Four Steps Towards the Widespread Adoption of a Semantic Web

Brian McBride

Hewlett Packard Laboratories, Filton Road, Bristol, BS12 6QZ

Abstract. This paper suggests four steps towards the realization of a semantic web. Promotion of the idea should be based on practical application. There is need for the immediate development of practical demonstration applications. Simplicity and tolerance of error should be prime targets of research and development. An Open Source project to develop and populate a framework of tools and applications should be started.

1 Introduction

The profile of the semantic web has increased over the past year. Higher visibility attracts both supporters and critics. Expectations have been set by articles such as that in the Scientific American[1]. Failure to deliver that vision or replace it with something similarly attractive will lead to a perception of failure. This paper describes four steps that might be taken to facilitate delivery of a semantic web. It is not suggested that these are sufficient, or even necessary. It is suggested that they are helpful.

2 Promote Practical Applications

Google classified last year's SWWS workshop under Computers> Artificial Intelligence . This year it is Science > Math > Logic and Foundations. Semanticweb.org is classified as Computers> Artificial Intelligence> Knowledge Representation. The perception that the semantic web is concerned with artificial intelligence is not helpful to its widespread adoption in the IT industry.

This perception is understandable. The semantic web is often presented as the technology that will achieve marvelous things, exemplified perhaps by a scene from "Star Cops", a 1980's television program:

> The workaholic hero Nathan is suspended from duty and decides to take a holiday. He calls across the room to his PDA and instructs it to get him two tickets for the Milan opera. A short time later the PDA replies that Nathan is booked on the next shuttle back to earth, on a fast train to Milan and into his favourite hotel for three nights. He has to two tickets to La Traviata on Saturday night.

Much of the excitement and motivational power of such scenarios come from the handling of speech, natural language processing, general purpose problem solving,

I. Horrocks and J. Hendler (Eds.): ISWC 2002, LNCS 2342, pp. 419–422, 2002.

scheduling, common sense reasoning and other features commonly associated with intelligence. These are not what the semantic web is about.

The semantic web is about creating an infrastructure in which information, from a variety of sources, can be integrated on demand to achieve some task. The semantic web provides mechanisms that enable access to and integration of information. Data is not the same thing as information. Data becomes information when it can be understood. Data with associated metadata to describe its meaning, becomes information that can be processed by a computer.

An intelligent agent will find access to such an infrastructure useful in accomplishing its goals. Until such agents are available however, this infrastructure will be very useful to everyday implementers of information systems using conventional programming techniques. Information integration is a key problem for many IT managers. Technology that will ease that problem is valuable.

Consider a simple example, a servlet, to display the members of a mailing list. The servlet first sends a query to the mailing list server, and receives in response an RDF graph that lists each member of the mailing list and their email address. It then sends the graph to an LDAP server with a request to add any information it has about any resources in the graph. The LDAP server, knowing that an email address uniquely identifies a person, can fill in the names, locations, URL's of home pages, link to the corporate organization chart etc and return the augmented graph to the servlet, which transforms it into a pleasant human readable representation. All it took to get the information was two simple RDF queries to the (appropriately augmented) servers.

The integration of information is possible because there is a standard syntax for the exchange of information, a global naming scheme that the systems share, a shared means of describing the characteristics of information (email address is a unique property) and a schema so that the servlet developer knows how to turn the information received into a presentable form. This ability to integrate disparate information sources is a clear commercial justification for the development of a semantic web.

3 Develop Applications Now

The specification of RDF, possibly the bottom layer of the semantic web, has been available since 1999. There are widespread applications that use it; Mozilla; RedHat's Update Agent, RSS and others. What is disappointing, indeed worrying, about these applications, is they do not demonstrate the advantages claimed for the semantic web, or at least not the advantages claimed by this author.

The first web, had not only freely available tools which folks could use to learn and experiment, it also had many examples of cool things to do, demonstrating its use and usefulness. These examples, as well as encouraging others to copy them, also stimulated innovation in the use of the technology. Innovation in the use of the web has been far greater than in its underlying technology. HTML and HTTP have not changed that much from their original form, but a considerable fraction of the planet has set about doing something original with them. Much of this innovation was in content, but it did lead also to demand for improvements in the technology.

The semantic web community should now be doing the same. Enough of the technology is available to create applications that demonstrate its value. Developing such applications will persuade the skeptical, stimulate innovation and speak most clearly of the value the semantic web can bring.

4 Simple and Tolerant of Error

HTML is simple, and was even simpler when the web was young. Web browsers are tolerant of error. They will try to do something sensible with whatever HTML they encounter. As a result, users in the early days of the web could learn to create web pages by cribbing from existing pages; it was easy to guess what the various bits of markup meant. The tolerance of the browsers ensured that it didn't matter much if the author of the content made some mistakes, there was a good chance the browser would do something acceptable with it. Inexpert users were able to be successful quickly, and thrilled with that success, were motivated to learn more and develop more sophisticated content.

Some argue that this tolerance was a mistake; that it caused many problems that were hard to resolve. But those who argue this are usually computer scientists for whom it probably did cause many problems that were hard to resolve. Users on the other hand don't complain that they did not get enough error messages. The simpler and more tolerant of error a system is, the more rapid and more widespread its adoption is likely to be.

This author recently wrote an RDF schema for a W3C specification. This simple task turned out to be harder than expected, even for someone with a good understanding of RDF and RDF schema. The lesson to be learned is that good schemas are hard to write; even for experts; even when the underlying data model is already done.

Unless we are willing to accept that schemas will only be written by a select few, well trained and expert in the art, an acceptance that contradicts the essence of the principles of the web and immediately places limits on its rate of growth, we have to accept that flawed information models will be prevalent in the semantic web. Models will use literals where they should use resources, will use instances where they should use classes, will confuse numbers and numerals and contain many other flaws.

Delivering a semantic web requires the technology be simple to reduce the possibility of error, and tolerant where there is error.

5 Open Source

The web has always been blessed by the availability of free, high quality software. The Mosaic browser, the CERN server, the Netscape browser/html editor, Internet Explorer, the Apache server and Mozilla have all meant that it was easy and inexpensive for individuals and organizations to first experiment and then to create production quality content and services. Cost was not a barrier to adoption by providers or users.

Commercial organizations like to dominate an infrastructure. The economic power this realizes has been apparent for over a century. Dominance of an infrastructure by a single organization can lead to the stifling of innovation. What would the web be like if its evolution had been controlled by a single corporation?

The Open Source community is enjoying some success in resisting the commercial domination of infrastructure. Where OS/2 failed to challenge the dominance of the Windows operating system, linux is succeeding. Apache is the most common web server on the internet. This should not be surprising. Open Source has a philosophy that combines collaboration and excellence. It is well suited to the development of infrastructure where it is in the best interests of all to cooperate and avoid dominance by a few commercial interests.

The semantic web community, though growing, is small. It cannot afford to use its limited development resources in wasteful duplication. How many RDF parsers does the world really need? There have been early experiments in the design of the lower layers of the technology. Enough has been learned to suggest that we have reached the time to create and populate an open source framework of tools for the semantic web. This "Apache" for the Semantic Web could be an independent organization, or could be born within the Apache organization itself as a peer of projects like Jakarta. Such an initiative would benefit all. The cost barrier to adoption of semantic web technology would be reduced facilitating early experiment and innovation in application development. The research community would also benefit, having a framework of tools to facilitate their research and into which to deliver the fruits of their labours. All can feel secure that no commercial organization can hope to attain a position of control of the infrastructure.

6 Conclusion

If a semantic web is to be successful and achieve wide adoption, its supporters should:

- emphasize practical applications
- software must be simple and tolerant of error
- develop applications now
- Develop a common open source infrastructure of high quality References

References

[1] Berners-Lee, T., Hendler, J., Lassila, O., : The Semantic Web. Scientific American, (2001)