> "*The concept of machine-understandable documents does not imply some magical artificial intelligence which allows machines to comprehend human mumblings. It only indicates a machine's ability to solve a well-defined problem by performing well-defined operations on existing well-defined data. Instead of asking machines to understand people's language, it involves asking people to make the extra effort.*" Sir Tim Berners-Lee, W3C.

What's up in XML-based text processing, from news feeds to the Semantic Web.

Introduction

This paper addresses the blurry boundary between hard data in databases and 'free' text – or 'unstructured' data. There are a number of intersecting fields of study close to this boundary. Text can be stored within a database, and hard data can be stored – as metadata, within a document management system. Meaningful searches work at this boundary – and need to be able to extract information from many sources – perhaps out on the web – and in a variety of published formats. Existing technologies make good use of full text search – *à la* Google – but as everyone has experienced, these can return a glut of information of doubtful provenance. Other search techniques apply 'artificial intelligence' to fine tune full text search by counting word occurrences in a document or by noting proximity of key words to extract context.

But our focus in this paper is not the search technology itself, but rather the various technologies which seek to better separate form and content in documents, and to add contextual information to make searching more robust. Adding context or metadata implies robust taxonomies which in turn impact structured data and information sharing. Context and shared taxonomies form a large part of the World Wide Web Consortium's (W3C) Semantic Web initiative – billed as the 'next generation' web. This in turn overlaps with much that has been termed 'knowledge management'. Another overlap is with data management itself – where much interest has focused of late on sharing metadata – well names etc. – across different applications.

A word of warning on the cutting-edge nature of some of these technologies; despite the intent to expose machine readable metadata to support interoperability, some of the mechanisms for this are not quite ready for prime time. We try in this paper to distinguish what is 'ready to run' from emerging specifications which may change or die off before reaching the marketplace.

Sources

My interest in what I later discovered was the Semantic Web (semweb) came from our online publication Oil Information Technology Journal. A subscriber to the online edition www.oilit.com asked if we had an 'RSS Feed'. RSS is a standard way of presenting news headlines and pointers to articles along with author and publication metadata. RSS stands for 'Really Simple Syndication' and also for RDF Site Summary – a foretaste of a degree of conflict and confusion in this emerging field. To fulfill this request we read Ben Hammersley's book[1] on RSS. This helped us to implement a

[1] Content Syndication with RSS – Ben Hammersley, O'Reilly, 2003.

simple RSS feed – this really is not rocket science, RSS feeds are deployed in many web logs (Blogs).RSS is based on another W3C standard, RDF – which led us on to Shelley Powers book[2] on RDF. Here, what really caught our attention was Powers' mention of the facts that she was previously with Halliburton, had worked on a POSC project and furthermore stated categorically that '*RDF/XML would fit the needs of POSC [..] if the interest were on merging data between organizations for more effective supply chain management*'. Now we had a new acronym and a pointer to a new technology that might impact our business. Other books followed – see the references section at the end of this abstract. This initial interest culminated in our attendance at the W3C's Technical Plenary in Cannes earlier this year. We met with some of the key players in this field and were privileged to hear from the horse's mouth where the technology is moving today.

From HTML to XML

HTML has proved immensely successful –because it allows naïve users to develop web pages simply and quickly and offers (usually) a fairly intuitive interface. The 'hard wired' formatting of vanilla HTML has proved a drawback as web pages are deployed on a wide variety of devices of different resolutions and capabilities. XML was designed in part to fix this, by better separating content and form (with XML documents carrying the information payload, and stylesheets the formatting). XML also introduced the capability of defining ad-hoc tags as required, allowing for unambiguous metadata and context to be embedded in web documents. We'll look next at a simple XML-based text application – the one that sparked off our interest in this subject – RSS.

Really Simple Syndication (or RDF Site Summary)

RSS is a way of sharing (or syndicating) news feeds. It is a simple way of presenting a headline, summary and metadata as to the publication and author. It is best understood by working through a simple feed – this is a doctored version of the Oil IT Journal feed – available at the url http://www.oilit.com/rss/RSSFeed.xml.

RSS feed part one…

```
<?xml version="1.0"?>
<rdf:RDF
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns="http://purl.org/rss/1.0/"
>
```

These 'scene setting' instructions are key to XML's robustness. Every XML document should contain references to its format. In the above example, the first reference is to the Resource Description Format (RDF) – more of which later – and the subsequent reference points to the RSS format itself. These references are supplied as an XML 'namespace'. The namespace limits conflicts due to accidental re-use of the same terminology – so a potentially ambiguous '<name>' tag is unique for the namespace. This referential completeness will become important later as we see how properly structured documents can be read ('parsed' in the jargon) and meaning retrieved by visiting the namespace for further details about the tags.

RSS feed part two…

[2] Practical RDF – Shelley Powers, O'Reilly & Associates, 2003.

```
<channel rdf:about="http://www.oilit.com/rss/RSSFeed.xml">
     <title>Oil IT Journal</title>
     <link>http://www.oilit.com</link>
     <language>en-us</language>
     <description>
     Oil and gas information technology newsletter and website.
     </description>
</channel>
```

This is the top level metadata about the feed – with the journal's title and home page, a description of the contents and scope which will be used by the feed readers to summarize the feed's content and scope. Looking further down the RSS Feed we get to the payload…

RSS feed part three…

```
<item>
     <title>A million miles of spaghetti eaten every day!</title>
     <link>http://www.oilit.com/2journal/2article/0403_3.htm</link>
     <description>Oil IT Journal editor Neil McN…</description>
</item>
```

Here we have a title, a link to the article and a 'description' – in the form of a sub-header. That's just about it for the complexity of the RSS format. The end result is shown below – in a freeware feed reader.

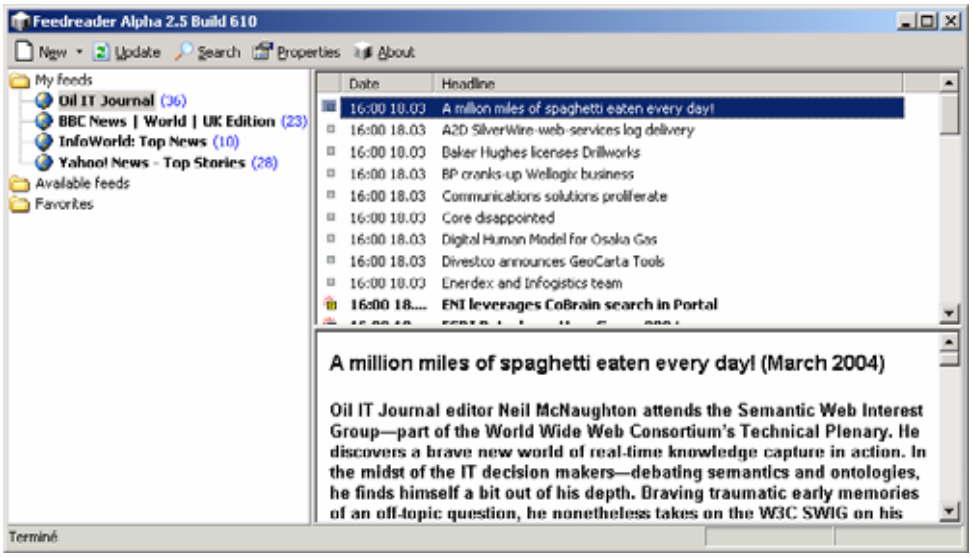

Figure 1 RSS feed – as viewed in FeedReader[3].

The point of all this? RSS provides a simple standard way of presenting a newsfeed – one that can be read and indexed by machines – or 'robots'. These simple programs[4] visit feeds on a regular basis and inform subscribers when a feed has been updated. A few minutes after we update the Oil IT Journal feed, subscribers using newsreaders are informed automatically of the new news.

[3] http://www.feedreader.com.

[4] Despite the way 'robots' are presented they are far from rocket science, performing a simple TCP/IP 'GET' to the RSS link every now and then – just like a browser does when you retrieve a web page. When they see a change, they refresh the feed.

Figure 2 Read RSS feed on your SmartPhone[5].

This is simple, standards-based publish and subscribe. Oil IT Journal receives several hundred hits per day from RSS robots – overkill for a monthly publication!

Figure 3 News pops up[6] as it comes in...

Resource Description Format

One issue with sharing metadata across different applications is that even if all applications use XML, and are therefore accessible, a lot of effort may be required to delve down into a complex XML document – visit with the namespaces and schemas to figure out what exactly is the scope and context of particular bits of information. Step in the Resource Description Format (RDF) already used above in the RSS feed. RDF (at least in this context) can be used to 'escape' from a complex XML document and point the parser to an info-element such as a piece of metadata. The idea is that for some uses, it may be overkill to figure out the whole XML document – but by scanning through looking for a particular RDF namespace, key metadata information may be obtainable from the XML document to help with its classification or other top level usage. The

[5] FeedBurner (www.feedburner.com) on Nokia 6600 SmartPhone.

[6] FeedReader – www.feedreader.com.

trick behind RDF is the 'triple' – a simple data modeling construct of subject, property and value as follows…

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#">
           <contact:fullName>Neil McNaughton</contact:fullName>
           <contact:mailbox rdf:resource="mailto:neil@oilit.com"/>
        </contact:Person>
</rdf:RDF>
```

There are two information 'triples' embedded in the above XML snippet as follows..

- A 'Person' whose 'fullName' property has a value of 'Neil McNaughton'.
- A 'Person' whose 'mailbox' property has a value of 'neil@oilit.com'.

The potential usefulness of the construct can be imagined by considering an ensemble of well related XML documents – say future versions of XML-based data exchange documents from different organizations such as POSC, PPDM, software and data vendors and companies. The use of RDF triples to make statements like

- A 'well bore' whose 'name' property has a value of '31/5B'.
- A 'well bore' whose 'elevation' property has a value of '33 meters'.

The use of namespaces as above can point to fuller descriptions of the resources or to value lists or taxonomies as we'll see later.

Dublin Core

RDF is a mechanism for embedding format-neutral metadata in an XML document – but where do the metadata descriptions come from? One widely used source is the Dublin Core metadata specification. This provides a standard, interoperable mechanism for describing resources such as books, documents, web pages, images etc. with a high-level description of items such as 'title', 'creator', 'subject', 'description' and includes references to unique object identifiers such as ISBN for books or the Universal Resource Locator (URL) for web pages. Dublin Core references can be used in HTML as follows to provide unambiguous, machine readable collateral…

```
<META NAME="dc.creator.e-mail" content="info@oilit.com">
<META NAME="dc.creator.name" content="Oil IT Journal">
```

Dublin Core is also amenable to the RDF treatment and is used to augment the RSS specification with information such as subject and locally-defined taxonomies[7].

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:taxo="http://purl.org/rss/1.0/modules/taxonomy/"
  xmlns="http://purl.org/rss/1.0/"
>
  ...
  <item rdf:about="http://c.moreover.com/click/here.pl?r123">
```

[7] This example from Web Resource - http://web.resource.org/rss/1.0/modules/dc/

```
    <title>XML: A Disruptive Technology</title>
    <link>http://c.moreover.com/click/here.pl?r123</link>
    <dc:subject>
       <rdf:Description>
          <taxo:topic rdf:resource=http://dmoz.org/Computer../XML/ />
          <rdf:value>XML</rdf:value>
       </rdf:Description>
    </dc:subject>
    <dc:subject>
       <rdf:Description>
          <taxo:topic rdf:resource="http://www.oreillynet.com/.." />
          <rdf:value>Data: XML</rdf:value>
       </rdf:Description>
    </dc:subject>
  </item>
```

Taxonomies

The Dublin Core metadata spec is of very limited scope – and is unlikely to satisfy a vertical like the oil and gas industry. Companies themselves have been trying hard to standardize corporate metadata by using taxonomies or catalogues of accepted names for many subjects such as wells, fields, accounting terms etc. Some of the work done by Shell Expro is the UK is now available through POSC[8] and has been widely disseminated by Flare Consultants[9]. Taxonomies have also been deployed inside oil and gas companies as corporate-standardized drop-down lists. These are used to capture data items like seismic field tapes, well tapes and other physical items in many ad-hoc developments. But while these efforts are important, there is a missing link here for two reasons. First, a taxonomy in the form of a drop down list – or even in the form of a Microsoft Word or Excel document – is not machine readable and is not therefore sharable across different applications. Second, it is rather unlikely that a single taxonomy will ever be generally accepted – even within a vertical like oil and gas. RDF-based taxonomies attempt to solve these two problems by exposing machine readable taxonomies which allow for 'discovery'. One example of such taxonomy deployment is seen in the RSS specification – and is used to provide an extended list of topics relevant to the RSS channel[10]…

```
<taxo:topic rdf:about="http://meerkat.oreillynet.com/?c=cat23">
    <taxo:link>http://meerkat.oreillynet.com/?c=cat23</taxo:link>
    <dc:title>Data: XML</taxo:title>
    <dc:description>A Meerkat channel</dc:description>
  </taxo:topic>

  <taxo:topic rdf:about="http://dmoz.org/Computers/Data_Formats/Markup_Languages/XML/">
    <taxo:link>http://dmoz.org/Computers/Data_Formats/Markup_Languages/XML/</taxo:link>
    <dc:title>XML</taxo:title>
    <dc:subject>XML</dc:subject>
    <dc:description>DMOZ category</dc:description>
    <taxo:topics>
```

[8] See POSC EpiCAT on http://www.posc.org/technical/epicat/epicat.shtml.

[9] http://www.flare-consultants.com.

[10] Example from - http://web.resource.org/rss/1.0/modules/taxonomy/.

```
      <rdf:Bag>
        <rdf:li resource="http://meerkat.oreillynet.com/?c=cat23">
        <rdf:li resource="http://dmoz.org/Computers/Data_Formats/Marku.../SGML/">
        <rdf:li resource="http://dmoz.org/Computers/Programming/Internet/">
      </rdf:Bag>
    </taxo:topics>
  </taxo:topics>
```

Ontologies

The simple 'triple' construct can be used to build extremely complex data structures. This (unfortunately?) seems to be where much of the W3C semantic web effort is going today. It would have been nice to present a straightforward, widely accepted means of sharing the simple stuff – namespaces and value lists (taxonomies) – but this is not enough for the research community – who seem to want the semweb to run before it walks. The 'running' takes the form of 'ontologies' – complex interleaving RDF graphs representing 'meaning' in a machine readable format. The idea again is to build machine-readable graphs of knowledge representation. The web ontology language (OWL) uses RDF to build parent-child and other more complex relationships. Parsers should be able to 'reason' by traversing graphs from different ontologies to arrive at conclusions. At the risk of confusing everyone, here follows a different approach to a taxonomy deployment – using multiple DAML[11] collections showing how semweb schemas can be combined through the namespace:

```
<rdf:RDF xmlns:xsd="http://www.w3.org/2001/XMLSchema";
   xmlns:daml="http://www.daml.org/2001/03/daml+oil#";
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#";
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#";>
<rdf:Description rdf:about="#InternalIntention2Type">
<daml:oneOf rdf:parseType="daml:collection">
  <rdfs:Class rdf:about="#intention" />
</daml:oneOf>
</rdf:Description>
<rdf:Description rdf:about="#intention">
<daml:oneOf rdf:parseType="daml:collection">
  <rdfs:Class rdf:about="#threat" />
  <rdfs:Class rdf:about="#sarcasm" />
  <rdfs:Class rdf:about="#joke" />
  <rdfs:Class rdf:about="#earnestness" />
  <rdfs:Class rdf:about="#request" />
  <rdfs:Class rdf:about="#truth" />
etc…
```

Conclusions

Just as RSS was a quick win for our website, other semweb technologies may prove quick wins for the deployment of taxonomies and for embedding sharable metadata in hardcore, vertical XML standards such as WITSML and the other 'xxxML's deployed in the industry. At this level, semweb technologies are extremely simple to leverage – as

[11] By David Dodds and Oasis – see http://lists.oasis-open.org/archives/humanmarkup-comment/200206/msg00057.html

witnessed by the near-triviality of RSS. But RSS works and has brought tangible benefits to the blogging and news reading communities at extremely low cost. Other equally trivial XML-structured text formats offer context capture on the fly for KM-related initiatives like WIKIs[12]. From our discussions at the W3C plenary we observed that current commercial semweb applications (notably Adobe's suite of applications) focus on the (relatively) simple side of RDF – the exposing and sharing of metadata from within documents, databases and XML files. Pushing the semweb boat out further into the uncharted waters of ontologies and machine-derived 'meaning' may be hazardous – but what a great field of research. Unfortunately for real-world users, much current research is focused on blue sky stuff – building complex, experimental 'reasoning' systems. The quick win field of deploying sharable taxonomies is in danger of being passed over without consolidation into a standard. Even the existing ways of using RDF are under threat – as moving XML standards begin to deprecate some of the early decisions.

As Tim Berners-Lee remarked in our opening quote, it is the 'extra effort' that people make capturing data and information at the source that will drive the semantic web. The data management and GIS communities are getting used to recording metadata up front. The 'report writing' community – and that is just about all of us – should begin to appreciate the extra leverage that can come from filling in a few text boxes before filing a document for posterity. In fact one approach to hardwiring consistent metadata into documents is through slightly more intelligent authoring systems – that for instance offer drop-down pick lists for insertion of well names, fields, associates etc. into the body of documents as they are written.

Neil McNaughton
info@oilit.com
Consultant – The Data Room
Editor – Oil IT Journal
April 2004

[12] WIKIs are collaborative web pages which can be edited by a whole community – much as in a discussion group. The XML-based format can be used to capture author and other information. See also IRC – collaborative, online 'chat' – recorded for posterity in XML.

References

Much information on the **Semantic Web** is available from the horses mouth – in other words from the World Wide Web Consortium – www.w3c.org. Look for Semantic Web - http://www.w3.org/2001/sw/ and RDF - http://www.w3.org/RDF/.

Information on RSS 2.0 is available from http://blogs.law.harvard.edu/tech/rss RSS 1.0 from http://www.purl.org/rss/1.0/ and Dublin Core from http://dublincore.org/.

Books

Content Syndication with RSS – Ben Hammersley; O'Reilly & Associates, March 2003, ISBN 0-596-00383-8. An authoritative account – well written and a good introduction to XML-based text processing.

Practical RDF – Shelley Powers; O'Reilly & Associates, July 2003, ISBN 0-596-00263-7. Interesting comments on POSC and projections as to the usability of RDF in taxonomies. Going gets tough as topics get to ontologies and the harder-to-explain stuff.

Spinning the Semantic Web – Dieter Fensel et al. MIP Press 2003, ISBN 0-262-06232-1. A collection of papers of widely differing styles and scope. Based on a seminar held in 2000 – so somewhat dated.

The Semantic Web – Michael Daconta et al. Wiley Publishing – 2003, ISBN 0-471-43257-1. Broad if rather uncritical coverage of Semantic Web topics and interoperable knowledge management.