South African Journal
of Information Management
[DoE accredited]

Student Work Vol.6(1) March 2004

Tim Berners-Lee's Semantic Web

G Kück
gregk@leafwireless.com
Leaf Wireless (Pty) Ltd
Post Graduate Diploma in Information Management
Rand Afrikaans University

Contents

1. Introduction
2. Introducing the Semantic Web dream
3. Method to the madness?
4. Semantic Web: the technologies
5. Issues and possible problems
6. Conclusion
7. References

1 Introduction

We all know what the Internet is. It is an enormous and complex network of computers and numerous other smart devices, all connected, sharing information between themselves across a variety of telecommunications media. It is an important business and information tool accessible from local and remote locations that uses the TCP/IP protocol suite to serve up Web pages via HTTP, files via FTP, electronic resource management systems via SNMP, electronic mail via POP3, SMTP and IMAP4 and data for wireless devices via WAP, etc. What it is not, however, is smart.

The current Internet environment is fundamentally a publishing medium. It is a mechanism through which data, in the form of images and text, are made available for public or personal consumption. Just as one magazine cannot interact with the content of another magazine, neither can the typical content of one Web site interact with the content of another Web site unless specific mechanisms are built into each to allow such interaction. The World-Wide Web is a tangle of information that, through the implementation of hyperlinks, allows a browser to navigate – usually quite randomly – from one Web site to another. The meaning, context and applicability of the content of each Web page needs to be interpreted by the human reader.

To laymen users accessing the Internet from their home computers, it appears as if the Web server through serving up Web pages and information to their desktop PC is interacting with their personal computer. In truth however, this is not the case. All that is typically occurring is a Web server responding with a preformatted – or predetermined format – page of static text, regardless of how dynamic the visible content is. This preformatted text is then interpreted – in computer terms, which is not to be confused with the human concept of understanding – by the Web client and displayed in accordance with this predetermined format. The browsing computer understands and reacts to the mark-up tags used to format the page, but it is clueless as to the actual content of the Web page. To interpret the content, some form of human interaction is required. But this is all about to change, or is it?

top

2 Introducing the Semantic Web dream

The concept of the Semantic Web is the brainchild of the original creator of the World-Wide Web, Tim Berners-Lee. The idea behind the Semantic Web is 'to weave a Web that not only links documents to each other but also recognises the meaning of the information in those documents.' (Frauenfelder 2001); in other words, to transform the current Web from a series of interconnected, but ultimately semantically isolated data islands into one gigantic, personal information storage, manipulation and retrieval database.

According to Berners-Lee, Hendler and Lissila (2001), 'most of the Web's content… is designed for humans to read, not for computer programs to manipulate meaningfully. Computers can adeptly parse Web pages for layout and routine processing – here a header, there a link to another page but, in general, computers have no reliable way to process the Semantics…', or the meaning of the content of the page.

Tim Berners-Lee sees it as being an extension of the current World-Wide Web that will bring a common structure to the content of Web pages, thereby providing such content with meaning which will allow external software agents to carry out sophisticated tasks on behalf of the reader or user and, as such, promote a greater degree of cooperation between humans and computers. In so doing, a new age of computing will be ushered in where machines are better able to 'process and "understand" the data that they merely display at present' (Berners-Lee *et al*. 2001).

This vision of a Semantic Web can therefore be viewed from three different perspectives: (a) a type of universal library which can readily be accessed and used by humans in their day-to-day information acquisition; (b) the backbone for software or computational agents to utilize autonomously in order to perform particular activities on behalf of their human counterparts; and (c) a 'method for federating particular knowledge bases and databases to perform anticipated tasks for humans and their agents' (Marshall and Shipman n.d).

2.1 Universal library

The concept of turning the Web into a universal library was at the heart of the earliest vision of a Semantic Web, and arose as a reaction to the chaos and disorder of the World-Wide Web. At this time there was a very real threat of great volumes of data being unreachable or accessed in an inefficient manner, and a general push towards taming the Web was favoured. Fortunately, Google and AltaVista came along with improved indexing and retrieval algorithms and to a large degree sorted this problem out. Since then, the focus of Semantic Web visionaries has changed slightly from a universal catalogue system to one consisting of the global cooperation of Web authors, seeing it as more of an extension of the current system rather than a remodelling of the existing Web.

2.2 Knowledge navigator

'The Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users' (Berners-Lee *et al*. 2001).

The idea behind this is to markup the content of a Web page in such a way as to create both a human-readable and machine-readable version of the Web page simultaneously. The machine-readable version can then be used by software agents to filter, search and prepare data in ways that can assist the human user while browsing.

2.3 Federated data or knowledge base

Federated data or knowledge base involves components of the Web being built in such a way as to have a prior knowledge of one another or, at the very least, to discover one another or anticipate the types of applications that will use the information they provide.

'The Semantic Web will provide an infrastructure that enables not just Web pages, but databases, services, programs, sensors, personal devices and even household appliances to both consume and produce data on the Web' (Berners-Lee, Hendler and Miller 2002).

top

3 Method to the madness?

According to Berners-Lee *et al*. (2001) in order 'for the Semantic to function, computers must have access to structured collections of information and sets of inference rules that they can use to conduct automated reasoning.' This idea is in no way new, in fact artificial-intelligence researchers were toying with and studying these types of systems before the Web even existed. What is needed is a way of representing knowledge that allows computers to both interpret it in the traditional sense of displaying it on screen in a human-readable format, as well as understanding it at a computer level, thereby allowing the computer to autonomously react to such knowledge.

Traditionally, knowledge representation systems have been centralized, meaning that in order for every person – or thing for that matter – to share the same meaning of a concept, they all need to share the same definition of that concept. In other words, for entities to communicate efficiently, they all need to understand what is meant by a particular concept, speak the same language and be able to participate in, or at least associate with, the particular context in which the concept is used. For example, to comprehend the concept of 'can', the interpreter needs to first be familiar with the language in which the concept is expressed – in this case English – in order to be able to generate meanings. But what does the concept mean? Is it a container, as in a can of beans? Does it relate an ability of the speaker, as in 'he can drink lots of beer'? Or is it being used as a euphemism for something else, as in 'going to the can'? To determine this, the interpreter needs access to other extraneous or contextual information. Unfortunately, it is easy to see that a system such as this can quite rapidly spiral out of control, making managing and controlling it more of a challenge than understanding it. The residents of the British Isles and North America all apparently speak the same language, but one country's 'trunk', is another's 'boot' and one's 'color' is the other's 'colour'.

In computer terms, a centralized knowledge system, such as this, with its own idiosyncratic rules, severely limits the types and extent of questions that the computer can answer reliably – or even at all. Fortunately, the aim is not to eliminate all paradoxes and unanswerable questions; instead it is to create rules that are 'as expressive as needed to allow the computers to reason as widely as desired.' This means that, to reduce the 'confusion', the language being used needs not only to express data, but to also express the rules governing the interpretation and reasoning about such data. Effectively, this implies that the Semantic Web will follow the traditional Web, in the sense that it too will be based on a decentralized model whereby each content provider will also provide the mechanisms necessary for any machine or smart device or appliance to interpret the content that is being provided.

According to Tim Berners-Lee, the concept and philosophy surrounding the Semantic Web is not dissimilar to that of the original Web. 'Early in the Web's development, detractors pointed out that it [the Web] could never be a well-organised library; without a central database and tree structure, one would never be sure of finding anything. They were right! But the expressive power of the system made vast amounts of information available, and search engines (which would have seemed quite impractical a decade ago) now produce remarkably complete indices of a lot of the material out there' (Berners-Lee *et al*. 2001).

top

4 Semantic Web: the technologies

To reveal the data that is generally hidden away in HTML files, Tim Berners-Lee *et al*. (2001) relies on five technologies all of which are, to varying degrees, already being implemented on the Web. These technologies are the following:

4.1 Identifiers

To identify items on the Internet, identifiers known as Universal Resource Identifiers (URIs), are used; the most common or familiar of these being the Universal Resource Locator (URL), which is used to identify the address of a Web page. Broken down, a URL consists of a section that identifies the computer and domain on which the page resides, virtual directory information and the actual file name of the page being visited. URIs naturally work in the same way but, in this sense, are used not as the address of a point, but as resource identifiers. Although the syntax for creating URIs is carefully governed by the IETF, the actual control over URIs is decentralized, in that no one person or organization controls who makes them or how they are used, meaning therefore that no permission is needed in order to create an URI. Unfortunately, this brings with it a few immediately apparent problems, such as multiple URIs pointing to the same thing, or similar URIs pointing to different things, etc. But if a concept as large as the Web and the Semantic Web were to function, trade-offs such as these would be necessary evils.

It is important to bear in mind that a URI does not act as a road map that tells the computer how to get a specific file (even though this may be one of the functions it performs, as in an URL). It is instead merely a name for a resource. 'This resource may or may not be accessible over the Internet. The URI may or may not provide a way for your computer to get more information about that resource' (Swartz n.d.) Effectively, even though a URI may perform other duties, its main function is to identify an Internet resource.

4.2 Documents

4.2.1 XML – eXtensible markup language

XML was designed as a simple way to store or send documents across the Web, which allows a developer to add meaning to the data being stored or transmitted. This functionality is made available by allowing a developer to create his or her own meaningful tags that contain data. When the XML file is then interpreted, a computer application can parse the tags and perform certain functions on that data as determined by the content and attributes of the tag, which encloses it.

Furthermore, XML allows for namespace declarations within each tag to hold URI information, thereby ensuring that name tags created by one person do not conflict with those created by another person and, moreover, making it the perfect mechanism for the Semantic Web. A namespace, even though it sounds very techno-speak, is nothing more than a way of identifying a part of the Web from which meaning for the tag names is derived.

4.2.2 XHTML – eXtensible or well-formed HTML

XML is a strictly typed mark-up language in that it is case sensitive and strict rules apply to the format it takes in order for it to constitute well-formed XML. For every opening XML tag there needs to be a closing tag. Moreover, owing to the parent-child relationships of the nodes within XML, nested children tags need to be entirely encapsulated within the opening and closing tags of their parent nodes. In other words, children tags within a parent node need to close within the parent node in the order that they were opened.

XHTML is merely applying these same rules of well-formed XML to traditional HTML pages, thereby ensuring that the page is not only HTML and Web-browsable but also well-formed XML that can be used by other applications as if it were XML – which, in essence, it now is. Furthermore, the class attribute of HTML typically used for the application of cascading style sheets can now be used to infer semantic information regarding the enclosed text of the tags.

4.2.3 XSLT – eXtensible stylesheet language transformations

XSLT is a mark-up language typically applied to an XML file, which transforms the richer information residing within that file into a format that an application requires. XSLT conforms to all the rules of XML and also contains a number of specialized tags that allow an application to perform a wide variety of functions on the XML data, such as conditional statements like <XSL:IF>, <XSL:WHEN> and <XSL:OTHERWISE> and value selection tags like <XSL:VALUE-OF>, as well as non linear processing tags such as <XSL:TEMPLATE>.

The combination of XSLT, XML and XHTML creates a powerful environment to sending and interpreting Semantically rich data, and providing mechanisms for enabling an application to better understand the information that it is receiving.

4.2.4 Statements

The main cornerstone of Berners-Lee's vision of creating an intelligent and meaningful Web lies with a technology known as the resource description framework (RDF). RDFs use a very simple data model consisting of URI triples; in other words, a combination of three URIs in a specific order, each representing identifiers to the subject, predicate and object of the data being described. The subject URI will usually constitute an identifier representing the origin of the RDF or where the document came from while the object URI acts as either the actual data, a pointer to the actual data or an identifier of the actual data. The object URI need not take the form of an URI and can be a literal string. The predicate URI relates the subject URI to the object URI.

For example:
<http://gregk.com>
<http://personalTaste.com/likes_dislikes/reallyLikes>
<http://SABmiller.co.za/beers/MillersGenuineDraught>

This triple clearly illustrates the subject-predicate-object model, or the Semantic Web, and indicates that Greg Kuck (subject), really likes (predicate) the beer Millers Genuine Draught as brewed by SAB Miller (object).

The one thing that should leap out immediately is that it is now possible for 'anything to say anything about anything' (Swartz n.d.). Herein lies the power behind RDF statements: because RDF is a suitable format to publish database information to the Web, other applications can now utilize or repurpose that semanticallyrich information for their own needs.

Although XML is typically used to serialize RDF data, known as RDF/XML, other Web technologies such as SOAP can just as easily be used for RDF models – perhaps this will be called RDF/SOAP! RDF. Schemas differ quite extensively from XML schemas and DTDs in that, instead of defining the permissible syntax that may reside within a tag, they define classes, properties and their interrelation and operate at a data model level instead of a syntax level.

It is important to note that XML, XHTML and XSLT together can perform a very similar function to that of RDF statements, in that semantic content information can be imbedded within the tags. This semantic information can then be parsed by the receiving Web application, allowing it to infer a specific meaning to the content. The problem with this lies in the fact that each Web developer can create his or her own proprietary Semantic Web applications. This meaning, however, will not be available to other Web applications unless it is directly communicated to other developers. Because there is little to no control over how these semantic inferences are created, there can be no uniformity and hence the already chaotic Internet can become more of a tangled mess than the World-Wide Web.

4.2.5 Ontologies

Because two different databases may use completely different identifiers to identify the same concept, such as lastname and surname, a program wanting to compare these two concepts needs to know that these two terms are being used to mean the same thing. To do this, an application needs to have a method of discovering such common meanings for whatever databases it queries.

This method of discovery is made available through what are known as 'ontologies'. An ontology, in this sense, refers to a document or file that 'formally defines the relations among terms' (Berners-Lee *et al*. 2001). The typical Web ontology consists of both a taxonomy and a set of inference rules. The taxonomy defines all the classes of objects and any relationships between them, for example, 'an address may be defined as a type of location and city codes may be defined to only apply to locations and so on' (Berners-Lee *et al*. 2001). The use of classes, subclasses and relations are very powerful tools to use over the Web, because they allow developers to express large numbers of relations among different entities by assigning properties to classes and allowing subclasses to inherit these properties.

The inference rules allow an application to make decisions based on the classes supplied without needing to actually understand any of the information provided. For example, an ontology may express the rule that 'if a city code is associated with a state code, and an address uses that city code, then that address code has the associated state code' (Berners-Lee *et al*. 2001). The receiving application can then infer that, if a particular city code is provided, that address must be in a particular province or state. Effectively, all that ontologies allow an application to do is manipulate the information provided according to predetermined rules and come to a logical conclusion about that data in the format that it requires.

Furthermore, ontologies can be used to perform a variety of different functions other than simple deductions. Because more information is presented about a concept, they can act to improve the accuracy of search engine requests and allow applications to perform a wide variety of tasks autonomously, as well as tackle complicated questions that current search engines are ill equipped to answer.

4.2.6 Agents

The final key to Tim Berners-Lee's vision of a Semantic Web lies with agents. These agents are the actual software applications that collect content from all over the Web, process the information and exchange the results with other software agents. These agents will provide the backbone to the Semantic Web, in that they will be able to exchange data with other agents even though the data is not specifically designed for the particular agent, eventually promoting the type of synergy that the entire Web community has been looking for.

Furthermore, these software agents are not only responsible for moving information backwards and forwards but also for exchanging digital signatures and proofs. Digital signatures are encrypted blocks of code that verify that the information being transmitted comes from a trusted source and through the use of CRC checks ensure that the data have not been tampered with, while proofs involve verifying that the data being transmitted are valid and true. To do this, the software agent can perform checks based on the RDF's triples and inference rules to ensure that the data it has received are accurate.

top

5 Issues and possible problems

To illustrate potential issues with the proposed Semantic Web, it is necessary to return to the three basic perspectives on what it is expected to accomplish, namely, (a) a universal library; (b) the backdrop for the work of computational agents; and (c) a method for federating knowledge and databases to perform certain anticipated tasks. Dealing with each of these perspectives individually, it is possible, according to Marshall and Shipman (n.d), to evaluate them and determine the possible outcomes and plausibility of the Semantic Web in achieving its aims. The first of these perspectives, that of taming the Web or generating a universal Web library, has pretty much become obsolete in the realm of the Semantic Web, because entities such as Google and AltaVista managed to create advanced Web site indexing and retrieval mechanisms. The second and third perspectives are however very much still part of the focus of Semantic Web activity groups and the W3C.

One of the biggest problems arising out of creating a type of knowledge-navigator lies in the fact that Web content has to cater to two distinct needs: those of the human reader and those of the machine reader. Certain human-oriented concepts, particularly abstract ones (e.g. love, hate and jealousy), are almost impossible to express in machine-readable terms. Furthermore, concepts that apply in one situation are often not as applicable in other situations; for example, a person might trust a Web site to deliver a particular book within a specified timeframe in a good condition, but not necessarily trust the views expressed based on taste and personal judgement of that book (Marshall and Shipman n.d).

These problems are, however, nothing new. Advocators of artificial intelligence have been struggling with the problems of acquiring, representing and using knowledge for over 50 years, with implemented solutions being created to understand specialized problems. The very nature of knowledge means that it is not possible to arrive at any one representation of a concept that applies equally to all circumstances, therefore creating a problem of unending definitions and contexts, as well as the very real possibility of conflicting representations of that knowledge. The implications of this on the concept of a Semantic Web are tremendous, especially with regars to development and processing efficiency. Included in the overheads is the time it takes for a Web author to learn, not only how best to represent the knowledge, but also the syntax, semantics, abstraction methods, etc. that such a representation must conform to, and the time it takes for a single resource to allocate and parse any extraneous RDF representations.

Another issue that is raised by the representation of knowledge lies in the fact that knowledge is constantly evolving. The context of a particular piece of information often changes over time, sometimes this evolution may be relatively straightforward, but

occasionally it will require the complete revision of entire concepts and the their interrelations. A good example is raised by Marshall and Shipman:
'Consider the addition of the microwave oven to the class of ovens in the 1970s. A microwave oven serves the same purpose (to heat things), but uses such different methods that the concept hierarchy for ovens will most likely have to change, resulting in the creation of additional abstract classes to express these similarities and differences' (Marshall and Shipman n.d.)

Not only does knowledge evolve, but there are also certain forms of knowledge that are tacit and, as such, are difficult to express, l*et al*one represent. This means that while the Semantic Web will have little to no problem representing physical concepts, such as products and services, abstract materials and concepts will prove to be another matter altogether. Furthermore, to minimize overheads, it will be necessary to know what knowledge should be enumerated and what constitutes fluff.

The last perspective, that of a federated knowledge or database, requires that all the components that are developed have some knowledge of one another and demands that, at the very least, these components are able to negotiate on the information that will be exchanged, what data are represented and how they will be made available. While not a problem *per se*, this does require some form of standardization and communication in how, where and why the data are shared, which, as we all know, has already presented problems in the short history of the Internet – one need look no further than the way that various browsers implement the W3C's javascript standards or the large differences between Microsoft's implementation of SQL and other vendors to see how far corporations are willing to conform to these standards.

Outside of these perspectives, the concept of a Semantic Web also raises some less theoretical and more pragmatic issues. The first of these revolves around the use of metadata. Tim Berners-Lee's vision is largely based on the decentralized use of metadata in order to create data that are machine-readable. Unfortunately, past Web experience has shown that, without some form of control over the use of these metatags, it becomes very difficult to determine the validity and accuracy of their content. This is even more so for a machine, as it possesses no reasoning power and can therefore only base decisions on the actual content. Furthermore, the syntax specified for use over the Semantic Web is rather complex, and as XML has already revealed, if not correctly formatted, will lead to all sorts of problems. What does an automated application do when it encounters syntax that it cannot parse? Does it ignore it, or do the software agent developers need to build super-parsers that will verify, correct and interpret loosely formed code? On what basis will a machine make these decisions? Will we need an RDF and inference rules to describe another RDF and its inference rules, and where do we draw the line?

top

6 Conclusion

The potential of the Semantic Web to solve real-world problems in inter-device communication, finding, sorting and classifying information, is tremendous. Unfortunately, to achieve this it is necessary to understand that its power is more applicable to certain types of information than it is to others. In this respect, it is doubtful that it will become the great panacea that will rid the Web of all its ills and bring its true potential to the fore. Even in situations where the application of semantic content is applicable, a great need exists for the concept to be narrowed down, well standardized and better defined so that developers and Web authors are in a position to apply it.

Tim Berners-Lee's vision of a machine-readable library of information accessible to both humans and machines, while expansive in scope, is unfortunately limited in applicability due, predominantly, to the nature and changeability of knowledge. This is not to say that it is not a viable solution, merely that its applicability will more than likely not be able to encompass the entire Web; instead it will find its place in specific niche markets or as a means of exchanging information within specific industries. It is doubtful that it will ever make the transition into mainstream information acquisition, dissemination and use.

top

7 References

Berners-Lee, T., Hendler, J. and Lissila, O. 2001. *The Semantic Web*. [Online]. Available WWW: http://sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21.

Berners-Lee, T., Hendler, J. and Miller, E. 2002. *Integrating applications on the Semantic Web*. [Online]. WWW: http://www.w3.org/2002/07/swint.

Frauenfelder, M. 2001. *A smarter Web*. [Online]. Available WWW: http://www.technologyreview.com/articles/frauenfelder1101.asp.

Marshall, C.C. and Shipman, F.M. n.d. *Which Semantic Web*. [Online]. Available WWW: http://www.csdl.tamu.edu/~marshall/ht03-sw-4.pdf.

Swartz, A. n.d. *The Semantic Web in breadth*. [Online]. Available WWW: http://logicerror.com/SemanticWeb-long.

Disclaimer

Articles published in SAJIM are the opinions of the authors and do not necessarily reflect the opinion of the Editor, Board, Publisher, Webmaster or the Rand Afrikaans University. The user hereby waives any claim he/she/they may have or acquire against the publisher, its suppliers, licensees and sub licensees and indemnifies all said persons from any claims, lawsuits, proceedings, costs, special, incidental, consequential or indirect damages, including damages for loss of profits, loss of business or downtime arising out of or relating to the user’s use of the Website.

ISSN 1560-683X

Published by InterWord Communications for the Centre for Research in Web-based Applications,
Rand Afrikaans University