Web Science: An interdisciplinary approach to understanding the World Wide Web

James Hendler
RPI
Dept of Computer Science
Troy, NY 12180
hendler@cs.rpi.edu

Nigel Shadbolt and Wendy Hall
University of Southampton
Electronics and Computer Science
Southampton, UK
{nrs,wh}@ecs.soton.ac.uk

Tim Berners-Lee and Daniel Weitzner
MIT
CSAIL
Cambridge, MA
{timbl,djweitzner}@csail.mit.edu

ABSTRACT
Despite the huge success of the World Wide Web as a technology, and the significant amount of computing infrastructure on which it sits, the Web, as an entity remains surprisingly unstudied. In this article, we look at some of the issues that need to be explored to model the Web as a whole, to keep it growing, and to understand its continuing social impact. We argue that a "systems" approach, in the sense of "systems biology" is needed if we are to be able to understand and engineer the future of the Web.

Categories and Subject Descriptors
K.m [MISCELLANEOUS]

General Terms
Algorithms, Performance, Design, Economics, Security, Human Factors, Languages, Legal Aspects.

Keywords
Web Science, World Wide Web

1. INTRODUCTION
Take a look at the "categories and subject descriptors" for this paper. This paper is explicitly about the World Wide Web itself. Yet despite the huge impact that the Web has had on computing, and on the field of Computer Science itself, the best keyword indicator one can find in the ACM taxonomy, the one by which the field of Computer Science organizes many of our research papers and conferences, is "Miscellaneous." Similarly, if you look at Computer Science curricula in most universities you will find "Web design" is taught as a service course and there might be a course on Web scripting languages, but there is not likely to be a course in the curriculum that teaches Web architecture or protocols. It is as if, below the browser, the Web simply doesn't exist. In many Information Schools or Informatics Departments the courses will focus on applications on the Web, or topics such as "Web 2.0," but again, the protocols, architectures and underlying principles of the Web *per se* are rarely included.

Simplifying a bit, part of the reason for this is that networking has long been a part of the systems curricula at many departments, and thus the Internet, defined via the TCP/IP networking protocols, has long been considered an important part of CS work. The Web, despite having its own protocols, algorithms and architectural principles, has been thought of by many people in the computer field as an application running on top of the Net, more than as an entity unto itself.

This is actually quite odd, because the World Wide Web has been the most used and one of the most transformative applications in the history of computing. In academia it has changed how we teach, how we communicate, how we publish and how we do research. For industry it has not only created an entire sector (or arguably, multiple sectors), but has effected the communications and delivery of services across the entire industrial spectrum. In government, it has changed not only the nature of how governments communicate to their populations, but also how those populations communicate and even how we end up choosing the government in those societies where this is done (for example, consider the US presidential debates where candidates for the highest office in the land took questions online and in the form of youTube videos). It is estimated that the size of the human population on Earth is on the order of 10^{10} people where the number of separate Web documents is over 10^{11}.

Of course, computing has made significant contributions to the Web -- our everyday use of the Web is also dependent on fundamental developments in computer science that took place long before the Web was invented. Today's search engines for example, are based on developments in information retrieval that have a legacy going back to the 1960s. The innovations of the 90s (Kleinberg, 1997, Page and Brin, 1998) provide the crucial algorithms underlying modern search, and are fundamental to Web use. New resources such as Hadoop (http://lucene.apache.org/hadoop/) make it possible for students to explore these algorithms, and to experiment with key, large-scale Web-programming practices like MapReduce parallelism (Dean and Ghemawat, 2004) in a way that has not previously been accessible outside of a few top universities.

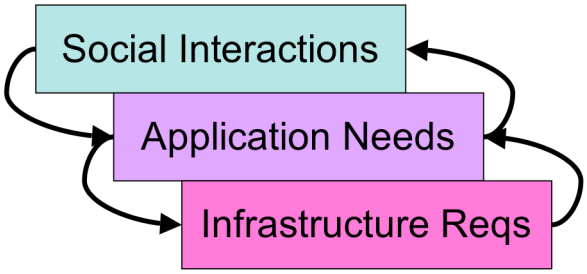


Figure 1 The social interactions enabled by the Web place demands on the Web applications underlying them. These in turn raise requirements on the Web's infrastructure.

Other aspects of the interactions occurring on the Web have been studied in other departments. Of particular note, many of the interesting aspects of the use of the Web, for example, social networking, tagging, data integration, information retrieval, Web ontologies, etc., have become part of a new "social computing" area that is evolving at some of the top Information Schools. These departments offer classes in the general properties of networks and interconnected systems, in the policy and political aspects of computing, and in the economics of computer use. However, in many of these courses, the Web itself is treated as a specific instantiation of more general principals. In other cases, the Web is viewed primarily as a dynamic content mechanism that supports the social interactions between multiple browser users. In short, whether in Computer Science studies or Information School courses, in all too many cases the Web is studied as the delivery vehicle for content, be it technical or social, rather than as an object of study in its own right.

In this article, we present the emerging interdisciplinary field of Web Science (Berners-Lee et al. 06), which does take the Web as its primary object of study. We will show that there is a significant interplay between the social interactions enabled by the Web's design, the scalable and open applications development that is mandated to support these, and the architectural and data requirements of these large scale Web applications (Figure 1). The study of the relationships between these levels is, however, often hampered by the disciplinary boundaries that tend to overly separate the study of the underlying networking from that of the social applications. In this article, we identify some of these relationships and briefly review the status of Web-related research within computing, However, we primarily focus on identifying some emerging, and extremely challenging, problems that we believe researchers from across the computational and information spectrum, in their role of Web scientists, need to explore.

2. WHAT IS "WEB SCIENCE"

"Web Science" is the emerging interdisciplinary field that views the World Wide Web as an important entity to be studied in its own right. Where physical science is commonly regarded as an analytic discipline that aims to find laws that generate or explain observed phenomena; computer science is predominantly (though not exclusively) synthetic, in that formalisms and algorithms are created in order to support particular desired behavior. Web science deliberately seeks to merge these two paradigms. The Web needs to be studied and understood as a phenomenon, but it also needs to be engineered for future growth and capabilities.

At the micro scale, the Web is an infrastructure of artificial languages and protocols; it is a piece of engineering. However, it is the interaction of human beings creating, linking and consuming information that generates the Web's behavior as emergent properties at the macro scale. These macro properties are often surprising and require analytic methods to understand them. Some properties are desirable, and therefore to be engineered in, others are undesirable, and if possible should be engineered out. We also need to keep in mind that the Web's use is part of a wider system of human interaction – the Web has had profound effects on society, with each emerging wave creating both new challenges and new opportunities in making information of different kinds available to wider sectors of the population than ever before.

It may seem that the best way to understand the Web is as a set of protocols that can be easily studied for its properties, with individual applications analyzed for their algorithmic properties. However, the Web wasn't (and still isn't) built using the specify, design, build, and test development cycle that Computer Science has traditionally viewed as the software engineering best practice.

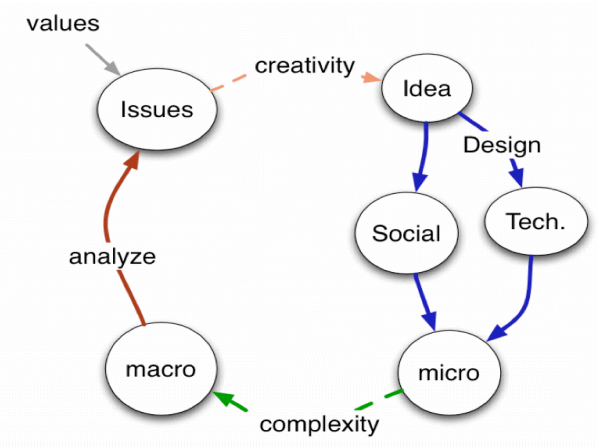


Figure 2: The Web presents new challenges to software engineering and application development.

Figure 2 shows a different way of looking at Web development. A software application is designed, based on an appropriate technology (algorithm, design, etc.) and with an envisioned "social" construct (it is a contradiction in terms to talk about a Web application built for a single user on a single machine). The system is generally tested in a small group or deployed on a limited basis – the "micro" properties of the system are thus tested. In some cases, when more and more people accept the micro system, a "viral" accelerating complexity of use occurs. For example, when MOSAIC was released publicly, the number of users grew by several orders of magnitude in a short time, with over a million downloads in the first year (for more recent examples, consider photo-sharing on Flickr, video uploading such as YouTube, social networking sites like Facebook, etc.).

The "macro" system, that is the use of the micro system by a great number of users interacting with each other in often-unpredicted ways, is far more interesting in and of itself, and generally must be analyzed in very different ways than the micro system. Also, these macro systems can have issues that cannot occur in the micro -- for example, the wide deployment of Mosaic led to a growing need for a way to find relevant materials on the growing Web, and thus search became an important area. In other cases, the large-scale system may have emergent properties not predictable by analyzing the micro, or unpredicted social effects. Dealing with these issues is often what leads to a next generation of technology, and so on. For example, the success of search engines has led to the development of techniques to game the algorithms (an unexpected result) to get better search rank, which has led, in turn, to the development of better search technologies to defeat such gaming.

The essence of understanding what succeeds on the Web, and how to develop better Web applications, is that we have to create new ways to understand how we can design such systems to have the eventual effect we envision. Currently, the best we can do is to

design and build in the micro hoping for the best – but how do we know if we've built in the right elements/functionality to ensure a macro take up? How do we predict what the side effects and emergent properties of the macro will be? Further, as the success or failure of a Web technology may involve aspects of social interactions between users, a topic we return to in Section 4, understanding the Web requires more than a simple analysis of technological issues, but also an understanding of the social dynamic.

Given the breadth of the Web, and its inherently multi-user (social) nature, its science is necessarily interdisciplinary, involving at least mathematics, computer science, artificial intelligence, sociology, psychology, biology and economics. We invite computer scientists to expand the discipline by addressing the challenges brought from the widespread adoption of the web and its profound influence on social structures, political systems, commercial organizations, and educational institutions.

3. BENEATH THE WEB GRAPH

One way to understand the Web, familiar to many in computer science, is as a graph whose nodes are Web pages (defined as static HTML documents) and whose edges are the hypertext links between these nodes. Kleinberg et al (1999) named this the *Web graph* and performed the first analysis thereof. Barabasi and Albert (1999) and Kumar et al (1999) showed that the in-degree of the Web Graph followed a powerlaw distribution and work by Broder et al (2000) showed a similar effect for the outbranching of vertices in this graph. An important result by Dill et al (2001) showed that large samples of the Web, generated by different methods, all had similar properties, which is important as the Web graph grows constantly, reported in 2005 to be on the order of seven million new pages a day (Gulli and Signori, 2005). Various models have been proposed as to how the Web graph grows and which models best capture the evolution of the Web (See Donato et al, 2007) for an analysis of a number of these models and their properties).

As well as analyses of this graph and how it grows, there have been a number of algorithms that exploit various properties of the graph. Kleinberg's (1997) HITS algorithm and Brin and Page's (1998) PageRank assume that the insertion of a hyperlink from one page to another can be taken as a sort of endorsement of the "authority" of the page being linked to, and this assumption led to the development of powerful search engines for finding pages on the Web. While modern search engines use a number of heuristics beyond these page authority calculations, in part because of competitive pressures from those trying to spoof the algorithms and get higher rank, these web-graph-based models still form the heart of the critical crawlers and rank assessment algorithms that make Web search work.

Given the importance of these search engines, and thus the importance of these graph-based web analyses, it is sometimes forgotten that in reality the Web Graph doesn't exist! The graph is at best a gross simplification of the structure of the Web -- the reality is far more complex. These graph analyses are done on the results of a crawl of the Web, in which the documents (nodes) are static representations hyperlinked together. However, in the actual Web, the pages are often created dynamically from rapidly changing databases, and the links are the results of particular invocations of specific protocols. Most of the Web graph results have been against crawls which only show the result of GET requests using the Hypertext Transfer Protocol and ignore both the fact that there are many variations on the interactions and results of HTTP GET requests that are not shown in the Web graph and also that there are other protocols, for example the Secure HTTP protocol, HTTPS, which includes processing (encrypting and decrypting messages) that are not represented at the graph level.

The problem is the Web graph is just one abstraction of the Web based on one part of the processing and protocols underlying its function. While it is an important result that the Web graph is scale free it is the design of the protocols and services that we now call the Web that make it possible for it to be so. The Web was built around a set of core design components defined in *The Architecture of The World Wide Web, Volume 1* (Jacobs and Walsh, 2004) as "the identification of resources, the representation of resource state, and the protocols that support the interaction between agents and resources in the space."

The links in the Web graph really represent single instantiations of the results of calling a protocol that returns a particular representation (in this case an HTML page) of a resource (the real-world entity being described) based on a Universal Resource Identifier (URI) which serves as an identifier that is common across the entire Web. So, for example, the URI http://www.acm.org/publications/cacm typed into a standard web browser will invoke the Hypertext Transfer Protocol (HTTP) and return an HTML page that contains content that describes the publication known as the *Communications of the ACM*. Note, however, that the content itself contains other URIs that are themselves pointers to objects that are also displayed (such as icons and images) and that the formatting of the page itself may require retrieving other resources such as cascaded style sheets or XML DTD documents. So what we think of as a single link from say a research group's web page to an article on the CACM page, may really involved a number of requests among a number of servers (at the time of this writing, typing the URI for CACM into your browser will cause more than twenty different HTTP-GET requests to occur for seven different types of Web formats).

Additionally depending on the details of the request's header, different representations may be served up to different requesters. For example, the URI may include information that is specialized to the backend of a particular application[1], in which case the HTML produced may vary based on conditions hidden from the client (for example which particular set of machines in a server farm the request is run on), there may be multiple representations available based on the form of the HTTP-GET request (Content negotiation), and what is served to the client may require additional requests to other servers (in the case of temporary or permanent redirection). Additionally, more and more frequently Web sites are running programs using Web scripting languages, making it harder for them to be represented as a simple set of links. None of these issues are directly accounted for in the Web graph models.

As an example of the problems with the Web graph model, consider the situation of a modern search engine. In a June 2007 talk, Udi Manber, Google's VP of Engineering talked about why Web search is difficult (Manber, 2007). He explained that on an average day, 20-25% of the searches seen by Google have never

[1] These are the characters that may follow the last "slash" in the URI, including "?. #, =,&" followed by keywords, making for the long URIs that often show up from dynamic content servers.

been submitted before. Each of these searches, however, generates a unique identifier (using the server specific encoding information) – so in a Web Graph model we would have to represent the requesting document (whether it be a user request or one generated by, for example, a dynamic advertisement content request) linked to the http://www.google.com node. However if, as is widely reported, Google receives over 100M queries per day, and if 20% of those are unique, then more than 20M links that have never been seen before show up in the Web graph every day, or around 200 per second! Do these links follow the same power laws? Do the same growth models explain these behaviors? We simply don't know.

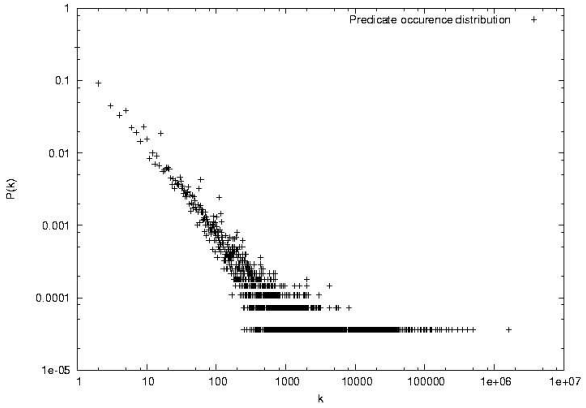
Further, the same request, offered at different times, may result in an actual transfer of information or not depending on the state of cached information both in the requesting client and the requested server even if the underlying resource has changed during the interval. The server may require that the requester has to provide authorization (in which case the link does not show up in the Web graph), may be temporarily down, etc. As well as GET requests, HTTP requests from one site to another may include PUTting or POSTing information that changes the state of the representation for future requesters (and most systems now use the special characters in the URL to do content updating in a RESTful manner, as it has come to be known). There are also a myriad of other special cases that may occur that are not represented in the results of a typical crawl.

Analyzing the Web solely as a graph also ignores many of the issues of the dynamics of the Web (especially at short timescales). Many of the phenomena known to Web users, for example denial of service attacks caused by flooding a server or the need to click the same link multiple times before getting a fast response, cannot be explained by the Web graph model and often can't even be expressed in terms amenable to such graph-based analyses. Representing these totally at the networking level, ignoring the protocols and how they work, also misses key aspects of the Web and a number of behaviors that emerge from the interactions of millions of requests hitting many thousands of servers every second. Huberman and Lakose (1997) explored Web dynamics over a decade ago, but (i) the exponential growth in the amount of Web content, (ii) the change in the number, power and diversity of Web servers and applications, and (iii) the increase in the number of diverse users from everywhere in the World makes a similar analysis impossible today without creating and validating new models of the Web's dynamics. Such models, however, must pay attention to the details of the Web's architecture, and to the complexity of the interaction actually occurring.

The purpose of this discussion is not to go into the details of how the Web protocols work, nor to discuss the relative merits of various three-tiered Web application designs, but rather to stress that these details are critical to the current and continued working of the actual Web. Understanding these protocols and issues is important to an understanding of the Web as a technical construct and to analyzing or modeling the dynamic nature of the Web. After all, our ability to engineer Web systems that have desirable properties at scale requires that we understand these dynamics. Doing this analysis and modeling is thus an important challenge to Computer Scientists if we are to be able to understand the growth and behaviors of the future Web, and to engineer systems with desired properties in a significantly less "hit or miss" way.

4. FROM POWERLAWS TO PEOPLE

Mathematically-based analyses of the Web also have another potential failing. Whereas the structure or use of various Web sites taken mathematically may have interesting properties, those properties may not be very useful at explaining the behavior of those sites over time. Consider the following example. Wikipedia (http://www.wikipedia.org), the online wiki-based encyclopedia, now has over two million articles in English and over six million in all languages combined. These articles are hyperlinked, and it is natural to ask whether these hyperlinks have similar structure to those on the Web in general, or whether, since this is a managed corpus, they have some other, properties.



There are a number of ways such an analysis could be done. Figure 3 shows the result of one such. In this case, DBPedia (http://dbpedia.org) which is a dump of the link structure of Wikipedia using the labeled links of the Resource Description Framework, RDF, has been analyzed with respect to the usage of these link labels (i.e. we are looking at the structure of Wikipedia as opposed to the linguistic content of the pages). This diagram shows the same kind of Zipf-like distribution found in the original web graph analyses. There is also some evidence (Golder and Huberman, 2005) and a lot of speculation (cf. Shirky, 2003) that similar effects can be seen in the use of tags in web-based tagging systems. Current research is also exploring whether these results depart from models such as preferential attachment (Barabasi and Albert, 1999) used to explain the scale-free features of web graphs.

Unfortunately, whatever explains these sorts of effects, there's another aspect of Wikipedia's use that is not explained by these models, and which does not necessarily follow from these properties. Wikipedia is built on top of the MediaWiki software package (http://www.mediawiki.org/wiki/MediaWiki), which is freely available and has been used in many other Web applications since Wikipedia. While some of these have also been successes, many have failed to generate significant use. Clearly, a purely "technological" explanation cannot account for this – rather, something about the organizational structures of Wikipedia, and the needs of its users, account for its success over that of other systems built from the same code base. The model by which articles are created, edited, and tracked is provided by the underlying technology. The social model that is enabled by humans interacting in the ways allowed by that technology, is harder to explain. Understanding the dynamics of a "social

machine" such as this is very complex, and dozens of academic papers, from a number of disciplines, have been written about it (http://en.wikipedia.org/wiki/Wikipedia:Wikipedia_in_academic_studies uses Wikipedia itself to keep an up to date reference list).

This idea of "social machines" was introduced in 1999's *Weaving the Web* (Berners-Lee and Fischetti, 1999). It was hypothesized that the architectural design of the Web would allow developers, and thus end-users, to use computer technology to help provide the management function for social systems as they were realized online. The social machine includes the underlying technology (mediaWiki, in the case of Wikipedia) but also the rules, policies and organizational structures that are used to manage the technology. Examples of these abound on the current Web. Consider the coupling of the Web application design of blogging support systems such as LiveJournal, WordPress and others, with the social mechanisms provided by blogrolls, permalinks and trackbacks that have led to the growth of the so-called "blogosphere." Similarly, the protocols used by social networking sites like mySpace, Facebook and hundreds of others have much in common, but the success or failure of these sites have hinged on the rules, policies and user communities they support. Given the success or failure of Web technologies often seems to rely on these social features, the ability to engineer successful applications requires a better understanding of the features and functions of the social aspects of these systems.[2]

It is also our contention that today's interactive applications are just very early social machines, and that they are limited by the fact that they function largely isolated one from another. We hypothesize that (a) there are forms of social machine that will be significantly more effective than those we have today, (b) that different social processes interlink in society and therefore must be interlinked on the Web, and (c) that these are unlikely to be developed in single deliberate effort in a single project or site – rather, the technology must be developed that allows user communities to construct, share and adapt social machines so that successful models can evolve through trial, use and refinement.

There are a number of research challenges that stand in the way of creating a new generation of interacting social machines that can be created and evolved in this way. Some of these include

- What are the fundamental theoretical properties of social machines, and what kinds of algorithms are needed to create them?
- What are the underlying architectural principles to guide the design and efficient engineering of new Web infrastructure components for this social software?
- How can we extend the current Web infrastructure to provide mechanisms that make the social properties of information sharing explicit and that guarantee that the uses of this information conforms to the relevant social policy expectations?
- How do cultural differences effect the development and use of social mechanisms on the Web? As the Web is now truly "World Wide," the properties desired by one culture may be seen as counter-productive by another. Can Web infrastructure help in bridging cultural divides and/or increase cross-cultural understanding?

In addition, a crucial aspect of human interaction with information is the ability to represent and reason over attributes such as trustworthiness, reliability, and tacit expectations about the use of information, as well as privacy, copyright, and other legal rules. While some of this information is available on the Web today, we lack structures for formally representing and computing over these qualities. Traditional cryptographic security research and well-known access control policy frameworks have failed to meet these challenges in today's online environments and will be insufficient as foundations for the social machines of the future. Recent work on formal models for privacy (Backstrom et al, 2007) has demonstrated that traditional cryptographic approaches to privacy protection can fail in open Web environments. Similar problems with copyright enforcement have also hampered the flow of a wide variety of commercial and scholarly information on the Web (Samuels, 1994). To this end, an examplar Web Science research area we are pursuing involved interdisciplinary research toward augmenting Web architecture with technical and social conventions that increase individual accountability to social and legal rules governing information usage. (Weitzner et al 2008) Continued failure to develop scalable models for handling policy will seriously impede the ability of the Web to become the best media for the exchange of cultural, scientific and political information. Furthermore, we can see from the explosion of new collaborative styles of creating and publishing information on the Web that many of the social institutions we historically rely on to judge trustworthiness, veracity, etc., are missing from our online information life. In short, understanding the Web and being able to engineer its future requires not only an understanding of the Web as a computational structure, but also how it interacts with, and supports the interaction of, people.

Currently, there is significant research exploring many different aspects of the influence of the Web on society. One important aspect of this work is on the creation of online societies using Web infrastructure to support dynamic human interactions (cf. http://trout.cpsr.org). This work explores how the Web can encourage more human engagement in the political sphere. Bringing this work in contact with the emerging study of the Web, and creating a coevolution of technology and social needs is an important focus of designing the Web of the future (cf. Shneiderman, 2006).

5. THE WEB OF DATA

One of the emerging areas of study on the current Web is the heavy use made of tagging provided by many of what have come to be known as "Web 2.0" technologies. On these sites, articles, blogs, photos, videos and all manner of other Web resources can be annotated with user generated keywords, called tags, that can later be used for search or browsing of these resources. Much has been made of how "folksonomies," taxonomies that emerge through the use of tags, can be used as "metadata" that helps to explain the content of the objects being described (cf. Gruber, 2007).

One aspect of tagging that has been generating recent interest is the need for "social context" in the tagging spaces (cf. Marcus and Perez, 2007). Many tags provide terms that are extremely

[2] It is important to note here that when we say "success" or "failure" we are referring not to the business factors that determine for example, whether Facebook or MySpace will attract more users, but rather to the success or failure of these sites to provide the particular types of social interactions for which they are designed.

ambiguous in a general context. For example, first names are very popular tags on Flickr even though they are not very good general search terms. On the other hand, in a specific social context, such as a particular person's photos, the same tag can be very useful since it can designate a particular individual. Thus, the use of the tag as metadata is often dependent on just such a context, and the "network effect" in these cites is thus socially organized (Hendler and Golbeck, 2008).

A more ambitious use of metadata can be found in the recent applications of Semantic Web technologies (Berners-Lee et al, 2001). This technology represents an important paradigm shift that will be a significant element of the next generations of Web technologies. This is because the Semantic Web represents a new level of abstraction from the underlying network infrastructure, as has the Internet and the Web before it: The Internet allowed programmers to create programs that could communicate without having to concern themselves with the network of cables that the communication had to flow over; the Web allows programmers and users to work with a set of interconnected documents without concerning themselves with details of the computers that store and exchange those documents.

The Semantic Web will raise this to the next level, allowing programmers and users to make reference to real-world objects -- whether people, chemicals, agreements, stars or whatever else -- without concerning themselves with the underlying documents in which these things, abstract and concrete, are described. While the basic Semantic Web technologies have been defined and are starting to be more widely deployed, and with further components of the architecture being the focus of current standardization efforts, there has still been very little work in understanding the impact of this new capability on the user interface and how it enables the connections of the Web of people who will use it (Shadbolt et al, 2006).

Currently there are two nexus of activity in the Semantic Web world. One of them, based largely on innovation happening around data integration applications, focuses on the development of Web applications that use very little semantics but provide a powerful mechanism for linking data entities together using the URIs that are the basis of the Web. Powered by the Resource Description Framework (RDF) these applications are largely focused on querying graph-oriented triple-store databases using the emerging SPARQL language. This provides a new means for creating Web applications and portals using REST-based models, but integrating data from multiple sources without preexisting schema. The second focus, based largely around the Web Ontology Language OWL, looks at providing models that can be used to represent expressive semantic descriptions of application domains and can provide inferencing power for both Web and non-Web applications needing a knowledge base. The first focus tends to be on data, the second on domain. The latter focuses on the *semantics,* the former on the *Web.*

With a growing industrial interest in the Semantic Web, a lot of work has been looking at extending each of these. Current research is exploring how the databases of the Semantic Web relate to traditional database approaches, and to scaling Semantic Web stores to very large scales (Abadi et al, 2007). In the modeling space, tools for providing faster inference in large knowledge bases (without sacrificing performance) is a goal, with recent work exploiting tradeoffs between expressivity and reasoning to provide capabilities designed to be deployable at a Web scale (Fokoue et al, 2006). A market is beginning to exist for both "bottom up" tools, driven by the data, and "top down" technologies, driven from the ontologies. Bottom up, creating backends for the Semantic Web is transitioning from an arcane art to an emerging Web application programming approach as new open source technologies are starting to integrate well with traditional Web servers. Top down, new tools are becoming available for supporting ontology development and deployment and literally tens of thousands of OWL ontologies are available for jumpstarting new domain modeling efforts. In addition, approaches using rule-based reasoning modified for the Web are also gaining attention (Berners-Lee et al, 2008). Engineering the future Web includes the design and use of emerging technologies such as these, and exploring the similarities and differences from the traditional approaches to databases, in the one case, and artificial intelligence, in the other.

We note that the Semantic Web is a good example of the issues we have been discussing in this paper about understanding Web systems. It is currently one of the key emerging technologies on the Web, but as evidenced in the above, there are different opinions of what it is best for and, more importantly, what the macro effect might be. Our lack of a better understanding of how Web systems develop and grow make it hard for us to know what this technology will impact at scale. What social consequences might there be of the greater public exposure and sharing of information that is currently locked in databases? A better understanding of how Web systems move from the micro to the macro would provide a better understanding of how this, and other new computing technologies on the Web, could be developed and fielded, and what the potential societal impacts might be.

6. CONCLUSION

The Web is different from most hitherto-studied systems in that it is changing at a rate which is of the same order as, or maybe greater than, our ability to observe it. It is an unavoidable fact that the future of the world is now inextricably linked to the future of the World Wide Web. We have a duty to ensure that the future developments in the Web will make the world a better place; corporations have a responsibility to ensure that the products and services they develop on the Web don't have side effects that will do harm to society; governments and regulators have a responsibility to understand and anticipate the consequences of the laws and regulations they make.

We cannot achieve these aims until we better understand the complex, cross-disciplinary dynamics that drive development on the Web. This is the main aim of Web Science. Just as climate change scientists have had to develop ways of gathering and analyzing evidence to prove or disprove theories about the impact of human behavior on our climate, web scientists need to develop new methodologies for gathering evidence and finding ways to anticipate how human behavior will impact on the development of a system which is constantly evolving at such an amazing rate. We have to consider what would happen to society if access to the Web was denied to some or all, and to raise awareness amongst major corporations and governments that the consequences of what appear to be relatively small decisions can have a profound impact on society in the future by affecting the future development of the Web.

Computing, in all its forms, plays a crucial role in the Web Science vision, and much of what we know about the Web today is based on our understanding of it in a computational way.

However, as we have argued in this article, to be able to engineer the Web applications of the future, there is significant research to be done. We have to understand the Web as a dynamic and changing entity, and to explore the emergent behaviors that arise from the "macro" interactions of people enabled by the Web's technology base and we need to understand the "social machines" that can be the critical difference between the success or failure of Web applications, and learn to build them in a way that allows interlinking and sharing.

7. ACKNOWLEDGMENTS

Figure 2 is taken from talks given by Tim Berners-Lee in 2007 available online from http://www.w3.org/2007/Talks/1018-websci-mit-tbl/Overview.html). Some of the ideas in this paper expand on themes that were discussed in Ben Shneiderman's recent viewpoint article (Shneiderman, 2007) building in part on an earlier article the five authors of this article had published (Berners-Lee et al, 2006). We also thank the other members of the WSRI Scientific Council (see http://webscience.org/about/people/) for inputs relating to the goals of Web Science and the interaction of the Web and Computer and Information Sciences. We are indebted to Konstantin Mertsalov for the DBpedia analysis discussed in Section 4.

8. REFERENCES

D. Abadi, A. Marcus, S. Madden, and K. Hollenbach, Scalable Semantic Web Data Management Using Vertical Partitioning, *Proc VLDB*, 2007

L. Backstrom,, C. Dwork, and J. Kleinberg, Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography, *Proceedings of the 16th international World Wide Web Conference*, Banff, Alberta, Canada, 2007

A. Barabasi and A. Albert, Emergence of scaling in random networks, *Science*, 286, 1999.

T. Berners-Lee, W. Hall, J. Hendler, N. Shadbolt, and D. Wietzner, Creating a Science of the Web, *Science*, *311,* 2006.

T. Berners-Lee, W. Hall, J. Hendler, K. O'hara, N. Shadbolt and D. Weitzner, *A Framework for Web Science,* Foundations and Trends® in Web Science, 1(1), 2006.

T. Berners-Lee, J. Hendler, and O. Lassila, The Semantic Web, *Scientific American,* May, 2001.

T. Berners-Lee, T. and M. Fischetti, Weaving the Web: The original design and ultimate destiny of the World Wide Web, Harper Collins:NY, 1999.

T. Berners-Lee, D. Connolly, L. Kagal, Y. Scharf, and J. Hendler, N3Logic: A Logical Framework For the World Wide Web, *Theory and Practice of Logic Programming*, 2008

S. Brin, and L. Page, The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Proc. WWW,* 1997.

Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, S. Stata, A. Tomkins and J. Wiener, Graph Structure in the Web, *Proc. WWW*, 2000

J. Dean and S. Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, OSDI'04: Sixth Symposium on Operating System Design and Implementation, San Francisco, CA, December, 2004.

S. Dill, R. Kumar, K. McCurley, S. Rajagopalan, D. Sivakumar, ad A. Tomkins, Self-similarity in the Web, Proc VLDB, 2001.

P. Domingos, J. Golbeck, P. Mika, A. Nowak. "Social Networks and Intelligent Systems," IEEE Intelligent Systems, Trends & Controversies, Jan/Feb 2005.

D. Donato, L. Laura, S. Leonardi, and S. Millozzi, The Web as a Graph: How far we are, ACM Transactions on Internet Technology, 7(1), 2007.

A. Fokoue, A.Kershenbaum, L. Ma, E. Schonberg, and K. Srinivas. The Summary Abox: Cutting Ontologies Down to Size, Proc ISWC 2006.

S. Golder and B. Huberman, The Structure of Collaborative Tagging Systems, arXiv:cs/0508082, 2005 (http://arxiv.org/abs/cs/0508082)

A. Gulli and A. Signorini, The Indexable Web is More than 11.5 Billion Pages, Proc WWW, 2005.

J. Hendler, Web 3.0: Semantic Web Chicken Farms, IEEE Computer, Jan, 2008.

Metcalfe's law, web 2.0, and the semantic web , James Hendler and Jennifer Golbeck, *Journal of Web Semantics*, 6(1), 2008

B. Huberman, and R. Lukose. Social dilemmas and Internet congestion. *Science* 277, 1997.

I. Jacobs and N. Walsh, Architecture of the World Wide Web, Volume One, W3C Recommendation. 15 Dec 2004 (http://www.w3.org/TR/webarch/)

J. Kleinberg, Authoritative sources in a hyperlinked environment, Journal of the ACM (JACM). Volume 46, Issue 5 (September 1997)

J. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, The Web as a Graph: Measurements, Models, and Methods, Proc. Computing and Combinatorics: 5th Annual International Conference, COCOON'99, Tokyo, Japan, July 1999

R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, Trawling the Web for emerging cyber communities, *Proc. WWW*, 1999.

U. Manber, Why Search is a Hard Problem, presentation at Supernova 2007

Marcus, Aaron and Angel Perez. "m-YouTube Mobile UI: Video Selection Based on Social Influence" *Proceedings of the 12th International Conference, HCI International*, 2007.

P. Samuelson, Copyright's Fair Use Doctrine And Digital, *Communications Of The ACM*, 1994

N. Shadbolt, W. Hall, and T. Berners-Lee, The Semantic Web Revisited, *IEEE Intelligent Systems,* May/June, 2006.

C. Shirky, Power Laws, Weblogs, and Inequality, Clay Shirky's Writings about the Internet, (Weblog), 2003 (http://www.shirky.com/writings/powerlaw_weblog.html)

B. Shneiderman, Web Science: A Provocative Invitation to Computer Science, *CACM*, 6/2007

D. Weitzner, J. Hendler, T. Berners-Lee, and D. Connolly. Creating a policy-aware web: Discretionary, rule-based access for the world wide web. In Elena Ferrari and Bhavani Thuraisingham, editors, *Web and Information Security.* IRM Press, 2006.

D. Weitzner, H. Abelson, T. Berners-Lee. J. Feigenbaum, J. Hendler, G. Sussman, *Information Accountability, Communications of the ACM*, June 2008.