Spinning the Semantic Web

Sasikumar Mukundan

The author

Sasikumar Mukundan is a Senior Research Scientist at the Centre for Development of Advanced Computing (formerly NCST), Mumbai, India.

Keywords

Worldwide web, Communications, Technology led strategy

Abstract

From the quiet new born days of early 1990s, the World Wide Web has had an exponential growth in the last decade or so. From the original goal of sharing research resources, Web today portrays a virtual world spanning from research to entertainment and e-commerce. This growth has necessitated substantial changes in the Web model. From the purely syntactic and relatively static framework of HTML, we have moved through DHTML and XML incorporating dynamicity and extensibility, and are now *en route* semantic frameworks starting with RDF. These allow Web documents to be comprehensible to machines (and not just to humans) allowing software agents to access and process such information on the Web. This leads us to semantic Web, and thus to a generation of Web applications based on Web services, adaptive content delivery, etc. *Spinning the Semantic Web* is based on papers presented in a seminar in Germany in 2000, and sketches the vario elements of semantic Web, the issues in realising it as well as some visions of the future. The stimulating forward to the book by Tim Berners-Lee, recently Knighted and widely regarded as the father of the Web, portrays his vision of semantic Web. The chapters explore specific issues such as ontologies, schema languages, annotations, applications, etc. The chapters are largely unorganised and presented without any cross-linking and most chapters use a fair amount of domain jargon. The book will be of value to those seriously interested in the field.

Electronic access

The Emerald Research Register for this journal is available at
www.emeraldinsight.com/researchregister

The current issue and full text archive of this journal is available at
www.emeraldinsight.com/1074-8121.htm

On the Horizon
Volume 12 · Number 2 · 2004 · pp. 74-78
© Emerald Group Publishing Limited · ISSN 1074-8121
DOI 10.1108/10748120410544153

This article is a review of:
Spinning the Semantic Web
Edited by D. Fensel, J. Hendler, H. Lieberman
and W. Wahlster
MIT Press
Cambridge, MA
2003
477 pp.

From the humble beginnings in the early 1990s, the World Wide Web has come a long way, well exceeding all imaginations. Envisaged by Tim Berners-Lee, regarded as the father of the Web, as a mechanism to share documents among researchers, the Web took birth at CERN - European Particle Physics Lab. Today, it has become a necessary component for not only researchers and developers, but almost everyone from teachers and doctors to farmers and students. From being an information sharing mechanism, the Web has grown to be a virtual world covering everything from shopping and entertainment to learning and collaborative research.

The simple framework of HTTP protocol for communication and a set of document annotation collectively called HTML (Hyper Text Markup Language), introduced in the early days of Web, still continue to form the core of the Web. While this fitted the original requirement of having a simple, easy to learn framework for non-IT users to work with the Web, these frameworks have today become totally inadequate. The Web has grown to billions of pages containing huge amount of information on essentially anything under the Sun. The task of finding the right information from this mind-boggling information space is daunting every user. Search engines have come a long way in using innovative ideas to make sense of this vastness and continuously compete with each other to offer usable results within reasonable response time. However, compared to the (ever increasing!) expectations of the various categories of users, the gap is still quite high.

Problems with the Web

The Web evolved as a largely syntactic framework, consisting of documents and resources meant for human consumption. Most users use HTML tags only as a formatting device, and not to identify types of content. It becomes nearly impossible to identify if an occurrence of a term such as "semantic Web" in a document is a casual reference or indicative of the content of the document. One needs strong context analysis technologies to know in what context and with what intention the author of the document uses the term. Further complicating this scenario are issues of synonyms, non-standard phrases, etc. Blind keyword searches have already reached their limits.

When you consider applications such as e-commerce and Web services, the problem is more severe. Every site offering a service would use its own convention and language to describe its services and conditions. A user looking for a service, need to be able to analyse these and identify a "good" offer, if not the best one. This is a time consuming task to be performed manually; one would like intelligent agents to locate relevant sites, analyse their offers, and recommend potential solutions. Given that most material on Web sites today are aimed at usage by human beings, the level of intelligence required for a software program (such as agents) to make a comparative analysis is very high. For example, one site may mention the price inclusive of delivery and taxes, whereas another may exclude some of these. There may be conditions (e.g. not available in country X) imposed on some of the sources, which may severely impact comparisons.

As we move toward the next generation of Web technology, we are moving (quoting from the book) "from keyword matching to intelligent search, from information retrieval to query answering, customised views of documents, widespread and easy exchange of documents", etc. The information we require may be spread over more than one document relating to different topics, perhaps. We need mechanisms which can pool these documents and give us a direct answer, rather than let us read the documents independently and figure out the answer ourselves. This requires the information to be machine processable, if not machine understandable.

One can find more examples of such problems in the book under review as well as other literature on the topic. The move towards semantic Web attempts to address these problems by encoding semantic information in the content.

Towards semantic Web

What does it mean to add semantic information to the content? Today's Web pages contain little machine understandable information about the context or specification of the target community. Even for a human user, he has to spend time reading through parts of the document to understand the information. Thus, when searching on a keyword, any occurrence of the word is considered same as any other. Similarly a term occurring in any document is considered equivalent, though they may refer to different concepts, possibly in different domains. One major step forward would be for pages to refer to ontologies, they relate to. Ontology defines a set of terms and their relation to one another. Course administration ontologies may define "entrance fee" as a kind of fee, and map it to "admission fee" used in another ontology. Such kinds of information enables a search engine to retrieve all relevant information and relate them to each other.

We move from a syntactic Web linked through manually inserted URLs, to a semantic Web, where the pages are related by talking about the same topic, offering similar services. As in real life, this enables users to pick relevant information from different sites, integrate them to meet one's requirement. Independent components from the Web are increasingly deployed together to solve a problem. The integration and coordination of these components to solve a given problem will be done more by software agents than humans. From the static pages, these components have been growing in complexity and sophistication, to relationships, processes, etc.

In order to enable such kinds of information to be shared not only for human users, but also for software systems, we need conventions and frameworks. For static data, this was a relatively easy task – only format information needed to be represented. For facts, we resort to frameworks such as XML. They allow users to define their own tags, thereby extending the simple HTML framework, and use them to annotate information such as price, name, etc. For expressing relations, we need more powerful frameworks along the lines of resource description framework (RDF). RDF provides a simple attribute-object-value mechanism to specify relations. It does not pre-define any attributes, nor does it have any reasoning mechanism exploiting notions such as hierarchy of concepts and inter-relationships of relations. RDF schema (RDFS) is proposed as an extension to RDF to provide these to some extent. For problem solving methods, specifying what knowledge to use and how to solve problems, we need still different frameworks such as UPML. While HTML, XML and to some extent RDF are being accepted as standards, the higher levels are yet to see any convergence towards a specific framework.

Ontologies are also topics of active research and the book devotes a fair amount of space to discussion of ontologies in the context of semantic Web. The discussion includes the major ontology frameworks ontology inference layer (OIL) and DAML-ONT.

The contents of the book

This book *Spinning the Semantic Web* is based on a seminar held in Germany in March 2000, and hence are in the form of fairly long papers by various research groups around the world. The papers cover a fair amount of breadth and depth on the various aspects of the semantic Web notion.

The book starts with a Foreword by the father of the Web, Tim Berners-Lee. Rather than have this based on the current status of the semantic Web, the authors have gone for reprinting an earlier (December 1997) article by Lee. While this helps to see the thinking from those early days, it mismatches with the rest of the book in wavelength. However, his ideas of an "oh yeah" button and "intercreativity" are interesting. The "oh yeah" button is introduced in the context of determining the authenticity of a particular Web content, wherein the system would look around for some justification to trust/discard the information mentioned in that page. The article also shows that we have not made much progress in the intervening years, in terms of tools and technology for addressing the major problems that he had paused at that time.

The book is divided into three parts:
(1) Languages and ontologies.
(2) Knowledge support.
(3) Dynamic aspect.

The Foreword and Chapter 1 are excluded from this classification, and precede them. Chapter 1, titled "Introduction", introduces the book briefly, explains the drawbacks of the current Web framework, outlines the basic ideas of semantic Web, identifies the major components for realising it, and briefly examines some of the potential applications.

The chapters in part I range from 2 to 6. Chapter 2, "SHOE: a blueprint for the semantic Web", discusses a language named SHOE as a blueprint proposed for semantic Web. SHOE follows an XML type of notation, introducing a number of tags for representing specific kinds of information. It allows pages to identify the ontology that it uses, which provides a way to identify significance of specific terms used and their relation to other terms. Existing ontologies can be extended by adding/redefining terms. The language primitives allow relations and entities to be specified. Limited inference capabilities are also supported by the language. The language specification and implementation are available for free as open source.

DAML is a US-funded initiative for developing a mark up language: DARPA Agent Markup Language. It has an ontology component, known as DAML-ONT. Chapter 3, "DAML-ONT: an ontology language for the semantic Web", focusses on this ontology framework. The chapter describes briefly the facilities to define classes (denoting entities), and for specifying properties such as cardinality, domain, range, etc. A good part of the paper is devoted to discussion of axiomatic semantics for the language by showing equivalent first order logic statements.

Chapter 4, "Ontologies and schema languages on the Web", takes a more general view of ontology languages, briefly looking at their relation to schema languages. The paper looks at RDF Schema and XML schema, and compares them to OIL taken here as an exemplar ontology language.

Achieving a user-goal requires, apart from static factual knowledge, knowledge on how to use such knowledge, namely, the problem solving methods. Problem solving methods are also reusable components and hence a candidate for standardisation in representation. Chapter 5, "UPML: the language and tool support for making the semantic Web alive", is concerned with this issue, and discusses briefly a markup language UPML – unified problem solving method development language. Such frameworks can be combined with suitable ontologies, by what are called knowledge brokers, to realise specific knowledge systems.

Chapter 6, "Ontologies come of age", introduces the term ontology from the fundamentals – the dictionary and the history of the term. This sounds out of place here, given that the previous four chapters have been talking about ontology and much beyond! Ontology is classified into: simple and structured. Simple ontologies are nothing more than machine processable glossaries. Structured ontologies represent attributes and interdependences among the terms as in the case of frames in AI parlance. Such ontologies would support consistency checking, completeness and limited forms of inferencing. The chapter also outlines possible uses for ontology, frameworks for developing ontologies, and issues in its development. Compared to earlier chapters, this chapter is much less technical.

Chapter 7, "Sesame: an architecture for storing and querying RDF data and schema information", opens part II of the book focussing on knowledge support. A system called Sesame is introduced in this chapter, for efficient storage and querying of large RDF data stores. With examples, it is suggested that data in higher level frameworks such as RDF framework should have semantic querying facilities. Syntactic querying frameworks such as Xquery views the data as pure XML and is nothing more than a pattern matcher. Even relational querying catering to RDF fail to exploit hierarchies, etc. The chapter proposes a language

RQL addressing these issues and providing powerful querying support on top of RDF Schema. Sesame is the core storage handler and engine for RQL.

Chapter 8, "Enabling task centered knowledge support through semantic markup", takes the task of information retrieval a step further to consider the task the user is currently involved in. The chapter focuses on the domain of aerospace customer support building a system around a query language called SiLRI. I found this chapter particularly poor with no clear focus or plan.

Knowledge bases are not islands; they need to work with other components including other knowledge bases. Just having a shared ontology is not enough. Chapter 9, "Knowledge mobility: semantics for the Web as a white knight for knowlege based systems", discusses these issues in the context of a military planning application. It proposes a resilient hyper knowledge base model, TRELLIS, which supports semantic links to various knowledge sources used and rationales for decisions. Much of this chapter seems to have little relevance to the topic of semantic Web.

The issue of using heterogenous knowledge sources is taken a bit further, in Chapter 10, titled "Complex relationships for the semantic Web". A framework called InfoQuilt is presented with an XML-based representation of knowledge about information available from various sources. Given a query, the system makes use of this information to pick relevant information from appropriate sites, combining them in fairly complex ways. Some of the notions are similar to partitioning a database vertically and horizontally across different sites.

Chapter 11, "Semantic portal: the SEAL approach" first describes an ontology building exercise and shares the authors' experience in the process. SEAL is a framework built on top of Ontobroker – the ontology system – with a set of adapter tools so as to present the same information in different formats for use by other software applications and human users. The chapter also discusses their approach to semantic ranking and semantic personalisation. This is another badly structured chapter with no clear message or focus.

Chapter 12 opens the last part of the book titled "Dynamic aspect". The chapter titled "Semantic gadgets: ubiquitous computing meets the semantic Web", sketches the notion of semantic gadgets, in the emerging world where every device from toaster to mobile-phones have embedded computing capability and are fully interconnected through appropriate technologies. Semantic gadgets are devices which can work coherently with other similar devices as a coalition and capable of semantic discovery (i.e. others can find out about you and your capabilities without human help). Ubiquitous computing refers to the spread of computing techniques in embedded form into various devices such as oven, phone, etc. The chapter is essentially confined to describing a possible application scenario in a museum where the user's semantic gadget interacts with a variety of other gadgets from parking-lot-planner to "rush" analyser. The chapter focus is on the need for devices to interact closely with a variety of other devices in a semantic way – not using rigid standard syntacic matches – and the issues involved in effectively doing so.

Chapter 13, "Static and dynamic semantics of the Web", introduces what is called dynamic semantics of the Web produced by rapidly changing content, and proposes a language called Glue as an extension of HTML/XML to encode procedural semantics. Very little information is given about the language, despite being the focus of the chapter. Particularly lacking is a good example showing its usage and strengths. The chapter prefers, instead, to describe the major features and provide a comparison to Java and Javascript. The comparison seems to suggest Glue as a marginal enhancement over Javascript!

When a given Web page is to be sent to a normal computer window as well as a mobile phone screen, a fair amount of adaptation is required to optimise the information presented on the small screen space available in a mobile phone handset. In some cases, this can be done with some static configuration information (e.g. suppress all images). Using such a situation as a driving example, Chapter 14, "Semantic annotation for Web content adaptation", proposes the use of semantic annotations. Since it is not possible to persuade all site authors to provide the required annotations, a general mechanism for external annotations is introduced here. A browser picks up the original page as well as any annotations on it (perhaps from another site), and delivers an adapted version of the page driven by the annotations. A language for specifying annotations – including specification of importance, role, etc of the various parts of the page – is also described briefly.

Chapter 15, "Task achieving agents on the World Wide Web", the last chapter of the book, discusses integrating planning technologies into the notion of semantic Web. An ontology model for describing activities in a planning application is proposed, and three application domains are described built on an adapted version of O-Plan - an AI-based planning system developed by the authors' group earlier.

Looking back

As one goes through the book, one cannot fail to see the striking similarity of the early work in the field of artificial intelligence (specifically knowledge representation) which is getting reinvented here. They are now packed in new bottles of OIL, RDF, etc. For example, DAML-ONT primitives have a high degree of overlap with the primitives of early frame based knowledge representation formalisms. But the description gives no such feeling – particularly in Part I – let alone acknowledge them! Chapters in Part II do refer to the frame representations as a source model.

Formal theories lose their momentum as they jump into the difficulties in guaranteeing soundness and completeness of inferences generated when full first order logic is used. This is true even when there is no uncertainty in the system, all knowledge are stored at one place, and consistency can be assumed. In a World Wide Web scenario, none of these assumptions are true. How much sense does it make to go for a logic based representation as the backbone of the Web of tomorrow? It puzzles me.. But there is no answer in the book.

Taking this point further is the lack of a critical look in the book. All papers have taken for granted the success of the semantic Web model as realisable through RDF, XML, and related technologies. None of the papers deal with semantic Web as an open topic, evaluating the current approaches objectively.

Given that the book is a collection of papers, not integrated with one another, a good overview of the contents of the book and how the different chapters relate to each other would have been useful. The existing introduction to the book is too short to do justice to this requirement. So, it is left to the readers to put the pieces together and evolve the outer framework, to whatever extent is possible.

Similarly, given the varied and stand-alone nature of the papers, a chapter at the end consolidating the various views, integrating them and positioning them in the overall picture of semantic Web would have been valuable. This is also unfortunately missing. The book all the way reminds the reader of its scattered-collection nature, and hence is nothing more than a workshop proceedings.

Most of the papers in the book, makes heavy use of jargon, making it nearly impossible for someone unfamiliar with the literature in this particular field to understand much of what is going on. When frameworks such as DAML-ONT, SHOE, etc are described, most aspects are introduced by relating them to other systems of a related nature. While this is useful for someone familiar with them, the average reader is quite lost as a result. At many places, this dependency could have been avoided with a little more effort.

The basis for organisation of the various chapters are not clear. Though most of them are talking about related ideas, and even refer to the systems introduced in other papers, no paper explicitly refers to the description in other papers, nor adjusts its presentation taking these into account. The result is discontinuity and overlap, as mentioned in the previous section. Even the splitting of the papers into the three parts make little sense – the book does not try to explain the parts nor the rationale for the grouping!

Even the editing and integration leaves much to be desired. Many of the papers contain typos and English errors. The index contains words such as payment, mapping, maintenance information, etc which are unlikely candidates for an index search. The references are given at the end of each chapter, and many are repeated from chapter to chapter since the chapter coverages overlap. A consolidated, and preferably annotated, set of references at the end, would be useful.

When I picked up the book for review, I expected to understand what makes a semantic Web, how the different components relate to one another, the strengths and weakness of the current models, the major issues etc. The book has been quite disappointing in this regard. It is recommended only for someone already well familiar with the field of semantic Web, including the various frameworks and techniques in place.