals when we set out to work on creating it, was creating what would be useful for others beyond Facebook. So, there's nothing in the Open Graph Protocol (opengraphprotocol.org) that's specific to Facebook. It is built around interacting with web pages in the social manner, but it's information that isn't specific to Facebook, and really could be used by anyone.

There's nothing in the Open Graph Protocol (OpenGraphProtocol.org) that's specific to Facebook. It is built around interacting with web pages in the social manner, but it's information that isn't specific to Facebook, and really could be used by anyone.

Zach: Any blog owner or content owner can markup their content using it, but then people can also develop services built on making use of that information outside of Facebook?

David: Absolutely, and certainly that's something where I think it makes a lot of sense for things like search engines to go and make use of some this information.

Zach: When you were developing the Open Graph Protocol around this social graph, did you look at other ontologies that exist like FOAF or SKOS? Why is it that you decided to develop a whole new one?

David: We looked a bit at some of the other ontologies, as well as other technologies. I mean, for what we recommend for specifying location information. We actually based all of the names off of the schema used by the H-Card microformat. So, I tried to reuse where possible, but realised that that wasn't something we could do all the time. And so, it really comes back to the idea of creating something simple enough for any web developer. We actually had a version of this that we were playing around with internally where we were using Dublin Core for the title. We had to go in to define type as part of the Open Graph schema we were using, FOAF to represent the image, and we were using Link Rel Canonical for the canonical URL. This meant that there were three name spaces on the HTML tag. We actually don't really expect most people to include any name space, not for any malicious reason, but just because we expect it to generally be forgotten. And then, when you're going and looking in this markup, you have a few different prefixes. You have two different tag types, and it just didn't seem very clean to us. It was hard for a developer to go and pick out which of these are the specific tags that are part of the Open Graph Protocol. And so, we really went and optimised for: How do we keep this simple? How do we keep this consistent? How do we go and reuse names from other schemas, even if we're going and using a different namespace there? And so, this was something that I think was really good that came out of the WWW Conference a month or so ago, was that a bunch of people in the semantic web community actually helped develop an RDF schema file for it that goes and just relates a lot of the properties that are defined in the Open Graph Protocol to other existing schemas. So, a RDF aware parser will be able to go and have some equivalency between these different things.

Zach: So, do you see Facebook making use of these external ontologies itself?

David: I think it depends on the different parts of the technology. I don't have an absolute answer right now, but there's sort of a general feeling that the basic information needs to be extremely simple for developers to interact with and to place. But when we start to get into things which might be a little bit more complex, it's probably reasonable that that information ventures outside of the head of the page but is actually included inside of the HTML in the body, much more like how microformats work, which is one of the things that we like about RDFa, is that you can go and have a consistent syntax whether you're the metadata in the head of the page or in the body of the page.

And probably, it is reasonable to go and break our one schema rule when we start to get into more complex information, as well.

Zach: OK, excellent. One of the things that comes up quite a lot when it comes to talking about Facebook, and I found it quite interesting looking at your title and the stuff that you've done—because you've worked with OpenID and you deal a lot with open source—is that Facebook is quite often seen as a walled garden on the web. There are a lot of posts around about how do we interact on the wider web when a lot of our information's within Facebook. I was wondering if you had anything to say in that direction?

Facebook really was the first site of its size to allow people to sign in using OpenID.

David: I think you've seen lots of different changes from Facebook over the past few years. Facebook really was the first site of its size to allow people to sign in using OpenID. We did a lot of work the past six to nine months on developing OAuth too within the ITF, and then we're really the first site to go and ship it at F8 for the graph API. I think even the graph API allows access to a large amount of information, assuming you have user consent to get at it. And so, you can go and get people's photos, people's content, all of those sorts of things that you would want to access for the information that you've created. We also did some changes to our data policies at F8. Previously you, as a developer, could only store data for 24 hours, at which point you had to go and refresh it. This was really trying to create a good experience by making sure that when somebody changed something on Facebook, that information was propagated around the web and kept up to date elsewhere. At F8, we removed that hard requirement of only storing data for 24 hours, but rather said you have to keep the data up to date. We launched a real-time API so that you can be proactively notified when information changes for your users. If a user asks you to delete their data, you have to delete it and you have to have a privacy policy and some other things like that. So I actually think that with a lot of what we've been doing the past year, you're seeing the ability to access more and more different types of information, always assuming that users remain in control over that, that they give you, as an application developer, the ability to do that.

We launched a real-time API so that you can be proactively notified when information changes for your users. If a user asks you to delete their data, you have to delete it and you have to have a privacy policy and some other things like that.

Zach: It sounds like a really difficult line to walk. On one side, you've got people criticizing Facebook because it's a walled garden: you've got data silos and content stuck inside. Then on the other, you've got the issue with privacy and people not necessarily wanting their details published in lots of ways. So, I have a lot of sympathy watching this from the outside. It is a hard balancing act. I watched the F8, and I watched the Twitter streams erupt in different ways as different things were announcing. Then you see the blog posts start—you get posts saying: "Hurray, it's all opening up," and: "Oh no, what about privacy?"

David: Yeah, often form the same developer it seems.

Zach: [laughs]

David: But I think this also goes back to one of the other things that we launched at F8, which is really focused on granular data permissions. It's actually been rolling out the past few weeks and will be rolling out even more, I think, over this next month, it's ability to go and say: "These are the specific pieces of information that I'd like access to." Then the dialogue that we present to a user is actually extremely clear about what information is this application asking for, including things like offline access and making that really clear so that people have a better ability to understand what developers are wanting from a data perspective and make a more informed decision about it.

Zach: Presumably, if I'm writing an application which wants to take your photo and then match it with somebody else, then I've got to ask specifically. So my application is going to take your photo, it's going to store it on a third-party server, it's going to do these things, do you grant me permission, that kind of thing?

David: Yeah. It'll ask for access to your photos. It doesn't go quite as far as saying that they're going to store it on a third-party server and exactly what they're going to do with it.

Zach: Right, obviously, you don't want to confuse people.

I think one of my goals is definitely seeing someone other than Facebook really make use of this data in an impactful way.

David: Yeah, but having people be very aware that they're going to get access to my photo albums and having that be something that's communicated both visually and in text, I think, is a really good thing overall.

Zach: OK. Yeah, it's absolutely fascinating to hear from within Facebook who's talking about it from a very open perspective. What I'm wondering is whether you've got anything that you see coming out of this in the next few months?

David: The most exciting thing in the next few months, I think going back to the original topic of the Open Graph Protocol, I think we're just going to continue seeing more and more adoption of all sorts of different sites around the web that are producing content in one form or another going and adding this markup. I think one of my goals is definitely seeing someone other than Facebook really make use of this data in an impactful way. Because I think it's something that if we can all be telling what developers same sort of message of this is how you should go in and include more meta data on your page, that will ultimately make the web better overall.

Zach: I was just having to look through some of the notes that I made in preparation for this podcast and I actually wrote a really quick overview of what I saw from F8 in Nodalities Magazine. My strongest impression after reading lots of different posts and lots of different perspectives on it is that what we're talking about is a huge influx of semantically marked up data coming out. Also with only the four quick meta tags, they're a very low barrier to entry. We could start seeing lots of these seeds for data being spread out there. I guess, from my perspective, the most interesting thing will be when lots of developers start to look at this data and start to say: "What can I build with

this? How can I tie in with the social information that's being published out there?" And possibly linking that with other ontologies, other services, all that kind of thing.

So it's fantastic to hear that that's also become the idea behind it with Facebook as well.

David: Yeah. I think you're exactly right. By focusing on the really simple case, making it easy to adopt, there's going to be vastly more semantic information on the web. I think that goes back to what happened a few weeks ago in terms of going and having a schema that relates it to other existing ontologies will also help people get an even deeper understanding out of this information. Really that bar for implementing a technology has a tremendous impact on its ultimate adoption. So hopefully by keeping this extremely simple, we'll see broad and widespread adoption.

Also because of the incentives, in terms of going and having traffic driven back to your site and being able to have social experiences around your site as well, will go and cause a lot of people to include this basic markup in a way that's not specific to Facebook that developers could go and build all sorts of things with.

Zach: OK, brilliant. Well, David, thank you very much for your time.

David: Thank you for having me.

David Recordon is Senior Open Programs Manager at Facebook

Nodalities Magazine is a Talis Publication

ISSN 1757-2592

Talis Group Limited
Knights Court, Solihull Parkway
Birmingham Business Park B37 7YB
United Kingdom
Telephone: +44 (0)870 400 5000
www.talis.com/platform
nodalities-magazine@talis.com

The 9th International Semantic Web Conference

Shanghai International Convention Center,
Shanghai, China

http://iswc2010.semanticweb.org

ISWC 2010

Semantics for a Better Web!

General Chair
- Ian Horrocks

Program Chairs
- Peter F. Patel-Schneider
- Yue Pan

Local Chair
- Yong Yu

Workshop & Tutorial Chairs
- Philippe Cudré-Mauroux
- Bijan Parsia

Poster & Demo Chairs
- Huajun Chen
- Axel Polleres

Industry & Semantic Web in Use Chairs
- Pascal Hitzler
- Peter Mika

Doctoral Consortium Chair
- Jeff Pan

Semantic Web Challenge Chairs
- Chris Bizer
- Diana Maynard

Publicity Chair
- Sebastian Rudolph

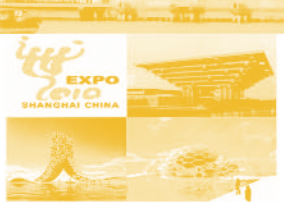
Metadata Chair
- Jie Bao

Proceedings Chair
- Birte Glimm

Sponsor Chairs
- Kendall Clark
- Anand Ranganathan

Local Organization
- Dingyi Han
- Gui-Rong Xue
- Haofen Wang
- Lei Zhang

Stunning Venue



Great Time

Fascinating City

Excellent Food

SWSA
Semantic Web Science Association

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

IBM Research