

## 1 ANATOMY OF A SEARCHMONKEY

Peter Mika's run-down of Yahoo's new Semantic Web search platform.

## 8 TWINE

Nova Spivack gives us an update from the beta of social interaction application.

## 10 GENERATION ZERO

Ian Davis asks where all the Semantic Web applications are.

## 13 A CALL TO ARMS

John Sheridan and Jeni Tennison on the Public Sector's use of Semantic Web technologies.

## 16 DISCOVERY AND USAGE ON THE WEB OF DATA

Michael Hausenblas and VOID.

## 22 FINDING ANSWERS ON THE SEMANTIC WEB

Mathieu d'Aquin and Lopez from KMI introduce Watson and PowerAqua.

# Anatomy of a SearchMonkey

written by Peter Mika



*SearchMonkey very simple app / that anyone can use / even monkey like you. - Paul Tarjan, "The SearchMonkey Song"*

Presenting compelling search results depends critically on understanding what is there to be presented in the first place. Given that the current generation of search engines have a very limited understanding of the query entered by the user, the content returned as a result and the relationship of the two, the opportunities for customizing search results have been limited.

In fact, this largely explains why the generic-purpose Web search engines we use every day present a similar interface: a list of ten web page results with title, abstract and syntactic metadata (e.g. size of the document) with a similarly-formatted set of advertisements placed on the top, right, or bottom.

This particular arrangement is not the ideal presentation of results for *any particular query*: there are hardly any queries for which the answer is an ordered list of titles.<sup>2</sup> It is, however, one

## 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](http://www.talis.com/nodalities) or email [nodalities-magazine@talis.com](mailto:nodalities-magazine@talis.com).

---

## MEET US AT

Talis staff will be presenting and attending several events over the next few months, including;

### Dublin Core Conference

Berlin, Germany 22-26 September 2008

### European Semantic Technology Conference (ESTC)

Vienna, 24-26 September

### Linked Data Planet West Coast

Santa Clara, 16-17 October

### International Semantic Web Conference (ISWC)

Karlsruhe, Germany 26-30 October 2008

### Defrag

Denver, Colorado 3-4 November 2008

For further information visit: [www.talis.com/platform/events](http://www.talis.com/platform/events)



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](http://www.talis.com/platform)



The views expressed in this magazine are those of the contributors for which Talis Group Limited accepts no responsibility. Readers should take appropriate professional advice before acting on any issue raised.

©2008 Talis Group Limited. All rights reserved. Talis, the Talis logo, Nodalities and the Nodalities logo where shown are either registered trademarks or trademarks of Talis Group Limited or its licensors in the United Kingdom and/or other countries. Other companies and products mentioned may be the trademarks of their respective owners.

## EDITOR'S NOTES



Hello, and welcome to Nodalities' fourth edition. You can receive all upcoming editions of Nodalities in print—for free—by signing up on our website, where you can also read past editions or download the pdf's: [www.talis.com/nodalities](http://www.talis.com/nodalities).

If the theme from the last edition of Nodalities Magazine was: "the Semantic Web is Here"; then this one must be: "...and this is what to do with it". In the Semantic Web space, there are many projects, companies, and people working to make the web work. From Yahoo, Peter Mika explores the anatomy of a Searchmonkey, showing us the ways and whys in which Yahoo's new search platform works. Nova Spivack, CEO and founder of Twine gives us a progress-report on the soon-to-be launched social interest tool Twine. We also have a brilliant introduction to the new Watson search engine from KMI written by Mathieu D'Aquin and Vanessa Lopez.

The Semantic Web, Linked Data, and other concepts and technologies have much to offer the Public Sector, as well. We have a call to arms by John Sheridan and Jeni Tennison, who've worked on various governmental projects including the London Gazette. From academia, Dr. Nicolas Morton explains the need for next-generation data tools in the study of medieval history, and Talis' own Tom Heath replies with ways in which the answers could come from the linked data space.

Alongside these, we also have a couple papers drawing attention to areas which need further work. Michael Hausenblas expands on his idea that the web is fundamentally broken, and explains where ontologies fill some of these gaps. Ian Davis, CTO of Talis, also expresses concern that there aren't enough applications which demonstrate the Semantic Web's capabilities, explaining that we are at "Generation Zero" when it comes to publicly-usable technology.

As editor for Nodalities, I am very keen to explore the issues raised by these articles as you read them. If you have any questions about them, replies to topics, or feel you'd like to join in the conversation you can visit the blog ([blogs.talis.com/nodalities](http://blogs.talis.com/nodalities)) or email me: [zach.beauvais@talis.com](mailto:zach.beauvais@talis.com).

that in principle suits the *average query* among the billions of queries search engines receive every day.<sup>2</sup> However, a problem with this argument is that there is no actual 'average query': research conducted by researchers at Yahoo shows that in a year-long query studied 88% of the unique queries are singleton queries, and 44% are singleton queries out of the whole volume, which means that the vast majority of Web queries are only seen once, even when looking at a full year of query production. Many of these queries are certainly spelling variants of the same concepts but still the number is indicative of the length of the long tail in information needs.

In the following, we introduce SearchMonkey, which opens up search by allowing user-created applications to modify the search experience based on structured data associated with web pages. Before delving into the technical details, we will first describe the benefits of an open search ecosystem to publishers, users and web developers. Next, we give a mile-high overview of the technical architecture and inspect the particular role that Semantic Web technology has played in realizing the system.

## Opening up search

The departure point for SearchMonkey is the observation that the vast majority of Web pages on the Web are generated from some form of a database and that more recently, Web site owners have started to publish some of the data as information embedded inside HTML pages (microformats, eRDF, RDFa) or using APIs. However, even in cases where the data is not directly available, it is possible to develop extraction methods (such as XSLT stylesheets) based on a single page and run it on all pages that conform to the same structure (typically, all pages that show the same kind of data formatted using the same template). This works particularly well for static 'resource type' pages showing the complete

**Topics for Getting Pregnant - BabyCenter**

Tools  
Before you Try  
Active Trying  
Having Trouble

BabyCenter's getting pregnant tools and information can boost your chances of conception by helping you chart your cycle, read your cervical mucus, and pinpoint ovulation.

www.babycenter.com/getting-pregnant - 76k - [Cached](#)

---

**Dr. Seuss' Horton Hears a Who (2008) Movie - Acme Movies**

Movie Details  
Showtimes & Tickets  
Trailers & Clips  
Critics Reviews

Critics Review: ★★★★★ (173)  
MPAA Rating: G  
Running Time: 1 hr. 28 min.  
Release Date: March 14th, 2008 (wide)

acmemovies.com/hortonhearsawho - [Cached](#)

Reviews ★★★★★ (77) | Cast & Crew | Netflix

**Movie Reviews - Flixster**

★ Top Critic  
Moviebuff71  
San Francisco, CA

Seuss lovers may now heave a sigh of relief. Unlike those behind the recent live-action grotesqueries *The Grinch* and *Cat in the Hat*, the makers of the *Horton Hears a Who!* do well by the great, good medic of metrical writing.

[See 76 more reviews](#)

Fig. 1

description of a single object: a single image, video, product, news item etc.<sup>3</sup>

There are possibly many ways to use metadata to improve search result presentation, but one of the obvious areas to focus on is the description of individual pages, i.e. the title and abstract generated. As automatic summarization is a difficult AI problem, automatically generated abstracts in particular often provide a very poor overview of the page. In resource type pages typically automatic summarization tends to pick up the wrong parts of the page (boilerplate text around the object described) and the description is often partial (focusing on the immediate area where the query words are found). Further, tables (containing the attributes of the object) are flattened into a sentence-like format while images are simply removed: again, the recognition and selection of appropriate images is difficult to automate in a reliable fashion. Links are treated in a similar fashion since showing links in the abstract text could disorient the user. (It might be unclear to the user which link is the link to the result page and which links take to other pages.)

Given the above, the SearchMonkey proposition is easy to explain: SearchMonkey reuses structured data to improve search result display with benefits to both search users, developers and publishers of web content. The first type of applications are focusing on remaking the abstracts on the search result page: Figure 1 shows the kind of presentations that structured data enables in this space. Based on data, the image representing the object can be easily singled out. One can also easily select the most important attributes of the object to be shown in a table format. Similarly for links: the data tells which links represent important actions the user can take (e.g. play the video, buy the product) and these links can be arranged in a way that their function is clear. In essence, knowledge of the data and its semantics enables to present the page in a much more informative, attractive, and concise way.

The benefits for *publishers* are immediately clear: when presenting their page this way publishers can expect more clicks and higher quality traffic flowing to their site. In fact, several large publishers have moved to

<sup>1</sup> A query like "first 10 albums of madonna in order of release" would come close but is rare.

<sup>2</sup> 'In principle' because there is undeniably also another effect in play: for better or worse this is the presentation users have gotten used to.

<sup>3</sup> Overview-type pages are more difficult to extract data from as they tend to be dynamic and showing only a limited view of each object.

implement semantic metadata markup (microformats, eRDF, RDFa) specifically for providing data to SearchMonkey.

In the following, we turn our attention to how these benefits are realized and the particular role Semantic Web technology plays in the design.

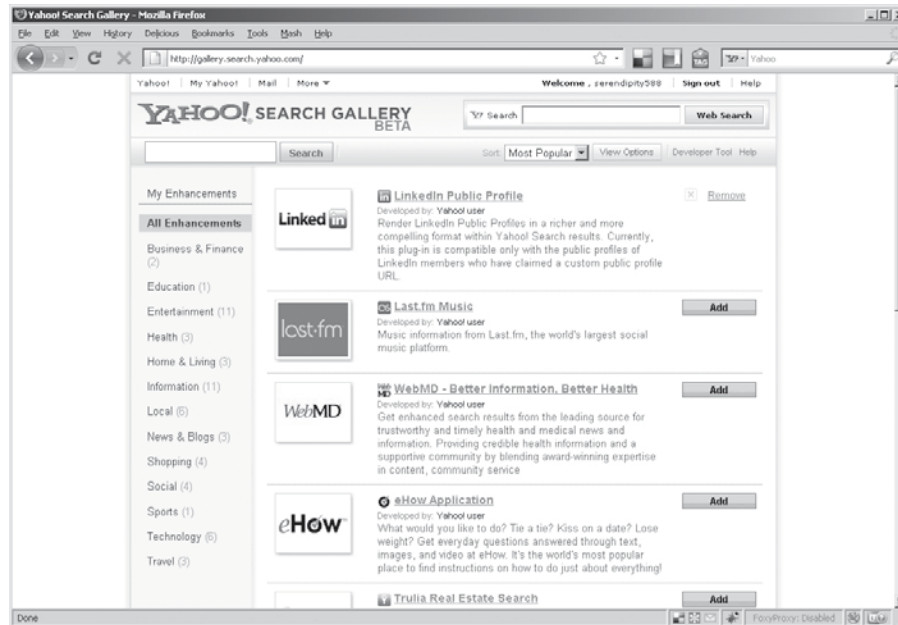


Fig.2

On the other hand, *users* also stand to benefit from a better experience. The only concerns on the users' part is the possibility of opting out and having a system free of spam. Both concerns are addressed by the Yahoo Application Gallery. As shown in Figure 2, the Gallery allows users to selectively opt in to particular SearchMonkey applications. Users also have a small number of high-quality applications that are turned on by default. The gallery is also an effective spam detection mechanism: applications that are popular are unlikely to contain spam. Also important to note that the presence of metadata does not affect the ranking of pages. Pages that are trusted by the search engine based on other metrics can be also expected to contain trustworthy metadata.

For *developers*, the excitement of the system is in the possibility to take part in the transformation of search and develop applications that are possibly displayed to millions of users every day. Needless to say, many publishers become developers themselves, creating applications right after they have added metadata to their own pages.

## Architecture

The high level architecture of the system (shown in Figure 3) can be almost entirely reconstructed from the above description. The user's applications trigger on URLs in the search result page, transforming the search results. The inputs of the system are as follows:

Metadata embedded inside HTML pages (microformats, eRDF, RDFa) and collected by Yahoo Slurp, the Yahoo crawler during the regular crawling process.

Custom data services extract metadata from HTML pages using XSLT or they wrap APIs implemented as Web Services.

Metadata can be submitted by publishers. Feeds are polled at regular intervals.

Developers create custom data services and presentation applications using an online tool (see Figure 4). This tool is a central piece of the SearchMonkey experience: it gives developers access to all their services and applications. When

defining new custom data services, first some basic information is provided such as name and description of the service and whether it will execute an XSLT or call a Web Service. In the next step, the developer defines the trigger pattern and some example URLs to test the service with. Next, the developer constructs the stylesheet to extract data or specifies the Web Service endpoint to call. (Web Services receive the URL of the page as an input.) When developing XSLTs, the results of the extraction are shown immediately in a preview to help developers debug their stylesheets. After that the developer can share the service for others to build applications on and can also start building his own presentation application right away. Note that custom data services are not required if the application only uses one of the other two data sources (embedded metadata or feeds).

Creating a presentation application follows a similar wizard-like dialogue. The developer provides the application's name and some other basic information, then selects the trigger pattern and test URLs. Next, SearchMonkey shows the schema of the data available for those URLs taking into account all sources (embedded metadata, XSLT, Web Services, feeds). The developer can select the required elements which again helps to narrow down when the application should be triggered: if a required piece of the data is not available for a particular page, the application will not be executed. Then as the main step of the process, the developer builds a PHP function that returns the values to be plugged into the presentation template. As with stylesheets, the results of the application are shown in a preview window that is updated whenever the developer saves the application. In practice, most PHP applications simply select some data to display or contain only simple manipulations of the data. The last step is again the sharing of the application.

As the description shows, the representation of data is a crucial aspect of SearchMonkey, since the



output is simply a result of executing a set of transformations on Web data. Some of these transformations are performed using XSLT and some are complex enough to require a full-blown programming language such as PHP. What connects these pieces is a canonical representation of structured data.

To understand better the choices made in dealing with structured data it is useful to summarize the main requirements that any solution must fulfill. These are as follows:

**An application platform based on structured data.** As described above, the starting point of SearchMonkey was the observation that the (vast) majority of Web pages on the Web are generated from some sort of a database, in other words driven by structured data. Publishers are increasingly realizing the benefits of opening up both data and services in order to allow others to create mash-ups, widgets or other small, non-commercial applications using their content. (And in turn, developers are demanding more and more the possibility to have access to data.) The preferred method of opening up data is either to provide a custom API or embed semantic markup in HTML pages using microformats. Thus SearchMonkey requires a data representation (syntax) that could generally capture structured data and a schema language that provides a minimally required set of primitives such as classes, attributes and a system of data types. The syntax and semantics should allow to capture the full content in typical microformat data and the languages used should be open to extensions as much as possible.

#### Web-wide application interoperability.

The queries received by a search engine and the content returned as a result cover practically all domains of human interest. While it would have been easier to develop, a solution that would limit the domains of application would not meet the need of a global search product as it would not be able to capture the long tail of query and content production. The question of interoperability is complicated

by the fact that the application may be developed by someone other than the publisher of the data. On the one hand, this means that data needs to be prepared for serendipitous reuse, i.e. a

**Direct display** If one would purely focus on how to get to the end result presented in Section x, it would appear that capturing the structure of data would

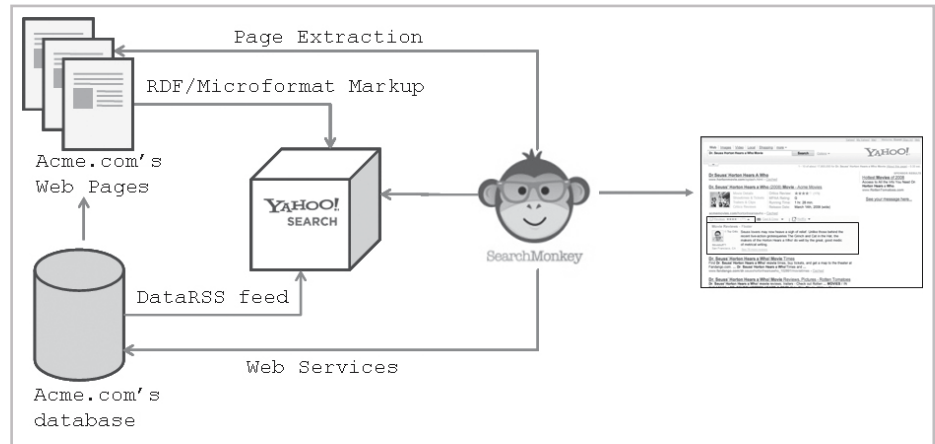


Fig.3

developer should be able to understand the meaning of the data (semantics) by consulting its description (minimally, a human readable documentation of the schema). On the other hand, the framework should support building applications that can deal with data they can only partially understand (for example, because it mixes data from different schemas). Changes in the underlying representation of the data also need to be tolerated as much as possible.

**Ease of use.** It has been often noted that the hallmark of a successful Semantic Web application is that no user can tell that it was built using semantic technologies. That novel technology should take a backstage role was also a major requirement for SearchMonkey: the development environment is targeted at the large numbers of Web developers who are familiar with PHP and XML technologies (at least to the extent that they can understand an example application and start extending or modifying it to fit their needs). However, developers could not have been expected to know about RDF or RDFa, technologies that still ended up playing a role in the design of the system.

#### The role of semantic technology

Before describing the solution that was taken in realizing SearchMonkey, let us consider the alternatives.

not be required at all. Since the presentation is fixed, publishers might as well markup directly in their webpage the elements that should go in each element of the template:

This would have obviated most of the engineering effort behind SearchMonkey, in particular there would have been no developer tool as there would have been no applications to be developed. This alternative however is only interesting to consider to see just how limiting it would have been. Namely, it would not have

```

<a rel="link1" href="http://
www.example.org/first/link/
to/show">First link to
show</a>
<span class="key1">Name</
span>
<span class="value1">Peter</
span>
  
```

been possible to develop applications that display the same information consistently across sites or adapt to the type of information displayed in any way. Needless to say it would have also ruled out the future option of providing search services over structured data.

### XML-based technologies}

The next possibility to consider is the use of XML as a representation syntax and XML Schema technologies to define the structure of the data, i.e. the contract between data producers and data consumers. This solution involves creating a number of XML schemas covering the domains of interest. This is also the direction taken by Google in the Google Base system. Information in the metadata feeds submitted to Google Base is expected to contain the item types and attributes defined by the company. Existing XML Schemas such as NewsML could have been reused. Microformat data also lends

### Semantic technologies

Semantic technologies promise a more flexible representation than XML-based technologies. Data doesn't need to conform to a tree structure, but can follow an arbitrary graph shape. As the unit of information is triple, and not an entire document, applications can safely ignore parts of the data at a very fine-grained, triple by triple level. Merging RDF data is equally easy: data is simply merged by taking the union of the set of triples. As RDF schemas are described in RDF, this also applies to merging schema information.<sup>1</sup> Semantics (vocabularies) are also completely decoupled from syntax. On the one hand, this means

are also a good match for some of the input data of the system: eRDF and RDFa map directly to RDF triples. Microformats can also be re-represented in RDF by using some of the pre-existing RDF/OWL vocabularies for popular microformats

Given the requirements, the benefits and drawbacks of these options and the trends of developments in the Web space, the choice was made to adopt RDF-based technologies. However, it was also immediately clear that RDF/XML is not an appealing form of representing RDF data. In particular, it does not allow to capture metadata about sets of triples such as the time and provenance of data, both of which play an important role in SearchMonkey. Other RDF serialization formats have been excluded on the same basis or because they are not XML-based and only XML-based formats can be input to XSL transformations.

These considerations led to the development of DataRSS, an extension of Atom for carrying structure data as part of feeds. A standard based on Atom immediately opens up the option of submitting metadata as a feed. Atom is an XML-based format which can be both input and output of XML transformation. The extension provides the data itself as well as metadata such as which application generated the data and when was it last updated. The metadata is described using only three elements: item, meta, and type. Items represent resources, metas represent literal-valued properties of resources and types provide the type(s) of an item. These elements use a subset of the attributes of RDFa that is sufficient to describe arbitrary RDF graphs (resource, rel, property, typeof). Valid DataRSS documents are thus only valid as Atom and XML, but also conform to the RDFa syntax of RDF, which means that the triples can be extracted from the payload of the feed using any RDFa parser. [Fig. 5] provides an example of a DataRSS

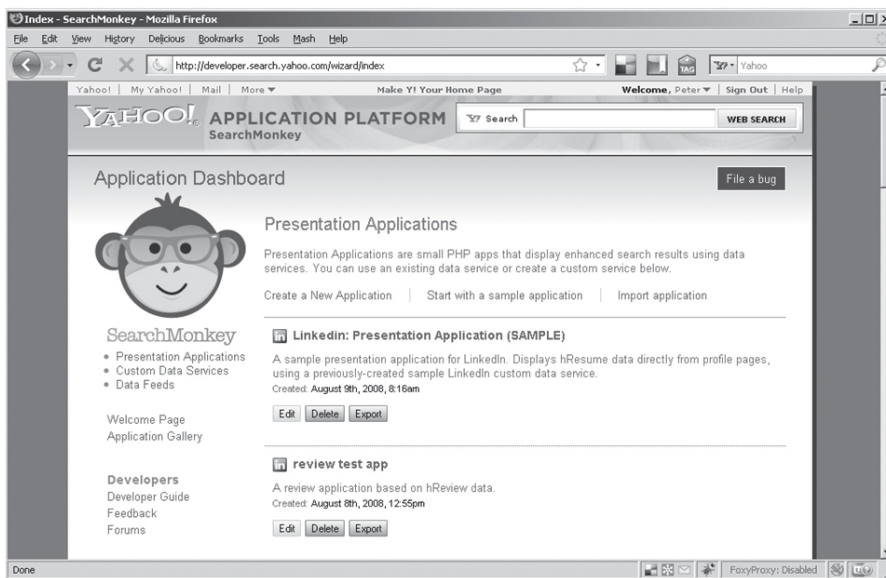


Fig.4

itself naturally to modeling using XML Schema since there are a small number of microformats to be captured with relatively stable schemas.

This solution offers a number of advantages. As the schema is fixed and known in advance, a fixed set of input forms can be generated from the schema and incoming information can be easily validated. As the information types are well-understood by the system, it is possible to provide default presentations of the information.

that RDF doesn't prescribe a particular syntax and in fact triples can be serialized in multiple formats, including the XML based RDF/XML format. (An XML-based format was required as only XML can serve as input of XML transformations.) On the other hand, it also means that the resources described may be instances of multiple classes from possibly different vocabularies simply by virtue of using the properties in combination to describe the item. The definition of the schema can be simply retrieved by entering URLs into a web browser. RDF-based representations

4 Obviously true merging requires mapping equivalent classes and instances, but that is not a concern in the current system.

feed describing personal information using FOAF (the Friend-of-a-Friend vocabulary).

A new format also brings along the question of query language. Since DataRSS is both XML and RDF, the two immediately available options were XPath and SPARQL. However, it was also immediately clear that both languages are too complex for the task at hand. Namely, presentation applications merely need to filter data by selecting a simple path through the RDF graph. Again, the choice was made to define and implement a simple new query language. (The choice is not exclusive: applications can execute XPath expressions, and the option of introducing SPARQL is

also open.) As the use case is similar, this expression language is similar to Fresnel path expressions [?] except that expressions always begin with a type or property, and the remaining elements can only be properties. [Fig.5] also gives some examples of this query language.

## Conclusions

In this article we have introduced the concept of SearchMonkey, the requirements that it presented and the architecture that has been implemented. We have seen the critical role that semantic technologies have played in realizing this web-wide platform for applications that build on structured data. As the lyrics of the SearchMonkey song suggest, the current applications

populating this platform are relatively modest transformations of structured data into a presentation of the summary of web pages. However, the platform is open to extensions that use structured data to enrich other parts of the search interface or to be deployed in different settings such as a mobile environment. Lastly, in building on semantic technologies SearchMonkey has not only accomplished its immediate goals but is well prepared for a future with an increasingly more semantic web where the semantics of data will drive not only presentation but the very process of matching user intent with the Web's vast sources of structured knowledge.

```
<?profile http://search.yahoo.com/searchmonkey-profile ?>
<feed xmlns="http://www.w3.org/2005/Atom"
xmlns:foaf="http://xmlns.com/foaf/0.1/"
xmlns:y="http://search.yahoo.com/datarss/">

  <id>http://www.linkedin.com/datarss/</id>
  <author><name>Peter Mika (pmika@yahoo-inc.com)</name></author>
  <title>Example data feed for social</title>
  <updated>2007-11-14T04:05:06+07:00</updated>
  <entry>
    <title>Peter Mika</title>
    <id>http://www.linkedin.com/ppl/webprofile?id=5054019</id>
    <updated>2007-11-14T04:05:06+07:00</updated>
    <content type="application/xml">
      <y:adjunct version="1.0" name="com.yahoo.social">
        <y:item rel="dc:subject">
          <y:type typeof="foaf:Person vcard:VCard">
            <y:meta property="vcard:fn foaf:name">Peter Mika</y:meta>
            <y:meta property="foaf:age">29</y:meta>
            <y:meta property="vcard:bday">1978-10-05</y:meta>
            <y:item rel="vcard:org">
              <y:type typeof="vcard:Organization">
                <y:meta property="vcard:organization-name">Yahoo!</y:meta>
                <y:meta property="vcard:organization-unit">Yahoo! Research Barcelona</y:meta>
              </y:type>
            </y:item>
          </y.type>
        </y.item>
      </y:adjunct>
    </content>
  </entry>
</feed>
```

### Queries:

foaf:Person/foaf:name = Peter Mika

foaf:Person/vcard:org/vcard:organization-unit = Yahoo! Research Barcelona

Fig.5

[1] Ricardo Baeza-Yates, Aristides Gionis, Flavio Junqueira, Vanessa Murdock, Vassilis Plachouras, and Fabrizio Silvestri. The impact of caching on search engines. In SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pages 183-190, New York, NY, USA, 2007. ACM.

# Twine Beta Progress Report

*Nova Spivack gives us an update on the beta of social interaction application: Twine*

By Nova Spivack



Twine (<http://www.twine.com>) is a new service for keeping up with content related to your interests that is powered by the Semantic Web. This article will briefly explain Twine, discuss what we have learned from the beta test period and where Twine is headed as a product.

Twine is focused on “interest networking”. As one of our product managers recently put it: “Twine is a serendipity engine.” The goal of the product is to help you keep up with, manage information, and discover new things that relate to your interests. In Twine, serendipity is facilitated by artificial intelligence and collective intelligence

and twines (groups) they might find interesting. We also use natural language processing to automatically read and tag content with semantic tags for relevant people, organisations, places and other concepts. We are currently working on topic detection and trend detection to help facilitate even better discovery in the future.

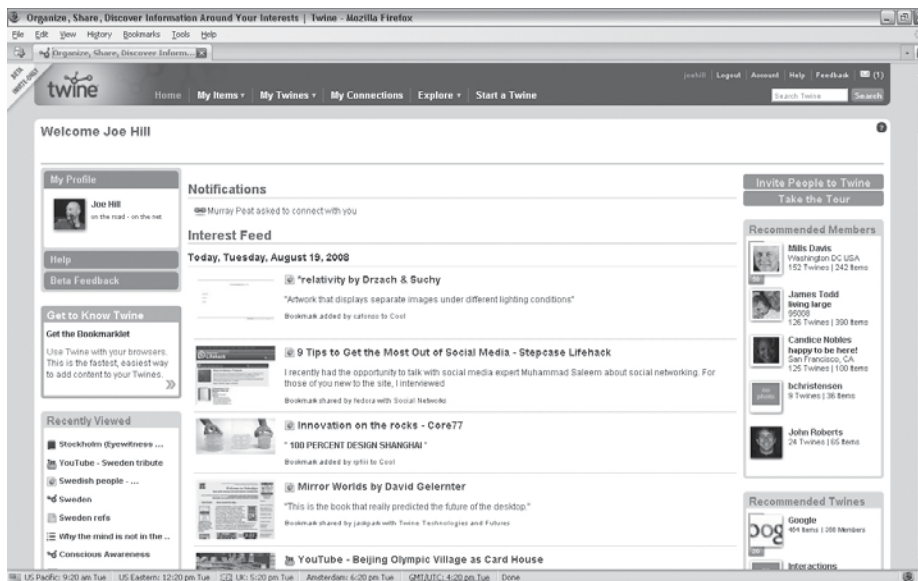
Twine applies collective intelligence to help find, organise and distribute interesting content. The users of Twine collectively scour the Web to find the most valuable content about various interests they have. They then share those items with various relevant twines (interest groups) they belong to, as well as with individuals that have relevant

technical level, Twine is a large and growing knowledge base comprising of a semantic graph. This graph connects and defines semantic objects that represent people, organisations, places, groups, bookmarks, videos, pictures, books, products, documents, emails, comments, tags, notes, and many other types of information. The graph in Twine runs on a custom-designed software platform that is fundamentally based on RDF and OWL, the open-standard languages of the Semantic Web.

There are many reasons why we chose to use the Semantic Web to build Twine. First, we want it to be easy to create and change data types and relationships in our data set. Eventually, we want to make it easy for others to do that as well. In addition, we want to make it easy to add other data from outside Twine. This vision for a Web of Data was an important factor in our decision to use the Semantic Web rather than more traditional data models.

Twine is an ambitious project. We chose to innovate on a couple levels – the product itself, and the underlying platform and technologies we built it with. This choice to innovate on both levels has been a blessing at times and a challenge at others. We began our original research in 2003, when the concepts and standards of the Semantic Web were still forming. There were few tools available for building semantic applications at that time and so we had to build our own development environment and platform and even our own triplestore (a form of database for storing semantic graphs).

We started by focusing on core R&D, eventually developing a prototype for a peer-to-peer personal knowledge sharing environment on the desktop. This became the basis of our later consulting work with SRI on a DARPA-funded



working together. Twine is different from social networking because the focus of our product is on tracking interests not on the social activity of friends and colleagues.

Twine applies artificial intelligence techniques to learn about individual interests and make recommendations to help users discover new content, people

interests. Members discuss and further propagate the content. We will roll out collaborative filtering capabilities in the form of a ratings system soon. The collective intelligence aspects of Twine are equally, if not more, important than the artificial intelligence.

All of the capabilities of Twine are powered by the Semantic Web. On a



research project (CALO), which focused on exploring next-generation solutions for collaboration and knowledge management. After 2 years of CALO work, we raised outside venture funding and began to develop a completely new, Web-centric platform on which to build our future commercial products and services. Our first product, Twine, was announced in October of 2007 and went into invite-beta in February of 2008.

We chose to be one of the first companies to build commercial products with the Semantic Web, and it has been a challenging road at times. Pioneering is never easy, but it offers the opportunity to reach territories, rich with untold opportunities, before anyone else. Our journey to-date has yielded us nearly 20 patent applications and a range of new

technologies that we are quite proud of. And of course, Twine, which is quickly becoming a widely-used product.

Twine has been in beta for approximately 6 months at the time of this writing. Today we have around 40,000 active users participating in the beta. They have created nearly 16,000 twines about various interests and they have contributed hundreds of thousands of pieces of content. There have also been over 40 articles written about Twine since we announced the product.

By all accounts the beta has been a success and the product is off to a good start. After many years of work to lay the groundwork, this is extremely gratifying to the team. The most valuable part of this process is what we have learned

from our users. We have conducted several usability studies, a survey of our users, and we have been actively discussing Twine with our users on the site on a daily basis. In fact, there has been so much user-participation in testing Twine, that we might say that Twine is actually now "user-driven software." Users have helped the team by providing suggestions, creating content, experimentation and catching bugs.

The chief learning from our beta test so far was that the first version of Twine was not easy enough to understand, making the learning-curve too steep. We have worked extensively in subsequent releases (we release every 3 weeks) to address those issues and the site is already much simpler and clearer. In particular we refactored major parts of the user-interface, changed the user-orientation process, and have added better help and tutorials. There is still more work to do on this front, but we have made solid progress.

Another major area of feedback was around functionality. Users have asked for a number of key features and capabilities, and the team has been busy working to deliver those. Chief among the requests is the ability to import bookmarks from other sites. This feature will be available in the next release. Several other important features have centred on improving search functionality and an API, both of which are under development. We are also working on speeding up the site. By implementing caching strategies, optimising the software, and data storage behind Twine, we will soon be making very big improvements in speed of page loads and queries.

The goal of our present work is to get Twine ready for mainstream use in early Q4 of 2008. After that, we will focus on rolling out a number of new features, including:

- better personalisation and recommendations



- integrating Twine with external services and data
- additional types of content to create and share
- customisation and extensibility
- improving relevance and surfacing more semantics
- new ways to use and create semantic data in Twine

It is important to point out that the underlying semantic platform, and the semantic graph behind Twine has only barely been exposed in the present version of the product. There is a lot more potential there, and we are planning to deliver on that in 2009. Our strategy has been to get the basics right

first, and then to start to surface more of the advanced semantics and capabilities of the product in future releases.

Ultimately, we want to make Twine the smartest place on the Web for individuals and groups to keep up with their interests. The beta has been an exciting and fruitful learning experience for the team and has already yielded many improvements to the product. As expected, during the beta period, the user-base has largely been comprised of early adopters. As we head into 2009, we will begin rolling out Twine to much wider audiences and focusing on everyday Web surfers as our core target user -

eventually making Twine into a product that will pass the most difficult hurdle of all: "The Mom Test."<sup>1</sup>

*Nova Spivack is the CEO and Founder of Radar Networks, the developers of Twine. His public twine can be found at: <http://www.twine.com/twine/1p2dqhdx-1jg/nova-spivack-my-public-twine>*

Nova Spivack  
San Francisco, CA, 94107

I am the CEO and founder of Radar Networks. Radar Networks is the maker of Twine.

Here is my bio...

Nova Spivack is a technology visionary and entrepreneur with nearly two decades of experience in pioneering ventures. In 1994, Mr. Spivack co-founded EarthWeb, one of the first Internet companies, where he was Executive Vice-President for Products, Strategy and Marketing. EarthWeb went public in 1999 and resulted in the Nasdaq's largest IPO single-day percentage point gain up to that point, spawning a wave of Tech IPOs. Mr. Spivack left EarthWeb's board of directors in 1999 and began advising startups and angel investing. During the down-years of the post-Internet-bubble, EarthWeb's content properties were acquired in 2000 by Internet.com. The company's Dice.com property remained a strong stand-alone business until it was acquired for approximately \$200 million in 2005.

While at EarthWeb he helped key cultural institutions and businesses develop their first large-scale Web presences, including the New York Stock Exchange, the Metropolitan Museum of Art, BMG Music Club, Sony, AT&T, US West, and others. He also helped to catalyze the adoption of Java technology by leading the production of large on communities for the IT professionals, including Gamelan.com, Developer.com, and Datamation.com.

Prior to EarthWeb, Mr. Spivack worked in a variety of roles from technology marketing to software engineering at artificial intelligence and next-generation computing ventures including Individual, Inc., Ray Kurzweil's pioneering OCR company, Kurzweil Computer Products which was sold to Xerox, and at Danny Hillis' legendary supercomputing venture, Thinking Machines. Mr. Spivack is also the founder of Lucid Ventures, an early-stage incubator that originated the technologies that are now Radar Networks. Mr. Spivack is a co-founder of the San Francisco Web Innovators Network (SFWIN), a network of several hundred technology innovators and business leaders who meet monthly in the Bay Area.

Mr. Spivack has extensive experience working on knowledge representation and the Semantic Web, and has authored and helped to design several large (500 to 3000 class) ontologies in the OWL language, the W3C open standard for ontology specifications. Mr. Spivack has also been a lead advisor to SRI International on the DARPA CALO program, a distributed research program encompassing several hundred top researchers across over 20 major research institutions focused on next-generation semantically-aware machine learning applications, and in particular on the IRIS Semantic Desktop project. Also with SRI and Sarnoff Laboratories, Mr. Spivack helped to co-found nVenture, SRI's in-house technology incubator.

Mr. Spivack has co-authored several books on Internet strategy and technology and led the EarthWeb Press publishing imprint with Macmillan Computer Publishing, one of the largest computer book publishers, which resulted

**Connect**

**Profiles**

**Nova's Connections**

- jayjurisich  
www.jurisich.com
- shal  
on vacation in Paris  
Budapest
- Twain  
I think, therefore I  
TWINE  
EC (Ether Cloud)
- Alexander Moore  
Chicago
- Bent Rasmussen  
Intertwined & Entangled  
Denmark
- Jack D. Logan  
Things are ... lookin' up!  
San Diego
- Lauren D

<sup>1</sup> The Mom Test is the most difficult challenge for a technology product to pass, perhaps even more difficult than the famous Turing Test, proposed by the computer scientist Alan Turing. While the Turing Test merely tested the ability of a computer to fool a human into thinking it was intelligent, the Mom Test is defined as "Making a technology product that moms will consistently use, by their own initiative, without technical training or support from anyone else, on a daily or weekly basis."

# Generation Zero

*Ian Davis, CTO of Talis, discusses the state of Semantic Web applications, and asks where the next apps will come from.*

**By Ian Davis**



RDF, the cornerstone of the Semantic Web technology stack recently celebrated its ninth birthday. At the time of its inception the notion

of a World Wide Web of data was rather obscure and the case for RDF wasn't helped by a impenetrable specification and murky syntax. The situation improved in 2004 when the W3C issued several documents that completely revised and clarified RDF. Their goal was to ensure that this important technology could be understood and used by as many developers as possible. However, today there are still only a handful of applications that incorporate RDF at their heart and none of these are using the full potential of the Semantic Web. Talis is one of several companies who are on a quest to radically change this situation and engender a whole new generation of applications that use and enhance the web of data.

The Semantic Web vision is of a world full of reusable data. Rather than data being created over and over again to varying degrees of quality, the Semantic Web enables it to be created once and reused many times. The Semantic Web technologies allow the cost of data production and maintenance to be shared across all stakeholders rather than being incurred multiple times. RDF makes interoperability and collaboration more efficient than other standards such as XML because it allows data publishers to encode and publish the meaning of their data along with the data itself. Additionally, by linking the data together like the World Wide Web links documents, the data becomes easier to discover and, critically, it becomes easier to automate that discovery.

None of these notions is new, however. They have been part of the Semantic Web vision from the start, and yet - even after almost a decade of intense work by the computer science community - we still do not see any compelling applications that incorporate a deep use of this proposed global web of data. What we do see are applications that are using some of these ideas to give better user experiences and achieve efficiencies in their processing of raw data. These applications can be thought of as generation zero of a new wave of powerful applications, tentatively exploring the technology space and evolving towards the first generation of truly connected applications.

*"These applications can be thought of as generation zero of a new wave of powerful applications..."*

Until recently one of the strongest impediments to applications participating in the Semantic Web was the lack of real, usable data. That barrier has been demolished by members of the Linking Open Data project who, over a period of about eighteen months, converted and published dozens of datasets comprising billions of individual facts. While this is still a tiny fraction of the data required for the large-scale success of the Semantic Web, it does demonstrate that even a modest volunteer effort can create and maintain significant quantities of readily-reusable data.

However, despite huge quantities of linked data being available for use, there are still no applications that take advantage of these rich resources and reuse them. This is not solely a Semantic Web and RDF phenomenon. The Microformats movement shares a similar data reuse philosophy and also suffers from a lack of serious applications. Microformats are standardised patterns of HTML markup that can describe common types of data such as events and contact information. They have been enthusiastically adopted by many large websites resulting in millions of facts being readily available; yet there are almost no applications that consume this data. The few that do are focused on the importing of data such as friend lists as a one-off process rather than a continuous part of the user's interaction with the application.

The experience gained from the Linking Open Data project and the Microformats community demonstrates that despite the availability of large volumes of useful data, the uptake by applications of that data has been very slow. For this situation to improve, three things need to happen: there needs to be a shift in the mindset of the application developers; there needs to be infrastructure in place to support the huge quantities of data that these applications require and there needs to be a legal framework which empowers applications to use and reuse data efficiently.

The first of these requires a radical rethink by application developers to deal with the issues around trust, reliability and openness on the web of data. Traditionally, applications have been built with the expectation that they control all of their data. They are tied to one or two databases that are under control of

the application's owner or developer. In a world of data reuse, this assumption is no longer applicable. The data accessible by an application could come from anywhere, and could have varying degrees of detail and/or suitability for the application's task. In some circumstances the data may be deemed to be untrustworthy. The developer has to compensate for this situation, and this adds complexity to their applications. As my colleague Nadeem Shabir wrote in the very first edition of Nodalities: "It's a completely different way of thinking about the problems we are trying to solve and the applications we are trying to build".

*The core of any significant Semantic Web application is its "triple store"...*

This kind of "open world" thinking is entirely novel for the vast majority of developers, so the full effects have yet to surface in most organisations. Adoption of this new attitude to application development has profound implications for the nature of companies and their operations. Most of the data in the world today is held inside businesses that invest huge amounts of effort creating and curating that data and then exchanging it with their commercial partners. These are the stakeholders that stand to gain the most from reusing and sharing data on the Semantic Web and this revolution in thinking needs to reach deep into these organisations much as the radical thoughts of the agile programming community has done.

The picture around the second requirement, that of vastly improved infrastructure, is also a positive one. The core of any significant Semantic Web application is its "triple store"; which is software that acts as a database of facts. It is critically important to these applications that their triple stores are capable of managing the huge volumes of data that are needed to provide their users with rich and compelling experiences. The triple store

software industry is at the same level of development as that of the relational database industry a quarter of a century ago. Even the well-established triple stores are unproven at high scale usage and none have the maturity that 25 years of stiff commercial competition and unforgiving customers brings.

It is not a coincidence that three companies that have produced the most prominent commercial applications with RDF at their heart - Talis, Garlik and Radar Networks - have all developed their own infrastructure to address their scalability and reliability concerns.

At Talis we hope that by opening up our infrastructure to other developers we can enable them to take advantage of the learning's we have made around building and running highly available and scalable applications. As a result, as much of the complexity of scaling up is taken away, we expect to stimulate the creation of many more Semantic Web applications targeting many different markets.

The final concern, that of appropriate licensing of data, has seen significant progress in the past year and is again an area in which Talis has been playing a leading role. Historically, there has been a great deal of confusion around the terms under which data can be reused and the extent to which copyright law can be applied. This has made it difficult for commercial applications to confidently connect to and use the web of data. In collaboration with Science Commons, Talis funded the legal work that resulted in the creation of the first licence specifically for data: the Public Domain Dedication and Licence. This licence, now under the guidance of the Open Data Commons organisation, specifically addresses the issues around whether an application has the right to reuse data. Applications are free to use, reuse, adapt and transform any data licensed under its terms and as such it opens a way for reuse of data on a massive scale.

So it seems that the last barriers to developing applications that realise the full Semantic Web vision are at last

crumbling. We have much more clarity around the licensing of data, we have scalable infrastructure becoming more widely available and we understand some of the education needed in the development community to enable the creation of the first generation of connected applications.

Talis Engage, Talis Prism, DataPatrol and Twine are good examples of generation zero Semantic Web applications. They have taken their first steps towards this new world of global data sharing by putting the core technologies to good use. As Nova Spivack explains elsewhere in this edition of Nodalities: using RDF gives Twine the ability to absorb and interpret data regardless of the source. Garlik's DataPatrol and Talis Engage also make use of this flexibility to represent a huge variety of types of data.

As they evolve to become first generation, they will have to overcome the hurdles of the open world and begin connecting and reusing data directly from the source. They will need to use linked data to discover the answers to questions at runtime; enrich the information they are providing and ensure that the information is as 'live' as possible. Additionally they will need to participate fully in the web of data to extend their capabilities beyond the behaviours programmed into them by their developers. By using the Semantic Web to communicate with other instances of their software and across vendor boundaries to other complementary applications, they have an opportunity give their users a rich, vibrant and connected experience.

## The Semantic Web Gang

The Semantic Web Gang is a monthly round-table podcast hosted by Paul Miller and featuring a regular panel of commentators on the Semantic Web.

[semanticgang.talis.com/](http://semanticgang.talis.com/)



# A Call to Arms

John Sheridan and Jeni Tennison explore the London Gazette and the public sector information space

By John Sheridan and Jeni Tennison



Government information can make a substantial contribution to the

development of the semantic web, yet the public sector has been slow to embrace the opportunities afforded by this technology. Why?

The UK Government is keen to unlock the benefits of public sector information (PSI) reuse, opening up its data for both economic and social gain. The Steinberg / Mayo Power of Information Report published in 2007 makes a powerful case for government engaging with online communities and releasing its data. This agenda has been embraced enthusiastically. Ministers like Tom Watson rightly talk about information as infrastructure. The Power of Information Taskforce is pushing forward initiatives such as the Show Us a Better Way competition. In return for a £20,000 prize, the competition asks "Do you think that better use of public information could improve health, education, justice or society at large?" and invites people to share their ideas.

New APIs and datasets have been released as part of the Show Us a Better Way competition, but the approaches for disseminating this data vary considerably - from SOAP and RESTful APIs through to simple spreadsheets. The data may be available, but that does not make it accessible on the semantic web. The Office of Public Sector Information (OPSI) is responsible for the government's information policy and specifically the management of Crown copyright. OPSI is keen to find strategies and approaches that enable serendipitous reuse. As Benkler's "The Wealth of Networks" discusses, public sector information

(or more specifically, free-to-reuse information as part of the commons) is a major factor driving innovation on the web. Web-enabled PSI supports new forms of social production of information and knowledge.

Semantic web technologies have enormous appeal, designed as they are for large-scale information aggregation. But we only gain the full benefit of the semantic web when substantial quantities of data have been made available. It is hard to write the business case to justify the expenditure of public money on semantically enabling government information, when the benefits hinge on others doing the same. The public sector is an environment where decisions are made on the basis of evidence and are

open to public scrutiny. You need to do more than take a "leap of faith". So approaches to exposing the semantics of government data need to be both pragmatic and achievable, without requiring wholesale changes to existing IT infrastructure.

Using content negotiation to serve RDF data to the web or offering up SPARQL endpoints are prohibitively complicated and difficult for many public sector organisations, which tend to be reliant on established suppliers, most of whom have yet to fully understand or embrace the semantic web. The barriers to action are as much to do with perceptions as they are real.

The screenshot shows the 'The London Gazette' website interface. On the left is a navigation menu with links: Home, About the Gazette, Browse, Search Tools, My Account, My Notices, Services, Placing a Notice, Help, and Registered Users (with fields for Username and Password). The main content area is titled 'Search Results' and shows details for a notice dated 24 July 2008, Issue Number 58775, Page number 11179. The notice is titled 'Water Resources' and 'Environment Agency', specifically 'WATER RESOURCES ACT 1991 (AS AMENDED BY THE ENVIRONMENT ACT 1995)'. The text of the notice describes an application for consent to discharge up to 16.25 cubic metres per day of Secondary Treated Sewage Effluent to Starr Carr Drain at National Grid Reference TA 11459 46587 from Land South of Cheyne Lodge, Starr Carr Lane, Leven Road, East Yorks, YO25 8RU. It also includes a statement that any person wishing to make representations should do so in writing to The Environment Agency, Water Quality Permitting Support Centre, PO Box 4209, Sheffield, S9 9BS, during the period 24/7/08 to 4/9/08 quoting reference NPSW00003364.

A typical notice in the London Gazette. The "Semantic Radar" plug-in for Firefox indicates the presence of RDFa

To push the reuse agenda forwards, OPSI is taking a key information asset it controls - the London Gazette - and is exposing it on the semantic web using RDFa. The London Gazette is the government's official newspaper and contains a very broad range of information. With over 250 different types of statutory notice, the London Gazette covers such diverse topics as transport, planning, the environment, insolvency, through to veterinary medicines and unclaimed premium bonds. Crucially, the London Gazette is a source of this information with an established provenance. With new notices published every day there is a temporal dimension to the data; the publication date of a notice in the London Gazette is often considered the point when that information entered the public domain and is used as a trigger point by the Courts and others.

Our work to semantically enable the London Gazette has two aims: firstly to find a practical way of publishing PSI in a way that maximises its re-use potential, and secondly to give the London Gazette a new role. Whenever a piece of legislation says that information must be published in the London Gazette, it will in effect ensure that information is made publicly available, in a consistent way and in a reusable form.

To serve Gazettes information we are using RDFa, which has been designed to bridge the human-readable web and the data web. This approach, currently being standardised by the W3C, allows RDF data to be embedded in XHTML and XML documents, by adding specialised attributes. The idea that your XHTML pages can serve a dual purpose of providing human-readable information and computer-understandable data has been around for a few years. Much of the recent progress is largely thanks to the microformats community. With GRDDL, both microformats and RDFa can be exposed to RDF-aware applications as RDF.

For the London Gazette we chose to use RDFa because:

- Microformats are deliberately limited to existing and widely adopted types of information, such as events and contact information. The Gazettes hold information of other sorts, such as references to legislation and Gazette-specific metadata, for which there are no microformats.
- RDFa offers more flexibility for dealing with some of the semi-structured text that we have to deal with. For example, it allows us to talk about the date and location of a meeting without having an element that wraps that date and location together.

What is particularly appealing about RDFa from the government's perspective is that existing database driven websites

*We have much still to learn about the use of semantic web technologies and their applicability to government information on the web*

can be more or less 'tweaked' to serve RDF data. The government has many such sites, serving lists of schools or hospitals, providing statistics and other information, underpinned by well-structured and defined data.

In principle, it should be as easy to add appropriate attributes to surface the semantics of the data displayed in the browser, so that these web pages can be parsed and the RDF data simply extracted by semantic-web applications.

The extent of the tweaking varies depending upon your starting point, but the typical webmaster in a government department knows how to implement a particular mark-up for a type of information (e.g. a job vacancy). Here then is potentially an easy(ish) route for

semantically enabling PSI and a design pattern that others can adopt and use.

Implementing RDFa for a website ought to be straightforward but it depends on how well the website adheres to standards-based design in the first place. In the case of the London Gazette, assigning consistent and persistent URIs to notices and ensuring the presentation of valid XHTML were pre-requisite steps that took time and effort. There were other requirements like the need to develop and reference a simple but bespoke DTD to support the namespaces used in the RDFa. In terms of ontologies, while there are schema's like Dublin Core and FOAF, one of the benefits of using RDF is the ability to "roll your own". Gazettes data references a lot of other information that does or should have its own ontologies. We reference the Administrative Geography ontology developed by the Ordnance Survey and have defined an ontology for legislative references, for example. Our hope is that these ontologies can then be more widely re-used and owned by appropriate organisations.

Perhaps the largest challenge is identifying the semantics in the Gazette data. Some notices are well-structured, containing known fields that are already captured using forms. But most are semi-structured, with the majority of the semantics embedded in free text. Even where the semantics of a phrase are clear, the editors who enter the text typically do not mark them up, or if they do they mark the phrase for style rather than semantics. Our aim is to automate the majority of the mark-up process by recognising key features such as postcodes and company names along with keywords specific to the type of notice. This should both ease the burden on editors and raise the quality of the semantic mark-up we publish.

Finally is the need to alert re-users to the presence of RDFa enabled web pages. We are assessing the use of Atom feeds and the sitemaps protocol as approaches for the London Gazette. Over the next few months we will be serving an increasing number

of Gazettes notices using RDFa and improving the quality and extent of RDFa that we currently provide.

We have much still to learn about the use of semantic web technologies and their applicability to government information on the web. At first glance, RDFa offers an important design pattern and a viable route for serving PSI to the semantic web; one which departments could practically adopt and use. What is needed now are more public sector ontologies. The taxonomists in government have been creating controlled vocabularies for years, from fish types to military vehicles – so we have many of the raw ingredients. The challenge is turning these into ontologies and providing them with appropriate URLs so that they can join the web of Linked Data. Using RDFa with the London Gazette we have a basic recipe for applying semantic web technologies to an existing government website and have developed techniques to overcome common problems.

This is a call to arms. What happens next hinges on how well the data is used. If the community responds as we hope,

with imaginative and creative uses of the data, we can build momentum for semantically enabling an increasing range of PSI assets. Take a look at the RDFa available in the Transport, Planning, Environment and Water notices on the London Gazette website at <http://www.london-gazette.co.uk/>. If you are interested in public sector information and the semantic web, would like to reuse data from the London Gazette or are planning to build an application using Gazettes data, then we would love to hear from you. You can contact us at: [john.sheridan@opsi.x.gsi.gov.uk](mailto:john.sheridan@opsi.x.gsi.gov.uk) and [jeni.tennison@tso.co.uk](mailto:jeni.tennison@tso.co.uk).

*John Sheridan is Head of e-Services at the Office of Public Sector Information. Jeni Tennison is an independent consultant currently contracted to The Stationery Office.*

## This Week's Semantic Web

by Danny Ayers

Catch up every week at [blogs.talis.com/nodalities/category/this-weeks-semantic-web](http://blogs.talis.com/nodalities/category/this-weeks-semantic-web)

## The Semantic Web Gang

The Semantic Web Gang is a monthly round-table podcast hosted by Paul Miller and featuring a regular panel of commentators on the Semantic Web.

[semanticgang.talis.com/](http://semanticgang.talis.com/)

# Nodalities Magazine 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](http://www.talis.com/nodalities) or email [nodalities-magazine@talis.com](mailto:nodalities-magazine@talis.com).

# Discovery and Usage of Linked Datasets on the Web of Data

By Michael Hausenblas



The Web of Documents (or Web x.0 or whatever) is a success due to the fact that it is actually lacking a feature. We all experience it every single day but not

many of us seem to notice it. Yes, I'm talking about search engines. And, yes, the missing feature I'm talking about is that there is no central registry for content. In my opinion, this decentralised aspect made the Web a huge success story.

Could you imagine reporting your new blog post, Wiki page or whatever you have to hand to an authority that takes care of adding it to a 'central look-up repository'? I can't, and there is at least one good reason for it: such things don't scale. However, there are ways to announce and promote the content. An entire industry - the Search Engine Optimisation (SEO) people - seems to make reasonable money with it. Basically, SEO-people will tell you what to put in the <head>, what other HTML elements to use, what to do with links and more in order to get your content seen by as many people as possible.

In the Web of Data we face a somewhat similar problem. Currently, over 40 linked datasets (cf. the nice introduction of linked data by Tom Heath in issue 1 of Nodalities Magazine) are available, and it seems that just about every week a new one pops up. This is great. However, in a couple years' time there might be thousands of them, many of them likely offering broadly comparable content. Now, the main challenge is: how can I, as someone who wants to build an application on top of linked data, find and select appropriate linked datasets? Note that there are two basic issues here: first, finding an appropriate dataset

(discovery) then selecting one - that is, you have a bunch of possible candidates, which one is the 'best suited'.

Whatever solution one can think of, it has to scale to the size of the Web. As we have learned from the Web of Documents and SEO, the basic idea is to let the owner of a linked dataset do the 'annotation part' (i.e. telling about the content and the features of her dataset) and then let a search engine - or semantic indexer, for what it's worth - do the dirty job of crawling these descriptions. The result is a highly flexible and scalable system where dataset consumers can obtain the information they need using a 'single point of access', while ensuring that the dataset providers have the freedom to announce and promote their content as they need.

There are some existing technologies such as semantic sitemaps, DARQ, UMBEL, POWDER, etc. that certainly can contribute to the scenario presented above. However, experience shows that still something essential is missing. Our take on this is the 'Vocabulary of Interlinked Datasets' (voiD) [REF-VOID]

a joint effort carried out by members of four institutions (Talis, DERI, University of Oxford, and JOANNEUM RESEARCH). Currently we are working on the use cases and gathering requirements which allow us in turn to develop the vocabulary.

Basically, voiD allows the description of content and the interlinking between linked datasets. For example one could state 'there are 200k links of type foaf: depicts between dataset X and dataset Y; and dataset Y mainly offers data about cars'. What is so nifty about this is the fact that this is done in RDF - again yielding yet another linked dataset (self-descriptive vocabulary). You may wonder where the magic happens, but to tell you the truth, there is nothing miraculous going on. The way we envision this is simply that when one is offering a linked dataset she will also publish a voiD description along with it.

In Fig. 1 the overall setup is depicted. The hook is the sitemap [REF-SITEMAP] with an according extension [REF-SEMSITEMAP] that gets crawled by a search engine (1). The sitemap

```
prefix dc: <http://purl.org/dc/elements/1.1/>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix void: <http://vocab.dowhatimean.net/neologism/void-
tmp#>
SELECT ?dataset FROM <http://sw.joanneum.at/void/demo/
demo3.rdf> {
    ?dataset a void:Dataset;
        dc:subject <http://dbpedia.org/resource/
Proceedings> .    ?datasetSrc a void:Dataset;
        foaf:homepage <http://dbpedia.org/> ;
        void:containsLinks ?linkset .
    ?linkset void:target ?dataset.
```

Fig.1



references the void description of the linked dataset which in turn is indexed by the search engine (2). As soon as the search engine has the RDF triples from the void description in its index it can answer arbitrary complex queries (3) such as:

Basically, this example query will return all datasets that have data about proceedings and which are linked from DBpedia. You may want to test this query yourself and see the result at [REF-VOID]. Finally, equipped with the list of 'best suited' datasets from (3), the consumer can exploit the data for his own purpose (4). In this setup, void plays an important role: as it defines the semantics of the terms used in the description of a dataset (such as void:containsLinks) it allows publishers to accurately promote their data and consumers to effectively find them.

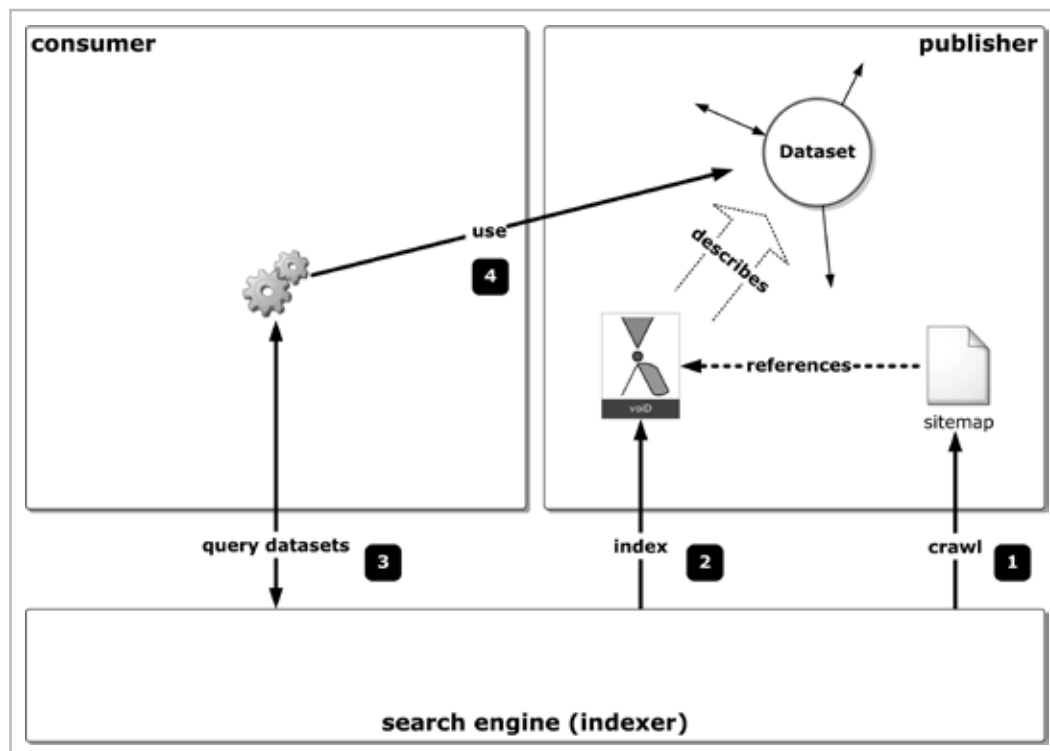
Our next steps regarding void include the creation of a stable and usable vocabulary and the publication of a

guide that explains how to use void from the perspectives of both consumers and producers. There are, however, other issues that matter in a real-world setup—trust being one of the most prominent ones that comes into mind. Selecting a certain linked dataset for for use, say, in a Web application is typically not only influenced by the content it offers. If two datasets both offer data about books, for example; which would you choose to use: the one from a rather dubious, unknown guy or the one which is published by an institution with a good reputation? Again, widely deployed Web of Data vocabularies (such as FOAF) can help to build a Web of Trust; and in the case of void, to establish the trust between a consumer and a publisher.

Although we have only recently started with our work, the feedback we get from the community is very promising. Many people seem to care not only to publish on the Web according to the linked data principles but also to properly describe what they offer. Let us together make the

Web of Data such a success as the Web of Documents by simply missing out a feature. Stay tuned!

*Michael Hausenblas is a senior researcher at JOANNEUM RESEARCH, an applied research company in Austria, working in the area of multimedia semantics utilising Web of Data technologies in a couple of national and international (EU) projects. Michael and his team have contributed to the linked data community project by publishing a dataset, working on discovery issues (void), proposing "User Contributed Interlinking" and addressing issues with multimedia interlinking. Currently, Michael is finalising his PhD.*



[REF-VOID] <http://community.linkeddata.org/MediaWiki/index.php?VoID>

[REF-SITEMAP] <http://www.sitemaps.org/>

[REF-SEMSITEMAP] <http://sw.deri.org/2007/07/sitemapextension/>



# ESTC2008

2<sup>ND</sup> ANNUAL EUROPEAN SEMANTIC TECHNOLOGY CONFERENCE  
24-26 September 2008 Palais Niederösterreich Vienna, Austria

## **SAVE THE DATES**

# REGISTER NOW

[www.estc2008.com/register](http://www.estc2008.com/register)

Europe's Business Platform of Semantic Technologies  
Actively participate and maximize your benefit from ESTC2008

---

24-26 September 2008  
Palais Niederösterreich, Vienna, Austria

---

[www.estc2008.com](http://www.estc2008.com)

# The Need for next Generation Technology in the study of Medieval History and the Crusades

By Dr. Nicholas Morton



History is everything - literally. Everything has a past from cathedrals to igneous rocks. Even those who claim that 'history is bunk' are not merely

plucking words out of the air but are making a statement that is based on past experience, using words which have etymological roots.

Over recent years numerous academic disciplines have begun to employ technologies which can draw new historical data from their sources. To this end, the Ocean floor has been combed for treasures, patterns of ancient land-use and settlement have been identified, bone fragments have been analysed for information on diet and culture. With these new techniques our understanding

*With these new techniques our understanding of our past has been dramatically enhanced.*

of our past has been dramatically enhanced.

Many of these advances have affected the field of medieval history. The study of tree rings has supplied information about the climatic conditions of the past centuries. Developments in Archaeology have cast light on the everyday life and material culture of this period. In these ways, many of the items that have come down to us from this period have been mined for their

secrets. Nevertheless, perhaps the most obvious place to look for the history of the middle ages is in the documents and narratives that were written at the time. In most cases these are our best record from the period. Such sources are a treasure trove of information with the illuminations, the calligraphy, the vellum or parchment each conveying messages to a trained eye. These elements have each been studied separately and in some cases modern techniques such as carbon-dating have developed our understanding. Even so, the words themselves - the actual messages of the text - have offered a seeming full-stop to the advance of technology. After all, the word 'dominus' remains 'dominus' however you look at it - there is no further meaning for computer technology to prise free.

As a result, although technological advances in related disciplines have provided useful context for the information contained in these works, it has not been until recently that applications have been found which can draw further messages from the text itself. These modern applications generally revolved upon the swift comparison of information drawn from thousands of documents through relational databases. With such technologies materials form hundreds of different sources can be cross-referenced in minutes, rather than years; providing a crucial tool for the historian. The applications of such databases are numerous and, for example, if you wanted to establish the itinerary of King Henry II of England and you had a database of all the documents issued by him then you could run a query on the place and date of issue of each charter to create a dot-to-dot impression of his progress around Western Europe.

This need for the comparison of volumes of information is a symptom of a growing trend within current historiography. Previously - and this is a gross oversimplification - historians tended to rely on the histories and chronicles written by contemporaries in the medieval period. After all, these are the most obvious source of information; if you want to learn about the First Crusade then where better to look than in a chronicle of this expedition written by one of the participants. Later historians, however, used a wider range of sources to build upon the world described by these authors. To take the above example, then the accounts of crusaders could be supplemented with financial documents which explain how such expeditions were paid for. Legal documents could help to explain some of the issues that absent crusaders could cause at home. Letters supply further information that supplements the narratives. Through such wider analyses,

*Through such wider analyses, a more rounded impression of the world of the crusades has been be established.*

a more rounded impression of the world of the crusades has been be established. Even so, as more and more materials began to be consulted there arose a need for computer applications which could compare tremendous quantities of data at speed - hence the need for relational databases.

One key aspect of these advances has been in a sphere of prosopography. Prosopography is the compilation of information on different people's lives. A prosopographical database supplies biographical data on a number of individuals and also highlights the links - familial or otherwise - between them. They have a range of applications. For example, if you wanted to establish why a group of 8 soldiers fought alongside one another during the siege of Damascus during the Second Crusade you might enter their names into such a database which would explain that three of them were brothers and the remaining 5 were hired mercenaries whose services had been purchased by the elder brother - thus answering your question.

Given this potential, there have been a number of major projects which have created prosopographical databases to support the work of historians. Recently, a team led by Katharine Keats-Rohan has created a database entitled, *The Continental Origins of English Landowners, 1066-1166*. Among a range of objectives it, 'provides a prosopographical register of all persons occurring in English administrative

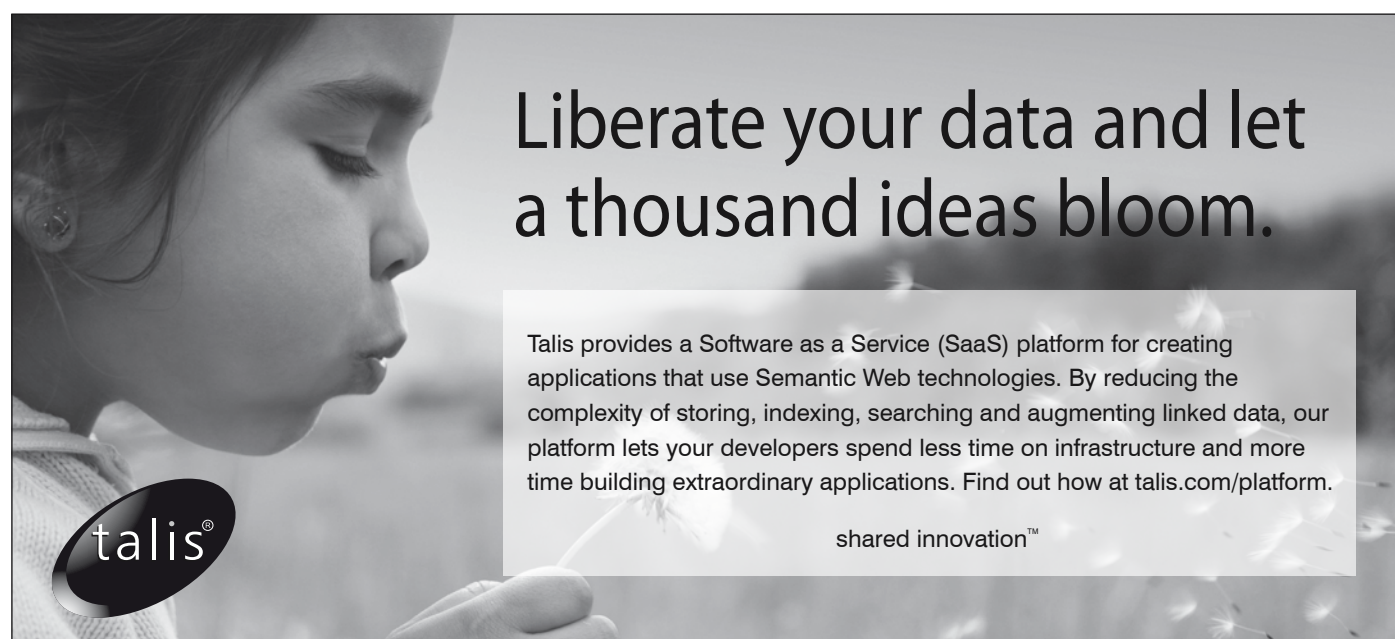
documents written in the century after the Norman Conquest' (quotation taken from the COEL website). The database itself highlights the connections between individuals and therefore helps to outline the blueprint of relationships between the most influential individuals of the medieval England.

Currently, at Royal Holloway University, there is a further project which will result in a database containing information upon every crusader to travel to the Holy Land along with his/her background, history and personal connections. This is a particular challenge given that crusaders were recruited from every region which paid allegiance to the papacy. As a result, it is necessary to draw upon tens of thousands of different sources from many countries in many languages. Nevertheless, with the resulting database, it will be possible to trace the international connections of loyalty, blood and heritage which bound so many of the crusading expeditions together.

One of the main features of this project is that it calls for the latest applications


of modern database technology. It is not sufficient simply for a separate record on each crusader to be logged because, like the above database by Keats-Rohan, the nature of the relationships between thousands of individuals needs to be traced; creating a highly complex web of information. The resolution of this issue poses a number of challenges; however these can be overcome with modern semantic technology which can highlight such connections. These examples serve as a demonstration of the possible applications of modern database technology for medieval studies. It reflects a growing need within current historiography to use semantics to draw together information from a range of sources allowing the true nature of the Crusades and the Middle Ages to be thrown into relief

*Dr. Nicolas Morton is a lecturer in medieval history and the Crusades at Royal Holloway and Queen Mary Universities. He is also currently researching database resources for further study from the Crusades period.*



# Liberate your data and let a thousand ideas bloom.

Talis provides a Software as a Service (SaaS) platform for creating applications that use Semantic Web technologies. By reducing the complexity of storing, indexing, searching and augmenting linked data, our platform lets your developers spend less time on infrastructure and more time building extraordinary applications. Find out how at [talis.com/platform](http://talis.com/platform).



shared innovation™



# What Next-Generation Technology Could Offer Historians

By Tom Heath



Dr. Morton's piece above nicely captures the richness and complexity of the world around us. Take almost any topic, explore it in

the slightest depth and you'll find an intricately woven tapestry of connections between seemingly diverse entities and artefacts. Legal documents are not the most obvious place to turn to expand our understanding of a topic such as the Crusades; financial accounts perhaps even less so. What these examples illustrate is that connections pervade every aspect of our lives, and show themselves in some unlikely places.

This intricate web of connections is what Tim Berners-Lee has referred to as the Giant Global Graph. Far from being "just" a network of interconnected computers or a Web of linked documents, the GGG represents the interwoven fabric of the world around us – the people, places, events and ideas that tie us together. Having reaped the rewards of connecting computers via the Internet and connecting documents via the Web, it is now time to bring this graph to the foreground and make first class citizens of these connections.

Just as we do not function in isolation from the world around us, so we cannot study history without appreciating the context in which events occur. To do so requires that we bring all relevant sources into the scope of investigation, however heterogeneous they may appear.

Dr. Morton illustrates the richness of these sources and highlights the need for relational databases capable of representing connections between disparate entities. These don't need to be relational databases in the conventional

sense, and in fact they shouldn't be. Whilst capable of providing the storage back-end for many diverse and powerful applications, conventional relational databases are relational only in a local context. What they do not provide is a global infrastructure that enables entities in one data set to be connected to those in another, especially where the second is remote and under the stewardship of a third party. What is needed is an infrastructure with which to represent the Giant Global Graph, irrespective of where the individual pieces of the jigsaw may reside.

If we wish to trace the movements of King Henry II across Europe, in the manner that Dr. Morton describes, not only do we need a rich data set of charters issued, but also the accompanying geographical information through which we can contextualise and visualise these movements. By definition historians are not geographers—at least this may not be where their greatest talents lie. Rather than relying on historians to create and maintain such a geographic database, this activity should be left to experts in the field, with others simply linking to the relevant entities in the database and consuming the data they require for their applications.

With a little effort these sorts of arrangements are feasible with conventional relational database technology - at least they are when only a handful of data sources are in the frame. The problem arises when we step beyond a specific, limited context and accept that vast numbers of data sources each reflect aspects of the Giant Global Graph. What is required is a mechanism with which to link related data across disparate sources.

This form of distributed database is precisely the goal of the Semantic Web movement, and is enabled by the Semantic Web technology stack. At the bottom of this stack are URIs that start with `http://` that we currently associate with the addresses of Web pages, but can be used to identify anything we like. Using HTTP URIs to identify all types of things - not just Web pages, but monarchs, places and historical events - provides a distributed infrastructure for globally unique identification, that is also coupled with a mechanism for retrieving information about those things.

For example, if I have some data about King Henry II of England, I might choose to create a URI to identify him, such as `http://monarchsdb.org/england/henryii`. If someone looks up that URI in a Web browser, I can choose to provide some relevant data from my database in response to that request, and can also choose to provide links to related data from other sources. The Resource Description Framework (RDF) is the format of choice for representing data in this distributed database, and creating the typed links that put the Web in Semantic Web.

The databases described in Dr. Morton's piece sound incredibly exciting. I hope that as these activities move forward they will adopt the principles of Linked Data and the Semantic Web. In doing so they will inherit not only a technology stack with which to publish their data on the Web for others to reuse, but a means to expose the rich connections that make up the historical fabric but can so easily remain buried.

# Finding Answers on the Semantic Web: What S said to W.

By Mathieu d'Aquin and Vanessa Lopez



Our story begins with two characters, which are neither turtle

nor Greek hero, attending Tim-Berners Lee's keynote speech at the WWW 2008 conference. After the talk, they gather at the bar for a casual discussion about their favourite topic: the Web. Let's call them S and W, which, while these are obviously not their real names, suit them quite well as the main characters of our story.

**W:** You know, He is always talking about the Semantic Web, but to be honest, I still don't get it.

**S:** Really? The great giant graph, information available for machines to process, web-scale intelligent applications...

**W:** I understand the principle, the vision, the idea, but I still don't see how this sort of technology benefits the user. But as you are being so passionate about it, maybe you could try to convince me.

**S:** Of course I am being passionate about it: there is a great deal out there that can be used to build a new kind of applications, something completely different from what has been done before.

**W:** Yeah, of course, but this is all geek things for geeks. What about the real users? For example, how can this make a difference in the way people query the Web?

**S:** Oh... I see. And what if I tell you that it is possible to actually ask questions to the Semantic Web and obtain answers?

**W:** You mean like when querying Google?

**S:** Of course not! I'm talking about actually getting the answers to your questions thanks to Semantic resources,

like asking "Who are the members of the rock band Nirvana?" and getting a list of names, not a list of documents containing the words "Nirvana" and "member".

**W:** This is nice, but I have seen it all before, question answering systems. They need to be fed with huge quantities of information, and are still limited to the one domain of application they are meant to work on. This is not even getting close to Google.

**S:** This is actually the beauty of it. You don't have to feed the system with knowledge acquired the hard way anymore, it can just use all the semantic information available on the Web thanks to Semantic Web technologies. In other terms, it just has to get the list of members of Nirvana from an RDF dataset somewhere on the Web.

**W:** But this would require to find this dataset, and to extract the answer from it, all that at run-time. How could it be done?

**S:** Well W, thanks to the Watson Semantic Web Gateway...

## The Watson Semantic Web Gateway

The goal of the Watson Semantic Web gateway is to support the development of a new generation of applications that dynamically retrieve, exploit and combine online semantic resources, instead of relying on ontologies that are gathered, processed and integrated manually at design time. It is a kind of Semantic Web search engine that uses a set of dedicated crawlers to collect online semantic content. A range of validation and analysis steps is then performed to extract several different types of information from the collected semantic documents: information about their content, about the employed languages, about their relations with

other documents, etc. This information provides a valuable base of metadata, exploited for indexing, searching and characterizing the content of the Semantic Web. Watson also provides a set of Web interfaces that allow human users to employ a variety of access mechanisms on the collected data (<http://watson.kmi.open.ac.uk>).

The core components of Watson are a set of Web Services and APIs which support the development of Semantic Web applications by providing various functionalities for searching and exploring online semantic documents, ontologies and entities. The major functionalities are:

- finding Semantic Web documents through sophisticated keyword based search, allowing applications to find, for example, semantic documents containing an individual Nirvana and a relation containing member.
- retrieving metadata (the size, language, label, logical complexity) about these documents, to help select the ones that are more suitable in the given scenario.
- finding specific entities (classes, properties, individuals) within a document, e.g. for locating the individual Nirvana.
- inspecting the content of a document, i.e., the semantic description of the entities it contains. This can be used to get the description of Nirvana (genre, dates and... members).
- applying SPARQL queries to individual Semantic Web documents, for example to get all the instances of RockBand.

The combination of mechanisms for searching semantic documents (keyword search), retrieving metadata about these documents and querying

their content (e.g., through SPARQL) provides all the necessary elements for applications to select and exploit online semantic resources, without even having to download the corresponding ontologies. Watson is a gateway to the Semantic Web as it provides applications with the ability to exploit the large-scale, distributed body of knowledge that is constantly growing on the entire Semantic Web, in a lightweight and robust fashion.

## Interlude

**W:** So, you can build applications that dynamically make use of the wealth of semantic information available on the Web.

**S:** Yes, and there are many applications already exploiting this idea, for semantic browsing, word sense disambiguation, etc.

**W:** But this is still pretty much for Semantic Web developers. I still don't know how real users can interact with the Semantic Web.

**S:** That, my friend, is where PowerAqua enters our story.

## PowerAqua: Deriving Answers from the Semantic Web

To some extent, PowerAqua (<http://kmi.open.ac.uk/technologies/poweraqua/>) can be seen as a straightforward human interface to Watson, and therefore to the Semantic Web. Using this interface, a user can simply ask a question, like "Who are the members of the rock band nirvana?" and obtain an answer, in this case in the form of a list of musicians (Kurt Cobain, Dave Grohl, Krist Novoselic and other former members of the group). The main strength of PowerAqua resides in the fact that this answer is derived dynamically from any dataset available on the Semantic Web.

Without going into the details, PowerAqua first uses some linguistic components to transform the question

into a set of possible "query triples", like <person/organization, play, rock group Nirvana>. The goal of these query triples is to translate at the same time the structure of the question (in this case, a "who" question) and its vocabulary (using synonyms) in a form that can be easily aligned with online ontologies. The next step consists in finding, thanks to Watson, online semantic documents describing entities that correspond to the terms of the query triples (obtaining for example, an individual Nirvana which is a rock band from a dataset about music). All the terms of triple queries are matched onto entities of the selected datasets, to transform the triple queries into formal queries that rely on the corresponding ontologies. For example, for in our music dataset, the query <Nirvana, has\_members, ?x: Musician> is generated. In a final step, PowerAqua simply has to extract answers from these many datasets by executing the corresponding queries, again using directly the Watson infrastructure. Several sources of information, coming from various places on the Web, may provide overlapping or complementary answers. These are therefore ranked and merged according to PowerAqua's confidence in the contribution of particular elements to the final answer.

Many details are skipped in this description (the interested reader can refer to [2] for more technical details on PowerAqua). To put it in a nutshell, starting from a question is natural language, PowerAqua locates, on the Semantic Web, datasets able to answer the question and transforms the question into queries directly executable on these datasets. Thanks to its linguistic components, it can generate "Semantic Web queries" from natural language, and take benefit from the Watson infrastructure to efficiently execute these queries, without the need for a complex local knowledge base. In this way, at run-time, i.e. within a few tens of seconds, PowerAqua is able to find answers to

your questions from anywhere on the Web.

## Conclusion

**W:** This is fantastic! It is like in this story, "A Logic Named Joe" [3]: a tool exploiting a network of resources containing the answers to all your questions.

**S:** It is even better as it constantly evolves from the contributions of thousands of people around the word.

**W:** Does it really work that smoothly?

**S:** To be honest with you W, there is still a lot of work to do to make it fully operational, but the foundations are basically here and it proves that it is feasible. Not only that, it also opens the way to new perspectives.

**W:** Like what?

**S:** Like the possibility to link it to classical Web navigation, providing the user not only with the straight answer to her question, but also with a link to the biography of Kurt Cobain, or to an online store to listen to, and buy, Nirvana's albums.

**W:** Hey S, you are getting passionate again! Now, you will think I am an annoying imaginary person, but there is one thing that still bothers me: Nirvana does not exist anymore, and some of the group members have only been part of it for a time. How is that handled?

**S:** This is actually an excellent question

**W:** but it is late already, so I guess this will have to be another story

Dr. Mathieu d'Aquin is a research fellow at the Knowledge Media Institute, the Open University concerned with tools and infrastructures supporting Semantic Web application development.

Vanessa Lopez is a research fellow at KMI, researching natural language front ends to query the Semantic Web.

[1] Mathieu d'Aquin, Enrico Motta, Marta Sabou, Sofia Anagnostou, Laurian Gridinoc, Vanessa Lopez and Davide Guidi. Towards a New Generation of Semantic Web Applications. *IEEE Intelligent Systems*, Vol. 23, Issue 3, May/June 2008. [2] Vanessa Lopez, Enrico Motta, Victoria Uren. PowerAqua: Fishing the Semantic Web. *Proceedings of the European Semantic Web Conference, ESWC 2006, Montenegro*. [3] Murray Leinster. *A Logic Named Joe*. *Astounding Science Fiction*, March 1946..

REGISTER NOW AT [WWW.DEFRAGCON.COM](http://WWW.DEFRAGCON.COM)

## The Defrag conference is accelerating the "aha" moment!

As online data is growing and fragmenting at an exponential pace, individuals, groups and organizations are struggling to discover, assemble, organize, act on and gather feedback from that data.

In the largest sense, we're all looking to augment the pace at which we achieve insights on raw data -- to accelerate the "aha" moment.

Defrag is focused on the tools and technologies that accelerate the "aha" moment, and is a gathering place for the growing community of implementers, users, and thinkers that are building the next wave of software innovation.

Defrag explores the intersection of topics like:

- Enterprise 2.0
- Online Collaboration
- The Implicit Web
- Collective Intelligence
- The Semantic Web
- Mash-ups
- Social Networking in the Enterprise
- Next-level Discovery

### Discussion Topics:

- The Quantification of Everything
- The Democratization of Data
- Social Search in the Corporate Environment
- Can Identity filter information overload?
- Finding serendipitous content through context
- Fixing Foundational information channels

### Featured Speakers:



Stowe Boyd



William Duggan



Esther Dyson



Paul Kedrosky



Charlene Li

REGISTER NOW AT: [WWW.DEFRAGCON.COM](http://WWW.DEFRAGCON.COM)