INFORMATION SCIENCE PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITB13366

This chapter appears in the book, *Advances in Electronic Business, Volume II*
edited by Eldon Li © 2007, Idea Group Inc.

Chapter I

From World Wide Web to Semantic Web

Charles Ling-yu Chou,
The Chinese University of Hong Kong, Hong Kong, China

Abstract

This chapter introduces the incentives for the creation of the Semantic Web, the methodology for its development, and the current status of this development. In contrast to the human function of understanding, the author summarizes four major steps in creating the ability for machines to understand Web content and generate responses. The semantics in the Semantic Web should be explicitly declared in a form that can be operated by a machine, knowledge organization should be provided to support semantic interpretation, software agents must rely on automatic reasoning ability to obtain implied knowledge, and procedural knowledge should be accessed in a community to generate response behavior. Through illustration of the anticipated research efforts in this technology, the author hopes to provide a clear picture of the current status of emergent Semantic Web technology and a suitable direction for its future development.

Introduction

The World Wide Web is an information universe, in which countless nodes of Web content that provide links to online and off-line resources are connected together. Although this information universe tends to be unbounded, the technology that supports it is simple. The World Wide Web is constructed using just three fundamental standards: the uniform resources identifier (URI), the hyper text transfer protocol (HTTP), and the hyper text markup language (HTML). These three standards perform different functions individually, but serve collaboratively to enable information consumer to access remote resources by retrieving Web content from remote nodes and presenting them on local machines.

The URI that is given to each document is a unique address on the World Wide Web, and is the universal identification of the node that is sought. HTTP is one of the communication protocols for information transmission between two participating applications, with which the requesting application can send enquiries to remote applications to retrieve Web content. The requested application then responds with Web content, which is usually packaged as documents in a message body, to the requesting applications.

Upon receiving the requested Web content, the receiving application launches a corresponding display device on the local machine to present the Web content. To be expressible, the Web content must be specified with some demonstrating properties that configure the expected presentation effects for the respective video or audio devices on which the content will be displayed. Content designers use HTML to annotate their configuration of content, which works like a markup that highlights elements of the content with explanatory notion. Thus, two different kinds of information, the main content and the annotated configuration, are synchronized in the same HTML document. Generally, an application that is specifically for remote resource access and Web content display is called a Web browser. When a Web browser receives a document, it knows which part of the document is the main content and how to display the main content with the expected effects using the configuring information that is provided by the other parts of the document. General presentation effects include control of the size, the color effect of the textual content, and the invoking of the configuration for specific devices to display images, videos, or audio files. The annotations that are marked in HTML allow the Web content to be presented in a flexible way by various media, as long as the host machine is properly installed with the necessary devices for presentation. Often the annotation may also include links to other relevant information that allows information consumers to "navigate" the information universe among relevant pages. Through years of effort, the World Wide Web can now be considered as the most influential technology of the late 20th century. Its widespread use among the general public has changed modern life in the developed world in many ways.

Changes in the channels of messages exchange. One of the most significant changes that have been caused by the World Wide Web is that people now increasingly rely on this new message exchange channel to discover and disseminate information. For example, academic scholars rely on the World Wide Web to search for literature. They can complete a comprehensive search for the digital images of journal papers, which cover almost all of the results of research that has been carried out in the last few decades in their field. Usually the number of returns from searches is much more than can be digested. For instance, one of provider of the digital image repository service, ProQuest Company, already provides links to more than ten million full-text items from periodicals and newspapers. Another example is the recently announced digital library project from the Internet search engine Google. The project intends to provide information explorers with comprehensive searching privileges to publicly accessible materials and the copyrighted items that are stored in university libraries. For those who intend to explore new business opportunities, the World Wide Web can be used to collect relevant links to corporate Web sites to obtain detailed product information by using the services of commercial directories, such as Yahoo and Google.

The World Wide Web at one pole demonstrates its dominance in information exploration; and at the other pole, it is a universal platform that people can use to promote their beliefs. As long as Web content is well formatted and is linked to the Internet, any information can be posted and people have an equal opportunity to speak out over the World Wide Web. Moreover, through the collection and dissemination of information, huge business opportunities can be created, such as occurred with the well-known case of Yahoo.com, which took the simple concept of a Web page directory that was originally created by two PhD candidates two years ago, all the way to a public offering that was worth 33.8 million dollars in 1996 (docs.yahoo.com/info/misc/history.html).

Changes in the pattern of collaboration. The World Wide Web also facilitates new forms of collaboration. In a highly differentiated society, most tasks must be done with collaboration among resources. It is almost impossible for a single organization, not to mention a single operator or knowledge worker, to complete a noteworthy task. Collaboration among a team of people requires coordination and communication to guarantee that the team is working in the same direction toward the same goal. Before networks become popular at the late 20th century, coordination and communication among people on a project was often implemented through a series of regular meetings of team members, piles of circulated documentations, and travel between distant cities. Since the appearance of the World Wide Web, the patterns of collaboration have changed, and new patterns have emerged that provide better synergy, efficiency, and convenience in the execution of teamwork. For example, a new car model can be developed over the World Wide Web by the simultaneous collaboration of engineers who are distributed across different locations and different time zones. New product data management (PDM) tools provide Web-enabled virtual working platforms that work with integrated digital vaults for engineering data and project documents, and allow remote design meetings to take place over the World Wide Web. Thus, the people who are working on the project are ensured access to the same version of the data, and the Web-enabled design meetings allow people in various locations to edit the same engineering components on a 3-D visual model. In the design meeting, one person rotates a component by a certain degree and marks a comment on a specific position of the component, while persons who are situated at distant places can see the rotation and the notes at their panel as soon as they are made. Similarly, the World Wide Web can enhance product data sharing in a supply chain. Using product data that are retrieved from a Web-enabled electronic catalogue on the supplier side, a designer can construct a 3-D visual model of the components that are needed and can simulate the assembly process virtually to decide the feasibility of using the components from the suppliers.

The Challenges

The deployment of the World Wide Web was constrained in the early stages by its presentation specification. Web content is designed for human consumption, and the presentation of Web content can only be interpreted and understood by human users. Unfortunately, this rudimentary characteristic of Web content has become a major limitation to the World Wide Web today.

According to an Internet domain survey by the Internet System Consortium, the number of hosts that advertised on DNS was 317,646,084 in January 2005 (www.isc.org). Hosts serve as gateways to Web pages, and the number of Web pages that are maintained by each host may range from tens to millions of Web pages. The huge volume of information and resources that are becoming available over the World Wide Web has exposed the limited ability of humans to screen information for relevancy. In other words, information overload has become a serious problem for humans in locating useful information for their own purposes. For example, a person searching for the lyrics to the song "Yesterday" by John Lennon could easily identify the right song from dozens of lyrics simply by searching using a few words from the song that they might be able to remember. However, if the searcher does not recall that the song was written by John Lennon or any of the words and initiates a search using the keywords "lyric" and "yesterday" in Google, the search will respond with 759,000 links to what it considers to be relevant pages. This creates a problem. How can a human being screen all of the pages to find the specific lyrics to the song?

The limits of human ability also prevent Web-enabled collaboration from becoming widely adopted in practice. As has been stated, Web-enabled collaboration is a new pattern of communication that participants from multiple corporations can use to form virtual corporations. However, this new pattern is currently only deployed among organizations that maintain a tight relationship with their various arms or that are targeting projects with long lifecycles. This is because the implementation of a virtual corporation relies on the deep integration of business data and enterprise applications, which ensures that virtual meetings are supported by seamless intra-operations among different proprietary systems, such as PDM and CAD. This integration also assures that message exchanges among organizations are not confused by ambiguous data definitions. To provide support in such level, the parties that are involved in a virtual corporation are required to make a huge investment of human effort to guarantee the successful integration of data and applications. This requirement has become the major barrier to the deployment of Web-enabled collaboration as a means of seizing contingent opportunities in the dynamic business environment.

The Vision

In identifying the improvements that need to be made to the present World Wide Web, several leading scholars have envisioned a new generation of the World Wide Web and the collaboration patterns therein (Berners-Lee, Hendler, & Lassila, 2001). In their vision, humans in the near future will rely on intelligent software agents to search and interact with Web resources (including Web content and automated services) over the World Wide Web. These software agents_are software objects that operate autonomously in a software environment to achieve certain predefined goals (Du, Li, & Wei, 2005; Fuggetta, Picco, & Vigna, 1998). Actually, using software agents as performers over the World Wide Web has already been carried out on a limited scale. A practical example is the software robot that helps marketers to collect and compare book prices over several specified online bookstores (Uschold, 2003). However, these shopping robots normally only surf specified routes and visit predetermined types of Web content, and the information that is being compared is easy to identify because unambiguous terms, such as the international library book code, quantity, and price, are used. Software agents in the future will be very different from their current incarnation, and will be empowered with humanlike intelligence. Typical humanlike intelligence will use practical logical reasoning to deduce and induce, and will recognize patterns through environmental stimulation to determine contextual influence, learn from previous experience, and find adjusting strategies to adapt to environmental changes (Croft, 1997). Software agents that have humanlike intelligence can navigate over World Wide Web, and operate with other automated Web resources on behaves of the humane being. In order to entertain the demand of the visiting software agents, the information being published over World Wide Web must be articulated with intended meaning in a way that machine can recognize and interpret. Such demand for information representation is not supported by current Web specifications, but it will be supported by new generation of Web, called "Semantic Web". Moreover, semantics will become a searchable, portable, and reusable object to assist software agents to cooperate with other automated Web resources to complete sophisticated tasks without human intervention (Berners-Lee et al., 2001).

What is the Semantic Web?

The Semantic Web is defined as "the extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" (Berners-Lee et al., 2001). In this definition, two views are highlighted. The first relates to the meanings of the Semantic Web, and the second relates to the functions of the Semantic Web.

First, the emergence of the Semantic Web does not entail the replacement of the existing World Wide Web. Instead, the Semantic Web will be implemented on current networked information space, and will utilize HTTP to transport messages. The difference is that the Semantic Web requires an unprecedented descriptive structure for content description to cater to the demands of the software agents (Goble, 2003). This does not mean that the conventional expression of the content of the World Wide Web for human consumption will be discarded, but rather that Web content will be expressed simultaneously in a way that machines can also understand.

Second, the newly established Semantic Web is expected to explore the collaboration of humans through software agents and automated Web services. In light of this expectation, scientists and industrial practitioners have invested much effort into constructing an infrastructure to facilitate the expected collaboration among humans and commissioned software forces.

Two questions arise from this definition of the Semantic Web. How do machines do to behave like it understand Web contents, and what should we do to make machines behave in that way? The following elaboration attempts to provide answers to these two questions, and is followed by a review of the current status of Semantic Web technology.

What is Semantics?

Semantics is the "meaning" of expression. As we use a symbolic system to represent the real world or the conceptual phenomenon thereof, semantics is the meaning of these symbols and sets of symbols. If relationships are implied between meanings, then semantics also refers to inference, that is, the transition of the meanings from one meaning to another one (Merriam Webster Online dictionary, www.m-w.com). People usually use many symbolic systems simultaneously to help communicate their thoughts to others. Some typical symbolic systems are natural language, mathematical and chemical formula, and programming languages for software programming. Each symbolic system has its own semantic system. And people usually know how to use them, when they decide to use that language system in communication. Additionally, there are multiple types of semantic contribute, either implicitly or explicitly, to successful communication. Let us take communication over the World Wide Web as an example. Obviously, natural written and spoken languages are the most common symbolic systems to express opinions. The meaning of the terms and the relationship between the terms defined in the natural language are the most basic type of semantics contributing to our communication. Machine communication protocols are also symbolic languages, but they are designed for communication between machines, rather than humans. As machine behavior is hard-coded by human programmers, such semantic types are also predefined and limited by the programmers. Another kind of semantic is generally needed when people make logical conclusions. People can read more meaning between the lines, which is known as referred meaning. Semantics that guides people, either explicitly or implicitly, to imply meaning from assertions is another type of semantics, and is sometimes called *rules* or *logic*. This type of semantics can be generally found in symbolic languages, such as mathematical formula and programming languages.

Beyond symbolic expression, background knowledge is also an important source of meaning. Such knowledge usually aids the inference of assertions by the receivers of expression, and sometime provides procedural knowledge to generate interacting patterns in the deliverers of expression. Procedural knowledge helps us to meet expectations when we interact with others in a specific process. For example, when we transfer an amount of money between accounts through an online banking service, procedural knowledge guides us through the process step by step, and we can anticipate that the amount will be deducted from our account accordingly once the process has been completed.

In the work of categorizing the general semantics that are used in human life, Uschold proposed a general framework for the analysis of semantics that employs three categories: implicit versus explicit, formal versus informal, and intended for humans versus intended for machine processing (Uschold, 2003).

1. **Implicit vs Explicit.** Explicit semantics differ from implicit semantics in that there is an explicit declaration of the specification of the symbolic system. In contrast, the meaning of the symbolic system in implicit semantics is normally conveyed based on a shared understanding that is derived from consensus. For example, when a shopping agent encounters price information, it will implicitly be aware that it may refer to the amount that we need to pay for the acquisition of certain goods. Although the definition of "price" is defined in certain dictionaries, it is seldom as clearly defined on a Web page or related pages in a corporate Web site.

   In contrast with this intuitive conception, Sheth proposed an innovative perspective for the interpretation of "implicit semantics" (Sheth, 2005). Sheth argued that implicit semantics can be implied not only from a shared consensus on the communicating content, but also from undiscovered knowledge from all kinds of data. From this perspective, "implicit" refers to intangible or undiscovered knowledge that cannot be transformed into explicit knowledge. For example, the attempt to allow marketers to extract information on consumer behavior from a marketing data set is a process of extracting so-called implicit knowledge and turning it into explicit knowledge. This interpretation implies that in the future, as we rely more and more on statistical analysis in our decision making, implicit semantics may be hidden in the process. And never may we know that this kind of semantics is being extracted from the data set and used to determine a decision, although we benefit from the operations.

2. **Formal vs Informal.** The difference between these two types of semantics lies in whether they are ambiguous or not. Semantics should be specified clearly through formal documentation or the specification of meaning. A formal language is often used to reduce ambiguity and avoid inconsistent and incompatible implementation. Examples such as Modal logic are used to define the semantics of ontological categories (Guarino, Carrara, & Giaretta, 1994). The articulation in the documentation helps humans to reduce ambiguity while using the idea of rigidity and identity in communication (Uschold, 2003). Modal logic is also used in agent communication language (Smith, Cohen, Bradshaw, Greaves, & Holmback, 1998) to define the semantics of performatives, such as informing and requesting. Such formal definition helps humans to hardcode software agents that can properly understand, evaluate, and compare exchanged messages with others using alternative agent communication languages (Uschold, 2003).

   Informal semantics are defined in an informal notation or natural language. For example, a glossary or a text specification document can explicitly denote certain concepts, but cannot infer decisive or determined assertions by a strict process of deduction. Therefore, this kind of semantics can rarely be understood and processed for further use by a machine.

3. **Intended for Human Processing vs Intended for Machine Processing.** Although all formal languages offer specific semantic meanings to disambiguate subtle

differences, not all formal specifications can be adopted by machines for the purpose of inference. Whether the semantics that is defined in formal languages can be simulated by machine calculation and processed to generate sound and useful inferential results thus marks the difference between these semantic types. The development of the enterprise ontology has provided many clear axioms and definitions (Uschold, King, Moralee, & Zorgios, 1998), but at the time of its development it was not expected to be used as the foundation for automated inference, and therefore further securitization should be undertaken before its adoption as a machine-processable semantic system.

Uschold points out that the semantic language of the Semantic Web should be machine understandable and explicitly specified, formal language (Uschold, 2003). However, no approach to the development of this kind of semantics has been developed. Therefore, in the following sections, I try to make an analysis based on the human operation of "understanding" to derive a proper approach for the development of explicit, and machine understandable semantics.

How do Humans Understand Expression?

Having explored the meaning of semantics, we can start to consider questions about the "understanding" of semantics. We can start with the question of what it means for humans to understand expressions, which is naturally followed by the question of how can we replicate the same behavior in a machine, and finally leads to the question of how we should adopt what we know about the process of human understanding to program a machine to "understand" semantics.

In a normal communication scenario, the "understanding" of a discourse is conceived as the activity of digesting the meaning of the discourse and then generating an appropriate reply to the issuer of the discourse. The digesting process is a procedure of reconstructing either the opinions of the author or speaker, or the statements of facts about the world (Russell, 2003). For example, while listening to a speech, we will try to capture the facts about the world that are revealed in the textual expressions and the opinions or belief that the speaker is trying to convince the audience to believe.

The reconstruction process actually involves several consecutive information processing procedures. These procedures are performed in sequence to achieve the sub-consequences of perception, analysis, disambiguation, and incorporation (Russell & Norvig, 2003).

- **Perception.** Perception is the incipient stage of the understanding process. A person who receives some forms of discourse through various external and physical forms, such as images, printed materials, and voice, first recognizes the symbols through the functions of the brains and the sensory receivers, and then transfers them into internal data correlations that can be analyzed in further processes. For example, if the medium is printed material, then the receiver first recognizes and interprets the markings of the texts into words in the sentences, and then puts them in a similar correlation of textual sequences in the memory. When listening to a speech, in contrast, we need to recognize the materials that are represented by the voice.

- **Analysis.** The analysis on the receiving discourses can be divided into two main portions: syntactical interpretation and semantic interpretation. Syntactical interpretation generally refers to the parsing process, in which a unit of a continuous stream of expression is decomposed into atomic components with meaning to determine the grammatical structure of the parsed elements with respect to a given set of grammar rules. Semantic interpretation, then, represents the process of associating the knowledge that we have in the memory of symbols and patterns with the parsed components and grammatical structure.
- **Disambiguation.** Ambiguity in the analysis results is a common phenomenon in most languages. For instance, a sentence in the discourse expressed in natural language may be analyzed with multiple interpretations, all of which are legal and can be alternative readings, of the same expression. This can cause ambiguous understanding, although that may not have been the speaker or author's intention. However, in considering other environmental determinants, humans can usually decide on the correct interpretation that best reflects the speaker or author's intention and the facts of the world. When humans practice this disambiguation process, domains of interest, the context of communication, and temporally related factors, among others, contribute to making the best decision as to meaning (Wittgenstein, 2001; Staab, Santini, Nack, Steels, & Maedche, 2002; Baader, 1999).
- **Incorporation.** After obtaining a direct interpretation from these processes, a receiver will incorporate their knowledge and the collected information into a set of confidential conclusions about the incoming discourse. This extra effort on the part of receivers generally includes drawing inferences from new assertions, and validating the plausibility of the conclusions that are explicitly stated in the discourse or are implied by inference.

The ability of humans to draw a new conclusion or to evaluate a proposed conclusion from information that has already been collected is generally referred to as "reasoning" (Wason & Johnson-Laird, 1972). Reasoning is usually either deductive or inductive. Deductive reasoning is the process of reasoning based on logical principles that lead us from known assertions to determined conclusions (Williams, 2000; Johnson-Laird, 2000). Deductive reasoning does not increase our knowledge to help us to interpret the world or discourses; it merely uncovers implicit facts that are not disclosed in the discourse. In contrast, inductive reasoning is the process of using specific facts to generalize a likely conclusion that may explain the facts. The conclusions that are generated from inductive reasoning may become a new axiom that enhances our knowledge base for future understanding (Johnson-Laird, 2000). The reasoning process takes all conclusions as legal inferences according to the principles of logic, yet we will not admit the truth of all of the conclusions that we derive from expressions and inference. Some opinions may not be true, and some of the implications of expression may contradict our beliefs and observations in a specific domain or certain circumstances. Therefore, humans may need to make subjective or objective judgments on the portions of the conclusion that are not acceptable to their knowledge. This authentication process, which is termed “proofing,” is the process of weeding out less plausible conclusions and accepting the most convincing conclusions. This process is carried out by taking facts or observations as evidence to validate the plausibility of a proposal from conclusions that are inferred from the aforementioned understanding processes. After the authentication process, the validated conclusions that are left at the end of the procedure become the set of propositions that represent what we know and believe about the intention of the speaker or author or the projection of the phenomenon from the expression.

- **Reply.** In a common communication scenario, the activity of “understanding” incoming message is normally followed by corresponding reactions. The reaction may be another expression that is generated to reply to the original author or speaker, or a series of operations that are correlated with the instructions that are given in the expression. To generate these reactions, people take the proofed conclusions as the premise and figure out how they will react. At this point, people deploy a special kind of knowledge, known as procedural knowledge, to govern the implementation of the replying processes (Sternberg, 2003). Normally, a task is done by a sequence of several operations. Procedural knowledge keeps information about the operations under consideration in specific step of the sequence, and information about the rules governing the determination of operations. The rules are contained in terms of causal relationships, which involve putting certain factual statements as the premise and then deriving the solution and action to take.

In our everyday lives, simply a tiny task would involve complicated engagement of multiple rules and determinations. These self-related rules exist in a natural structure correlated by the common factual states of the world in the causal relationships. Cognitive psychologists have found that these modules are normally recalled and implemented unconsciously in groups (Sternberg, 2003). By the same token, these patterns are adopted by computer engineers to group rules into subroutines and modules of operations. It is interesting that the deployment of procedural knowledge in the process of generating corresponding response behavior can be likened to the utilization of predetermined modules of procedures in a computer application, in that each module of procedural knowledge is prepared to cope with a specific operational demand. These may be formed by common norms, special regulations, or agreement among groups. Once the demand is identified, the module will be brought up as the proposed piece of corresponding behavior to be enacted.

Efforts to make machines or applications understand instructions from humans have been made since machines were found to be an efficient and effective way to execute repeated tasks. Generally, applications and machines are programmed in the following methods. While an instruction is input through interface, the machines know how to parse the instructions, and then interpret the meaning of the parsed tokens, invoke relevant modules for operations, and finally generate responses to humans to show they have understood the instructions well and can carry out requested tasks. For example, most general Database Management System (DBMS) provide database administrators with a common communication option to interact with the system using a special query language, which is called structural query language (SQL). Database administrators use SQL to issue their orders to query and update data sets that are stored in the database and that are maintained by the DBMS. A general SQL instruction appears in a similar format to the following example:

Select *Part_No, Product, Price*
From *price_list*
Where *Product = Digital_Camera*

The process of understanding an instruction can be elaborated as follows. First, the DBMS invokes a parsing program to parse the instruction. In this specific example, the parser separates the instruction into three sets of tokens series and then translates the three series into grammatical structures by associating the relationships between the parsed keywords (typed in bold italics in the scripts) and the roles of the parameters (typed in normal italics in the scripts). After parsing the instruction, the DBMS will look in its command dictionary for the semantics of the keywords and the roles of the parameters. Through the identification of the keywords and the roles of the parameters, the DBMS is able to understand the intention of the administrators. The intention here is to request the DBMS to retrieve several records from the table "price_list" that satisfy certain requirements in the "product" column and then list out the value of the named features in the retrieved records. The intention is thus executed by the DBMS through the invocation of a series of routine modules using the parameters that were assigned in the instructions. Finally, the DBMS reports the results of the work to the terminal device for review in a predefined format that is understandable to the administrators.

The same approach in the example is generally adopted in the development of many languages for communication between humans and machines, as well as in many of the protocols between two applications, such as HTTP on the World Wide Web. Yet, the common strategy is taken under several presumptions that prevent it from becoming a solution to demand from Semantic Web. First of all, the responsibilities of the applications or machines are contained in specific domains, and the knowledge that is required to assure successful communication is of a limited scale. Secondly, since that the knowledge domain in use is easily predicted, the keywords and syntax are simple and barely ambiguous so as that the analysis of the semantics can be carried out by simple parsers. Thirdly, the semantics of the symbolic system is shared in an implicit consensus among the users of the application. Application programmers hardwire the semantics into the application procedures, and the application users should learn to communicate with the application using the same agreed-upon semantics. Since the semantics is specifically contained in the agreed keywords and syntax. No ambiguous expressions are allowed in the instruction; otherwise the parsers will deny the orders for further interpretation. Fourthly, it is clear that both disambiguation and reasoning are unnecessary in the understanding procedure. Lastly, the procedural knowledge always pertains to the domains of the application, and its operation is also hard coded as modules in the application.

Requirements for the Semantic Web

I would now like to conclude this section by looking at how software agents can be made to understand Web content. There are several requirements that need to be fulfilled in the development of this capability.

The semantics should be declared explicitly. Humans use natural languages and other complex symbolic systems that possess complicated syntax and ambiguous semantics to format their Web content. The convenience of language selection makes it impossible for software agents to ascertain the meaning of such Web content using parsers, each of which represents the articulation of knowledge in a particular domain. Additionally, it is less likely that software agents will be able to negotiate in advance with the potential information providers to identify the relevant parsers that would be needed to analyze the semantics in specific domains. Taking these concerns into consideration, the first breakthrough in the development of Semantic Web will be the presentation of semantics in a form with which machines are familiar. This breakthrough could be implemented in two ways: through the transformation of Web content to a form that software agents can process, and by providing software agents with clues as to the semantics that occur on the World Wide Web.

Normally, software agents can recognize data, which is complied with a predefined data schema, and propositions. A typical example of the data in a schema is a set of records of personal information. A schema, such as (<u>Personal_ID</u>, Name, Birthday, Gender, Nationality, Address, Phone_number), reveals the important information of a residence registry. A proposition, in the view of cognitive psychologists and artificial intelligence experts, is a factual assertion that reveals a determined relationship between two or multiple concepts. Philosophers generally believe that any phenomenon that reveals complex facts about the world can be analyzed into a simpler phenomenon, and that this recursive process can be carried out until an atomic form of the original proposition is reached (Wittgenstein, 1974; Staab et al., 2002). Eventually through this process the real world can be represented using a group of propositions and a set of factual data. Following this methodology, information inside Web content that is relevant to further processes of reasoning and response generation would be possible to be presented in forms of data with predefined data schema and factual propositions.

However, data schema and the implicit knowledge of concepts that is contained in propositions cannot make themselves known to software agents, and therefore an auxiliary artifact is needed to make the implicit information explicit. The artifact that is used is similar to that which is implemented by HTML. Analogous to the same method that is used to annotate the configuration to present effects by the HTML label, the implicit semantics of the elemental components of Web content is marked up by the annotation of an explicit terminological system, which will be related to an explicit declared knowledge base. By the supply of the relevant information in the knowledge base, the software agent would be able to acquire enough auxiliary clues to the interpretation of the discourse in Web.

A knowledge organization should be provided for semantic interpretation. As has been stated, the attempt to explicitly declare semantics presumes the existence of a relevant knowledge base that helps software agents to recognize the required knowledge. Therefore, the establishment of a useful knowledge base to the interpretation of the annotation becomes one of the important subjects under discussion. The approach of deploying a knowledge base as the solution to the provision of implicit information has been studied by artificial intelligence experts for a long time. The foundation of the knowledge base approach is the dismantling of the bundle of procedural knowledge and declarative knowledge. This intention is similar to the same practice in the implementation of database management systems, which involves the efforts to separate the data schema from the future processes that will deploy it. By the same token, philosophers and cognitive psychologists believe that the knowledge that we have about the world, or declarative knowledge, and the knowledge that we use to implement processes, or procedural knowledge, are separable according to the structure and functions of knowledge. Yet, in classical programming practice, while an application is tailored to a specific problem domain, the knowledge that is used to describe the problem, which relates to declarative knowledge, and the knowledge that is used to describe the application domain, which relates to procedural knowledge, are implicitly intertwined in the coding of the application. Moreover, the knowledge to develop the applications is only contained by experts in such knowledge domain. A knowledge base is the aggregation of explicit descriptions that is extracted from the experts of specific problem domains. A suitable knowledge base in Semantic Web should be represented in a form that is neutral to the applications that will operate using the knowledge, which means that the reusability of the declarative knowledge will be extended and will not be constrained by the future procedural knowledge.

To assure the relevant and efficient retrieval of declarative knowledge, the propositions and concepts should be correlated in a well-constructed knowledge organization. There are several possible models to provide such well-constructed knowledge organizations. Among those proposed solution, two models are most representative and widely adopted: the categorical concepts model and the semantic network model. The categorical concepts model mainly provides a taxonomical architecture for the maintenance of implicit relationships among concepts and categories. The concepts in this model are described by a set of feature components that singly perform unique elemental concepts and jointly define the distinctive existence of the mother concepts in the organization. According to the similarity of the shared defining features, several concepts can be grouped into a less detailed category. This consolidation process can be iterated from the bottom to the upper level until ultimately general categories are reached. Knowledge Organization constituted with this model provides two kinds of crucial information, which can be directly revealed in its structure. It tells us the elemental features that used to define and identify the concepts. And, it tells us the inheritance relationship as well as the differences among affiliated concepts. The semantic network model use a propositional network to relates propositions. The first model of a propositional network, the semantic network, was presented in 1969 by Allan Collins and Ross Quillian (Collins & Quillian, 1969). In this framework, the concepts are also represented by elemental nodes in the network, yet the connections between nodes are determined by the relationships asserted by propositions, and it is labeled with different lexicon. The connections utilized in propositional network then can represent more than one kind of relationships. The information that can be provided by the propositional network is also revealed in its structure. The semantics of the specific labeled relationships correlates individual concepts to factual propositions. And the related propositions are maintained and organized in the network adjacent to each others. Using chaining operations both forward and backward, the most relevant factual propositions could be discovered from the semantic networks. Additionally, in a semantic network, the theory of conceptual categories can also be built as one kind of labeled relationship. In consequence, the information can be revealed by concepts categorization is usually contained in a knowledge organization constructed with semantic network model.

Semantics will be inferred by agents with automated reasoning ability. In the articulation of the processes of human understanding in the previous section, it was stated that humans usually use reasoning to gain a deeper understanding of the discourse of others. By the same token, as the agents in the Semantic Web are required to perform as human proxies, it is necessary for them to maintain a similar level of understanding of Web content as a human might attain. Therefore, several automated reasoning abilities have been proposed by artificial intelligence experts to simulate similar reasoning principles to those that are normally practiced in the interpretation process. Generally, automated reasoning ability is provided by special modules of applications that are called inference engines. Inference engines use formal logic to support concluding inferences and induce new knowledge. To verify different reasoning processes with their different goals and contexts, logicians have developed many distinctive forms of logic. Illustrated in the following elaboration, three examples among others are picked to demonstrate the role of formal logics implemented in inference engine. First order propositional logic, for instance, deals with the most general requirements for the implementation of a rule-based inference engine. If a subtle prediction that should made base on several alternative possible results, then modal logic can be used to enact a computational reasoning process in the inference engine. Moreover, humans often make conclusions based on a locally available knowledge base. Sometime, as updated information results in a conclusion that conflicts with previous predictions, a human must judge between the conflicting predictions and revise the conclusion. This reasoning process could be simulated by the adoption of a non-monotonic logic in inference engines. More details on these different kinds of logic can be found in Baader (1999), Brewka (1991), Chellas (1980), and Ginsberg (1987).

Procedural semantics should be developed in the community for collaboration. The ultimate goal of the Semantic Web is to facilitate collaboration among humans and software agents. Speaking more specifically, the realization of the vision of the Semantic Web depends on the pragmatic materialization of automatic collaboration of an agent society. This means that an agent that is commissioned can autonomously search for help on the Web to gain resources, normally from its agent colleagues, to achieve its goal of solving certain problems. In this sense, the Semantic Web infrastructure provides a preliminary but fundamental infrastructure to identify the necessary resources. Unfortunately, standing in the way of this ultimate goal of automatic collaboration in an agent society is a problem that has puzzled artificial intelligence experts scientists for a long time—that of how to give software agents problem solving abilities. Problem solving ability has been modeled in the field of artificial intelligence as a series of processes that includes recognizing problem types, formulating a goal as a testable criterion, formulating the problem of what action to take and what states to consider in the description of the phenomenon, and then finding the path to guide the transition from the initial states to the final goal state (Russell & Norvig, 2003). Artificial intelligence scientists have proposed two approaches in finding an appropriate path to reach the goal: problem-solving agents and planning agents. Both agents may function well in a well-formulated problem in specific domains, but real-world problems are usually ill defined, and the analysis that is required to map potential problems into solvable models is usually lacking, which hinders these two solutions from being pragmatic proposals in the beginning stages.

However, the pressures from business demands often result in the speedy resolution of problems, and in this case instead of providing agents with the ability to compute a solution to a problem, some practitioners have suggested simply articulating all possible paths to the goal and maintaining them as templates. Once the problem types have been identified and matched with the premise of the template, then the template can be adopted. A typical example is the adoption of the workflow templates as a solution to generate collaboration process among participating parties for the enterprise application integration. The RossetaNet (www.rossetanet.org) , a nonprofit consortium being consisted of more than 500 organizations mainly in computer and consumer electronic industry, provides a directory of “Partner Interface Processes” as the instant access procedural knowledge for business collaboration. This comprehensive framework categorizes all possible business processes into seven clusters according to different roles of business operations of the supply chain. Obviously, these kinds of solutions must acquire consensus in a community before they are put to use. Within the community, procedural knowledge is then agreed upon in the form of standards or specifications that are publicly accessible by community members.

Semantic Web Technology

It should be evident from the previous sections that the development of Semantic Web technology has employed a lot of research results from cognitive psychology and artificial intelligence (Schwartz, 2003). In this emergent stage of Semantic Web technology, scientists in the artificial intelligence community seem to be the most engaged in the problem, and many research results on the implementation of the Semantic Web have been adapted from artificial intelligence techniques, such as knowledge representation and organization, planning and rule-based systems, machine learning in dealing with large scale ontology extraction, and ontology management for developmental evolution and comparison processes (Goble, 2003).

However simply perceiving the Semantic Web to be an extended application of artificial intelligence theory and technology is a misconception (Goble, 2003). First of all, research on Semantic Web development and artificial intelligence development differ in the goals that are to be achieved. Artificial intelligence research aims to help systems to think like humans and act like humans, or at least to think rationally and to act rationally (Russell & Norvig, 2003). In fact, this goal has been considerably reduced from the original goal, which was to create a machine with complete humanlike intelligence (Searle, 1980). However, this goal of creating a machine that could compete with the human brain was found to require not only an algorithm for the ability to reason, but also a huge amount of memory space. A successful example of the application of artificial intelligence was the defeat of the human world chess champion Gary Karsparov by the famous IBM project "Deep Blue" in May 1997 (www.research.ibm.com/deepblue/), but the achievement in this specific task took a enormous investment on the establishment of super computing ability. The investment in hardware is a 30-node RS/6000 SP-based computer system that was enhanced with 480 ASIC chips that specialized in playing chess. And the super computing ability can perform 100 million position evaluations per second. As a comparison, the milestones for the Semantic Web are not founded on the breakthrough of reasoning using a huge information space. Instead, the aim is to provide the efficient and effective automation of the comprehension of information and the provision of procedural integration among networked resources. From this perspective, machines in the future may well be able to defeat humans in the performance of iterative operations in terms of reliability and efficiency.

Secondly, the basic distinction of the Semantic Web is the Web itself. The Web has several distinct characteristics (Goble, 2003; Mendelson, 1997; Risvik & Michelsen, 2002) that serve to differentiate Web-oriented technology from other technology. As an

Table 1. Basic characteristics of the Web and its challenge to Semantic Web technology (Summarized from Goble, 2003)

Basic characteristics	Challenge to the researcher
The Web is vast	Solutions should be able to scale the large Web resource universe.
The Web already exists	Solutions should consider migrating the existing non-semantic Web to the envisioned semantic Web.
The Web is democratic	Solutions should cater to the demands of the full range of information users and creators.
The Web grows from the bottom up	Solutions should consider the nature of the Web architecture, which is better suited to bottom-up development and peer-to-peer structure than top-down development and a centralized structure.
The Web is volatile and changeable	The availability of online resources is uncertain, and the resource content will change frequently.
The Web is dirty	It is difficult to ensure the consistency, trustworthiness, and provenance of the information on the Web.
The Web is heterogeneous	There will be multiple solutions, ontologies, and conceptual models to agree on before translation to a key practice.

elaboration of the distinctive features of the Web environment, Goble provided a framework (please refer to Table 1) to categorize the challenges to researchers who are interested in deploying Semantic Web technology (Goble, 2003).

The Development of the Semantic Web

Initial attention to the concept of a Semantic Web can be dated to 2000 (Fensel, Lassila, et al., 2000). In February 2004, the first Semantic Web language specification, RDF schema 1.0 (www.w3.org/TR/rdf-schema/), was accepted as a stable version of the World Wide Web Consortium's (W3C) recommendations (www.w3.org/2004/02/Process-20040205/) for industrial usage. Judging from this, Semantic Web technology can be considered to be at the emergent stage.

In the following discussion, I introduce the development of Semantic Web technology, in which the technology is separated into three main functions: language, infrastructure, and application. Semantic Web language provides a group of Web-specific languages for the presentation of knowledge and the building of a knowledge organization. Semantic Web infrastructure aids the acquisition, development, and management of the knowledge organization, and Semantic Web applications reflect efforts to leverage the infrastructure for better collaboration in future society, which involves contributions from both humans and software agents.

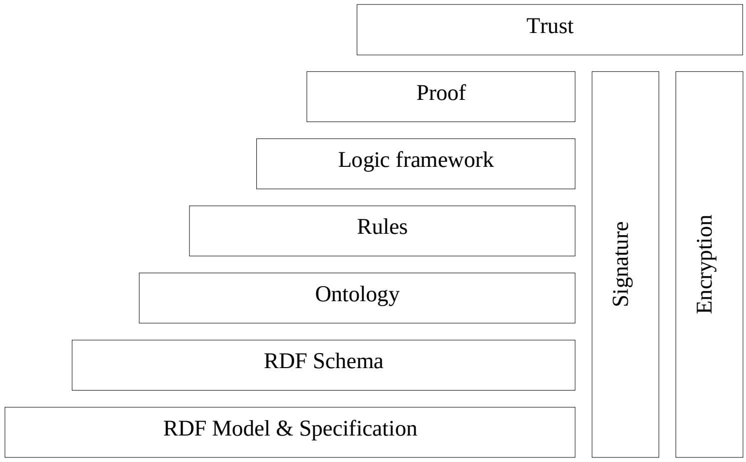
The development of a formal logic is also of importance to Semantic Web technology, as it will provide the formalism for knowledge representation and the rational reasoning for the inference machine. However, the development of formal logic is more related to the research of logicians and therefore in the following sections this topic is not covered.

Semantic Web Language

Semantic Web language provides two basic utilities for the representation and organization of knowledge. First, it provides a neutral language for knowledge representation that enables the construction of a portable knowledge organization, and second, it provides an annotation system to denote the relation of an expression in Web content to a pre-specified knowledge organization to enrich the semantic capacity of the expression. A new knowledge representation language has had to be developed for the Semantic Web because of the distinct nature of the Web. The Web is heterogeneous and democratic, and therefore we need a neutral language system to enhance its compatibility. Knowledge organizations that are built using Semantic Web language can be published over the World Wide Web, easily transported to the Internet, and recognized and reused by interested parties, which is why they are known as portable.

As Semantic Web language is a fresh descriptive foundation for existing Web expression, it basically makes use of extensible markup language (XML) to specify a new

Figure 1. Semantic Web layer cake (Adapted from Berners-Lee, 2000)

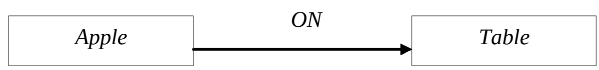


labeling schema. The annotation of discourse by the labeling system serves as the bridge between the discourse and the referring meaning in the knowledge organization. In this way, the meanings of the key words used in the documents are enriched by the recognition of the referring meaning in the knowledge organization.

Using XML as the specification language provides another benefit to the development of Semantic Web language. As the World Wide Web evolves with the advancement of the Web technology, the legacy Web must be migrated to the new infrastructure. It is also expected in the way we represent semantics of the Web content. As new technologies or new conceptual constructs are introduced to provide new capability for semantic interpretation, the new Semantic Web language will be established by the provision of new XML schemas. And the newly established XML schema is able to contain existing schemas as its foundation. In this way, the evolution of the Semantic Web language extends the semantic capacity by adding a new layer to the existing foundation stacks. Thus, the original layer of Semantic Web language will not become obsolete as concepts evolve.

In explaining the language advantage and developing the roadmap for Semantic Web languages, Tim Berners-Lee has proposed a “Semantic Web layer cake” of the expected standards and technologies. In the lower half of the cake, the resource definition framework (RDF) model and specification (www.w3.org/TR/rdf-concepts/) and RDF schema (RDFS) (www.w3.org/TR/rdf-schema/) represent two specifications that collaboratively work together to express propositions. Above this layer is an ontology layer, the corresponding language of which, ontology Web language (OWL) (www.w3.org/TR/owl-features/), enables the construction of a kind of categorical organization of concepts, or ontology. W3C has already proposed recommendation standards for these

Figure 2. An RDF graph describing the relationship "On (Apple, Table)"



three layers. Base on these layers, academia and industry are still working on proposing clear-cut executive implications and corresponding language standards.

In the following sections, I elaborate some of the basic concepts that are related to Semantic Web language.

Resource description framework (RDF). The RDF is a data model for the presentation of propositional relationships in a graph-like illustration. The most atomic relationship is a *Triple*, which relates a *Subject* to an *Object* through a directed relationship, which is described by a *Predicate*. The symbolic representation of the *Triple* relationship can simply be *Predicate (Subject, Object)* (please refer to Figure 2 for an example of an RDF graph).

This simple *Triple* representation can be used to describe any proposition that involves tangible or intangible entities in the real world as a kind of resource that is described on the World Wide Web. For example, the preposition "*the Apple is ON a Table*" can be represented as *ON(Apple, Table)*. To maintain relationships among more than three elements, the presentation allows a null resource assignment (expressed as _), and therefore the *Triple* can be extended by chaining elements via null elements. For example, *Predicate1(Subject, _), Predicate2(_, Object1), …, Predicate2(_, ObjectN)*. In this way, a set of propositions that is composed of a long chain can be employed to represent a complicated expression.

Ontology. An ontology is the one of the most widely used way of representing knowledge organizations. An ontology provides shared and common domain propositions and axioms to rule out ambiguity, while conducting communication and reasoning processes (Ding, 2001). As the Web is heterogeneous, democratic, vast, volatile, and changeable, an ontology is one of the most important components for the integration and reconciliation of ambiguity, and research on the applications of ontology engineering has become a core academic interest of the Semantic Web technology (Geller, Perl, & Lee, 2004; Ouksel & Sheth, 1999; Staab, Gomez-Perez, Daelemana, Reinberger, & Noy, 2004).

Ontology is an ancient theory that dates back to ancient Greek philosophy, but the term was later adopted by computer scientists in 1991 by Neches et al. (Corcho, Fernandez-Lopez, & Gomez-Perez, 2003; Neches et al., 1991). The knowledge representation community that deals with ontology includes various disciplines, such as knowledge engineering, knowledge representation, qualitative modeling, language engineering, database design, information retrieval and extraction, and knowledge management and organization (Guarino, 1998). Since ontology caught the attention of computer scientists, its conceptualization and definition have gone through many iterations as scholars from different disciplines have used different approaches. The history and research of the various approaches can be found in the literature (Crow & Shadbolt, 2001; Ding, 2001; Gomez-Perez, Fernandez-Lopez, & Corcho, 2004; Guarino, 1997). Here, I use the most commonly cited studies in the knowledge representation community and focus on the

Semantic Web in looking at ontology. The following definition of ontology combines the ideas of Gruber (1993), Borst (1997), Corcho et al. (2003) and Studer, Benjamins, and Fensel (1998): "Ontology is a formal, explicit specification of a shared conceptualization." "Conceptualization" refers to the abstract model by which the phenomena in the real world are represented (Studer et al., 1998). *Explicit* here means an explicit definition of the types of concepts, and constraints on the usage of these types should be provided. *Formal* refers to the features of the specification that serve to exclude ambiguity in communication between machines. *Shared* refers to the fact that once parties adopt a specific ontology, they all agree to conceptualize phenomena in the same way.

In terms of Semantic Web technology, an ontology can be considered as a typical form of implementation of a knowledge organization. It is a taxonomy for a set of concepts that includes concepts, categories, and the relationships between them, as well as a set of inference rules for the engagement of useful reasoning functions (Berners-Lee et al., 2001; Ding, 2001). Ontology developers use their own views to develop their own ontologies according their preferred goals and favored development methodologies, and this wide variety of perceptions and methodologies leads to a rather volatile research environment. General examples of ontology include the Yahoo directory classification, or enterprise ontology (Uschold et al., 1998), in which the taxonomy consists only of a lexicon and the semantics of the taxonomy depends on human interpretation. More complex taxonomies involve concept tokens that have a higher semantic capacity, which can be created by assigning concept tokens with described or defined features and then relating the separate concepts to propositional relationships. The cost of improving the usage of semantics in an ontology is paid off by the fact that other parties with similar interests can share and reuse the ontology (Gomez-Perez et al., 2004). Moreover, the domains and functions of ontological taxonomies also differ. A discussion of the different kinds of classifications for various kinds of ontologies can be found in Ding (2001), Fensel, Hendler, Lieberman and Wahlster (2003), Gomez-Perez et al. (2004), and Jurisica, Mylopoulos, and Yu (2004).

W3C provides an ontological description language, OWL, to cater to the demands for managing ontologies over the World Wide Web. The goal of the OWL specification is to facilitate better ontology management for the reuse, change, integration, and detection of inconsistency in ontologies (Heflin, 2004). The taxonomy that is offered in OWL provides the ability to assert basic logical descriptions by sanctioning reasoning against description logic that refers to the concept that describes a domain. Description logic is developed to represent terminological knowledge and concept classification (Baader, 1999). The utility of description logic seems to properly match the need for ontological development in the Semantic Web paradigm. Therefore, it is natural to leverage software agents that can infer additional assertions on the back of stated assertions and ontologies. The basic concept in the utilization of description logic is to develop subsumption relationships in a list of terms. Therefore, inference engines that reason using description logic provide efficient computing power to assert additional propositions from a repository of facts and an ontology. In addition to the efforts of W3C, other ontology languages have also been proposed and used in different contexts, tools, and applications for various purposes. Nevertheless, they have not been proposed for use in a Web context. Details can be found in Corcho et al. (2003) and Ding, Fensel, Klein, and Omelayenko (2002).

The Semantic Web Infrastructure

The Semantic Web infrastructure refers to the facilities, services and installation for transferring current Web content into the forms that the software agents are able to understand. It is done by adding formal explicit semantic, which is in terms of metadata and knowledge organization, to the Web contents. In this perspective, the services or facilities to provide Semantic Web infrastructure can be divided in three major clusters: (1) the services to transfer the discourse formatted in natural language into data and proposition forms, complying to Semantic Web languages specification, (2) the services to provide large scale availability of annotation within Web contents to correlate the metadata of the knowledge organization to the occurrence of the key words in Web content, (3) and the services to establish the metadata and knowledge organization to a scale of critical mass so that it can support enough information for semantic interpretation. The study of the first clusters is crucial to the enablers of the ability for software agent to understand meaning of Web content commonly formatted in natural language today, yet the research efforts are mainly related to the works done by those researchers who are dedicated in natural language processing. Therefore, the elaboration of this subject is considered as irrelevant to the scope of my discussion. In stead, the elaboration on major academic efforts on providing Semantic Web infrastructure focus on the following two clusters of infrastructure.

The efforts done to provide services of second cluster can be generally referred to "semantic annotation", which is assisted by tools to annotate Web pages or RDF documents with semantic tags that reference to ontology or other forms of knowledge organization. Typically three approaches can be found in literature to provide cluster two services. The first approach provides tools to link ontology with structural data. It is done by the mapping between ontology and the data schema of the structural data. The second approach uses an existing ontology to describe semi-structured Web content, and a corresponding XML DTD, XML Schema, or RDFS document is then generated as a reference for the validation of the Web content and to provide hints as to the links between the ontology and the Web content. The third approach converts totally unstructured content into a format that conforms to RDF/XML specifications and generated XML schema referencing to existing ontology (Dill et al., 2003; Kiryakov, Popov, Terziev, Manov, & Ognyanoff, 2004). There are several tools that can found in literature to semantic annotation services, such as Protégé-2000 (Noy et al., 2001), OntoAnnotate (Hartmann, Staab, & Ciravegna, 2002), Annotea (Kahan & Koivunen, 2001), SHOE (Heflin & Hendler, 2000), and SemTag (Dill et al., 2003) among others. However, the task assisted by the state of the art still takes much human effort, and the annotation results are prone to containing errors (Erdmann, Maedche, Schnurr, & Staab, 2000).

Current research efforts on the cluster three services focus on the development and management of large scale ontologies. It is elaborated here in two sub-subjects: one is Ontology editing and generation, and the other is Ontology engineering.

Ontology editing and generation. An ontology is either generated by human experts with the assistance of ontology editors, or is automatically established by an ontology learning mechanism. An ontology editor such as Protégé 2000, which was developed by

Stanford Medical Informatics (SMI), assists experts to build ontologies manually (Noy et al., 2001). Ontology learning is the ability to automatically learn and validate the ontology of a domain from groups of Web pages (Maedche & Staab, 2001; Missikoff, Navigli, & Velardi, 2002), and generated ontologies are managed and queried in an ontology repository. Further details of this function are available in the study of Ding (2002).

Ontology engineering. As the knowledge in an ontology is generated by various parties, it is not unified. Moreover, the perception of an ontology will change frequently to conform to the dynamic features of the World Wide Web. Therefore, ontology management, which is also called ontology engineering, is faced with the problem of how to align, merge and coordinate the differences among ontologies and how to manage the evolution issue in ontologies.

Research on ontology engineering predates the emergence of the Semantic Web (Ding, 2001; Ding & Foo, 2002), and actual Semantic Web ontology research was essentially launched after the proposed ontology languages for the Semantic Web became widely accepted and applied to research (Broekstra et al., 2002; Fensel, van Harmelen, Horrocks, McGuinness, & Patel-Schneider, 2001; Gomez-Perez & Corcho, 2002). In the Semantic Web scenario, ontology engineering must confront new challenges that result from the natural features of the Web, in which task-relative ontologies are established by different autonomous organizations. Therefore, ontology engineers need to tackle the problems that arise from the operation of ontologies when they merge (Cruz & Rajendran, 2003; Michalowski et al., 2004), translation and mapping between ontologies (Martin & Azvine, 2003), the evaluation and selection of an appropriate ontology (Lozano-Tello & Gomez-Perez, 2004; Staab et al., 2004), the evolution of ontologies that adapt to changes of environment, and the management of different versions of the same ontology that are created over time (Noy & Musen, 2004).

Applications of the Semantic Web

Since the emergence of Semantic Web research, scholars from various disciplines have tried to apply Semantic Web technologies to many areas. These applications can be classed into two categories. The first category simply uses portions of Semantic Web technology to provide new services over the World Wide Web, some typical examples being calendar agents (Payne, Singh, & Sycara, 2002), which surf several Web pages to negotiate and schedule a satisfactory meeting; agenda; shopping agents (Guarino, 1997), which shop online for goods by the semantic meanings of product information, rather than by keywords; and ontology-based searches on a digital map (Hubner, Spittel, Visser, & Vogele, 2004).

The applications in the second category adopt Semantic Web technology as an enabler to extend the new vision of the Semantic Web to old application areas, some typical examples of which are knowledge management and Web services. In the following section, I illustrate more specifically how Semantic Web technology is applied in these two areas in this category.

Knowledge management. Knowledge management involves the acquisition, extraction, management, and dissemination of human knowledge in explicit forms. Recent research and practice on knowledge management has focused on the extraction of knowledge from Web pages (Ding et al., 2002; Fensel et al., 2000). As all of the content on the World Wide Web can be transformed into machine-readable propositions and knowledge organizations, the Semantic Web itself will be the biggest repository of knowledge in the world. Researchers who are interested in making this vision feasible have identified three major achievable components: a query interface to assist humans to formulate their expectations of the knowledge that they want to extract from the World Wide Web as a proper query, a Webcrawler to navigate the World Wide Web to collect the required information, and an inference engine to extract the relevant knowledge from the collected information (Ding et al., 2002; Fensel et al., 2000).

Web services. A Web service in its broadest sense is an e-Service synonym that refers to concepts that use an information system as the service performer or mediating channel to reach the consumers of the service (Hultgren & Eriksson, 2005). According to this loose definition, an online banking service or gateway for providing online news can be considered to be a Web service. However, there is now a consensus in the computer industry that a Web service should refer to a narrower view of service than e-Service. The narrower view describes a new operation paradigm in which the coarse granular functionality is performed by an encapsulated software component that can be accessed and invoked over the Internet by others. In the invocation process, service consumers need not know the details of the implementation procedures within the Web service, but must learn in advance the effect of the execution of the service (Chung, Lin, & Mathieu, 2003). Several recent articles (Curbera et al., 2002; Tsalgatidou & Pilioura, 2002; Wang, Huang, Qu, & Xie, 2004; Zhao & Cheng, 2005) and industry standards (W3C, 2004) have concurred that the Web service paradigm should comprise two features. First, the Web service should adopt the two standards that have been proposed by W3C, the Web services description language (WSDL) and simple object access protocol (SOAP). These two standards serve collaboratively to provide separate functions that are required to invoke a Web service. WSDL provides a formal, machine-readable description for service providers of the information that is necessary to invoke a service over the Internet. This includes the specification for accessing interface, accessing protocol, and endpoint. SOAP serves as a message exchange protocol through which collaborating parties convey messages over the Internet. Second, the Web service architecture should have an active strategy to aid service discovery in the World Wide Web universe. A registry service provider will stand between services seekers and providers to broker search enquiries. In actual practice, universal description, discovery, and integration (UDDI) servers will be the mediating performers. To make their services known to others, service providers will register related information on the UDDI servers, and all of the registration records will be maintained by the UDDI as in a Yellow Pages directory. Service requesters will go to the directory to find useful services, and once a service has been identified, the service requester will download specific WSDL documents for further usage.

Traditionally, computing resources are kept inside a boundary and cannot be reused easily by outsiders. Creating a messaging channel for resource exchange between application islands and providing remote access to computing resources consumes a great deal of human effort, and is deemed to be impractical. The Web service paradigm is therefore a promising foundation for process integration (Staab et al., 2003; Tsalgatidou & Pilioura, 2002), because of its higher interoperability, better protection of sensitive data, lower implementation cost, easier and faster deployment, and lower complexity in delivering the encapsulation of internal processes (Staab et al., 2003; Zhang, Chung, & Chang, 2004). Thus, the development of the Web services paradigm is a vital component in the realization of the Semantic Web vision of better collaboration between human and software agents. The implementation of Semantic Web applications should benefit from the breakthrough in the Web services paradigm to automatically integrate software agents as typical processes in collaborations. The development of a Web services paradigm also takes advantage of Semantic Web technology. In a practical implementation scenario, Web services must collaborate with each other to deliver complicated tasks. Given the nature of Web-oriented resources and the human limitations on the size of datasets, there is a tendency to rely on intelligent software agents as performers in many aspects of the deployment lifecycle, some typical examples being service discovery, invocation, composition, analysis, cancellation, and monitoring processes. To ensure the successful operation of software agents, Web services, which are important resources in the Semantic Web, must be represented by machine-readable semantics that are relevant to the whole deployment cycle. Research in this area is deemed to be the foundation of Semantic Web services (Paolucci & Sycara, 2003; Sycara, Paolucci, Ankolekar, & Srinivasan, 2003).

Currently, research on Semantic Web services is still in its emergent stage, as with Semantic Web technologies. A major research focus is the Web service language with more semantic descriptive capacity, such as WS-CDL (W3C, 2004), and OWL-S (www.daml.org/services/owl-s/1.1/overview/), and workflow language specific catering to the description of the Web services composition in the workflow, such as WS-BPEL (www.oasis.org). Other major areas of research are the discovery of sophisticated Web services in the resources universe (Klein & Visser, 2004; Paolucci & Sycara, 2003; Sreenath & Singh, 2004), and the development of a framework for Web service composition to establish and implement complicated service organizations (Benatallah, Dumas, & Sheng, 2005; Benatallah, Sheng, & Dumas, 2003; Medjahed & Bouguettaya, 2005; Sivashanmugam, Miller, Sheth, & Verma, 2003).

As both the Semantic Web and Web services are part of the same vision, they will evolve together to create a society that is composed of two equally contributing groups of citizens: humans and software agents.

Conclusion

This chapter describes the semantics for the Semantic Web as a type of explicitly declared machine-understandable semantics. The processes for the implementation of this type of semantics involve four steps. The semantics should be explicitly declared in a form that can be operated by a machine, knowledge organizations should be provided to support semantic interpretation, software agents must use automatic reasoning to obtain implicit knowledge, and procedural knowledge should be accessed in a community to generate response behavior. An inspection of the current progress of the development of the Semantic Web reveals that many enablers that are necessary to make the Semantic Web work are still missing. The Semantic Web languages that have been developed thus far only cater to the demand in describing instances and classes in a common knowledge organization. Researchers and practitioners are still debating the proper form that the language for higher level semantics should take, such as rules, proofs, and security. Most of the existing enablers for the establishment of a comprehensive Semantic Web infrastructure are lacking, and there is no consensus on a proper mechanism for large scale semantic annotation. The heterogeneity that commonly exists in the development methodology, schema, and conceptual models for the establishment of distributed ontologies also requires further development. The nascent status of Semantic Web technology has made it difficult to develop software agents that will enable automatic collaboration. Indeed, it is hard to predict at this point whether the Semantic Web will be successfully achieved in the future. However, once the limitations of the current World Wide Web become clear and the utility of the World Wide Web extends as it moves from cable connection to wireless connection, increasing investment will be made to forge the Semantic Web.

References

Baader, F. (1999). Logic-based knowledge representation. In M. J. Wooldridge & M. Veloso (Eds.), *Artificial intelligence today* (Vol. LNAI 1600, pp. 13-41). Berlin Heidelberg: Springer-Verlag.

Benatallah, B., Dumas, M., & Sheng, Q. Z. (2005). Facilitating the rapid development and scalable orchestration of composite Web services. *Distributed and Parallel Databases, 17*, 5-37.

Benatallah, B., Sheng, Q. Z., & Dumas, M. (2003). The self-serve environment for Web services composition. *IEEE Internet Computing, 7*(1), 40-48.

Berners-Lee, T. (2000). *RDF and the semantic Web*. Paper presented at the Presentation at XML2000 Conference. Retrieved June 27, 2005, from http://www.w3.org/2002/talks/04-sWeb/slide12-0.html

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American, 284*(5), 34-43.

Borst, W. N. (1997). *Construction of engineering ontologies*. Unpublished PhD thesis, University of Tweenty, Enschede, NL.

Brewka, G. (1991). *Nonmonotonic reasoning*. Cambridge, UK: Cambridge University Press.

Broekstra, J., Klein, M., Decker, S., Fensel, D., van Harmelen, F., & Horrocks, I. (2002). Enabling knowledge representation on the Web by extending RDF schema. *Computer Networks, 39*(5), 609-634.

Chellas, B. F. (1980). *Modal logic: An introduction*. Cambridge, UK: Cambridge University Press.

Chung, J.-Y., Lin, K.-J., & Mathieu, R. G. (2003). Web services computing: advancing software interoperability. *Computer, 36*(10), 35-37.

Collins, A. M., & Quillian, R. M. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior, 8*, 240-248.

Corcho, O., Fernandez-Lopez, M., & Gomez-Perez, A. (2003). Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data & Knowledge Engineering, 46*, 41-64.

Croft, D. W. (1997). *Intelligent software agents: Definitions and applications*. Retrieved from http://alumus.caltech.edu/~croft/research/agent/definition/

Crow, L., & Shadbolt, N. (2001). Extracting focused knowledge from the Semantic Web. *International Journal of Human-Computer Studies, 54*(1), 155-184.

Cruz, I. F., & Rajendran, A. (2003). Semantic data integration in hierarchical domains. *IEEE Intelligent Systems, 18*(2), 66.

Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., & Weerawarana, S. (2002). Unraveling the Web services Web: an introduction to SOAP, WSDL, and UDDI. *Internet Computing, IEEE, 6*(2), 86-93.

Dill, S., Eiron, N., Gibson, D., Gruhl, D., Guha, R., & Jhingran, A., et al. (2003). A case for automated large-scale semantic annotation. *Web Semantics: Science, Services and Agents on the World Wide Web, 1*(1), 115-132.

Ding, Y. (2001). A review of ontologies with the Semantic Web in view. *Journal of Information Science, 27*(6), 377.

Ding, Y., Fensel, D., Klein, M., & Omelayenko, B. (2002). The Semantic Web: Yet another hip? *Data & Knowledge Engineering, 41*(2-3), 205-227.

Ding, Y., & Foo, S. (2002). Ontology research and development. Part I—A review of ontology generation. *Journal of Information Science, 28*(2), 123.

Du, T. C., Li, E. Y., & Wei, E. (2005). Mobile agents for brokering services in the electronic marketplace. *Decision Support Systems, 39*(3), 371-383.

Erdmann, M., Maedche, A., Schnurr, H.-P., & Staab, S. (2000, August). *From manual to semi-automatic semantic annotation: About ontology-based text annotation tools*. Paper presented at the COLING 2000 Workshop on Semantic Annotation and Intelligent Content, Saarbruecken, Germany.

Fensel, D., Decker, S., Erdmann, M., Schnurr, H.-P., Studer, R., & Witt, A. (2000). Lessons learned from applying AI to the Web. *Journal of Cooperative Information Systems, 9*(4).

Fensel, D., Hendler, J., Lieberman, H., & Wahlster, W. (Eds). (2003). *Spinning the semantic Web: bringing the World Wide Web to its full potential*. Cambridge, MA: MIT Press.

Fensel, D., van Harmelen, F., Horrocks, I., McGuinness, D. L., & Patel-Schneider, P. F. (2001). OIL: An ontology infrastructure for the semantic Web. *IEEE Intelligent Systems, 16*(2), 38.

Fensel, D., Lassila, O., van Harmelen, F., & Horrocks, I., et al., (2000). The semantic Web and its languages. *IEEE Intelligent Systems & their Applications, 15*(6), 67.

Fuggetta, A., Picco, G. P., & Vigna, G. (1998). Understanding code mobility. *IEEE Transactions on Software Engineering, 24,* 342-361.

Geller, J., Perl, Y., & Lee, J. (2004). Editorial: Ontology challenges: A thumbnail historical perspective. *Knowledge and Information Systems*, *6*(4), 375-379.

Ginsberg, M. L. (Ed.). (1987). *Reading in nonmonotonic reading*. San Francisco: Morgan Kaufmann.

Goble, C. (2003). The semantic Web: An evolution for a revolution. *Computer Networks, 42*(5), 551.

Gomez-Perez, A., & Corcho, O. (2002). Ontology languages for the Semantic Web. *IEEE Intelligent Systems, 17*(1), 54.

Gomez-Perez, A., Fernandez-Lopez, M., & Corcho, O. (2004). *Ontological engineering: With examples from the areas of knowledge management, e-commerce and the semantic Web*. London; New York: Springer-Verlag.

Grosof, B. N., & Poon, T. C. (2004). SweetDeal: Representing agent contracts with exceptions using semantic Web rules, ontologies, and process descriptions. *International Journal of Electronic Commerce, 8*(4), 61.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition, 5*, 199-220.

Guarino, N. (1997). Understanding, building and using ontologies. *International Journal of Human-Computer Studies, 46*(2/3), 293-310.

Guarino, N. (1998, June). *Formal ontology and information systems.* Paper presented at the Formal Ontology in Information Systems, Trento, Italy.

Guarino, N., Carrara, M., & Giaretta, P. (1994). *An ontology of meta-level categories.* Paper presented at the Principles of Knowledge Representation and Reasoning: Proceeding of the 4th International Conference (KR94). San Francisco: Morgan Kaufmann.

Handschuh, S., Staab, S., & Ciravegna, F. (2002). *S-CREAM—Semi-automatic creation of metadata.* Paper presented at the 13th International Conference on Knowledge Engineering and Management (EKAW 2002), Siguenza, Spain.

Hartmann, J., & Sure, Y. (2004). An infrastructure for scalable, reliable semantic portals. *IEEE Intelligent Systems, 19*(3), 58.

Heflin, J. (2004). *Web ontology language (OWL) use cases and requirements*. Retrieved from http://www.w3.org/TR/Webont-req/: W3C

Heflin, J., & Hendler, J. (2000). *Searching the Web with shoe*. Paper presented at the AAAI-2000 Workshop on AI for Web search, Austin, Texas.

Hendler, J. (2001). Special issue on agents and the Semantic Web. *IEEE Intelligent Systems, 16*(2), 30-37.

Hendler, J. (2003). Science and the Semantic Web. *Science*, *299*(5606), 520.

Hubner, S., Spittel, R., Visser, U., & Vogele, T. J. (2004). Ontology-based search for interactive digital maps. *IEEE Intelligent Systems, 19*(3), 80.

Hultgren, G., & Eriksson, O. (2005, March 15-16). *The concept of e-Service from a social interaction perspective*. Paper presented at the ALOIS 2005, Limerick, Ireland.

Johnson-Laird, P. N. (2000). Thinking: Reading. In A. E. Kazdin (Ed.), *Encyclopedia of psychology* (Vol. 8, pp. 75-79). Washington, DC: American Psychological Association.

Jurisica, I., Mylopoulos, J., & Yu, E. (2004). Ontologies for knowledge management: An information systems perspective. *Knowledge and Information Systems, 6*, 380-401.

Kahan, J., & Koivunen, M.-R. (2001). A*nnotea: An open RDF infrastructure for shared Web annotations*. Paper presented at the World Wide Web.

Kiryakov, A., Popov, B., Terziev, I., Manov, D., & Ognyanoff, D. (2004). Semantic annotation, indexing, and retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web, 2*(1), 49-79.

Klein, M., & Visser, U. (2004). Semantic Web challenge 2003. *IEEE Intelligent Systems, 19*(3), 31-33.

Lozano-Tello, A., & Gomez-Perez, A. (2004). ONTOMETRIC: A method to choose the appropriate ontology. *Journal of Database Management, 15*(2), 1.

Maedche, A., & Staab, S. (2001). Ontology learning for the Semantic Web. *IEEE Intelligent Systems, 16*(2), 72.

Martin, T. P., & Azvine, B. (2003). Acquisition of soft taxonomies for intelligent personal hierarchies and the soft Semantic Web. *BT Technology Journal, 21*(4), 113.

McDowell, L., Etzioni, O., & Halevy, A. (2004). Semantic email: theory and applications. *Web Semantics: Science, Services and Agents on the World Wide Web, 2*(2), 153-183.

Medjahed, B., & Bouguettaya, A. (2005). A dynamic foundation architecture for Semantic Web services. *Distributed and Parallel Databases, 17*, 179-206.

Mendelson, E. (1997). *Introduction to mathematical logic* (4th ed.). London: Chapman.

Michalowski, M., Ambite, J. L., Thakkar, S., Tuchinda, R., Knoblock, C. A., & Minton, S. (2004). Retrieving and semantically integrating heterogeneous data from the Web. *Intelligent Systems, IEEE* (see also IEEE Expert), *19*(3), 72-79.

Missikoff, M., Navigli, R., & Velardi, P. (2002). Integrated approach to Web ontology learning and engineering. *Computer, 35*(11), 60-63.

Neches, R., Fikes, R. E., Finin, T., Gruber, T. R., Senator, T., & Swartout, W. R. (1991). Enabling technology for knowledge sharing. *AI Magazine, 12*(3), 36-56.

Noy, N. F., & Musen, M. A. (2004). Ontology versioning in an ontology management framework. *Intelligent Systems, IEEE* (see also IEEE Expert), *19*(4), 6-13.

Noy, N. F., Sintek, M., Decker, S., & Crubezy, M., et al. (2001). Creating Semantic Web content with Protege-2000. *IEEE Intelligent Systems, 16*(2), 60.

Ouksel, A. M., & Sheth, A. (1999). A brief introduction to the research area and the special section. *SIGMOD Record* (Special Section on Semantic Interoperability in Global Information Systems)*, 28*(1), 5-12.

Paolucci, M., & Sycara, K. (2003). Autonomous Semantic Web services. *IEEE Internet Computing, 7*(5), 34-41.

Payne, T. R., Singh, R., & Sycara, K. (2002). Calendar agents on the Semantic Web. *IEEE Intelligent Systems, 17*(3), 84.

Risvik, K. M., & Michelsen, R. (2002). Search engines and Web dynamics. *Computer Networks, 39*(3), 289-302.

Russell, S., & Norvig, P. (2003). *Artificial intelligence: A modern approach* (2nd ed.). Upper Saddle River, NJ: Prentice Hall.

Schwartz, D. G. (2003). From open IS semantics to the Semantic Web: The road ahead. *IEEE Intelligent Systems, 18*(3), 52.

Searle, J. R. (1980). Minds, brains, and programs. *Behavioral and Brain Sciences, 3*(3), 417-457.

Sheth, A. (2005). Semantics for the Semantic Web: The implicit, the formal, and the powerful. *International Journal on Semantic Web & Information Systems, 1*(1), 1-18.

Sivashanmugam, K., Miller, J. A., Sheth, A. P., & Verma, K. (2003). *Framework for Semantic Web process composition* (No. Technical Report 03-008): Large Scale Distributed Information System Lab, Dept of Computer Science, UGA.

Smith, I. A., Cohen, P. R., Bradshaw, J. M., Greaves, M., & Holmback, H. (1998, July 3-7). *Designing conversation policies using joint intention theory.* Paper presented at the 3rd International Conference on Multiagent Systems (ICMAS-98), Paris.

Sreenath, R. M., & Singh, M. P. (2004). Agent-based service selection. *Web Semantics: Science, Services and Agents on the World Wide Web, 1*(3), 261-279.

Staab, S., Aalst, W. V. D., Benjamins, V. R., Sheth, A., Miller, J. A., & Bussler, C., et al. (2003). Web services: been there, done that? *IEEE Intelligent Systems, 18*(1), 72-85.

Staab, S., Gomez-Perez, A., Daelemana, W., Reinberger, M.-L., & Noy, N.F. (2004). Why evaluate ontology technologies? Because it works! *Intelligent Systems, IEEE* (see also IEEE Expert), *19*(4), 74-81.

Staab, S., Santini, S., Nack, F., Steels, L., & Maedche, A. (2002). Emergent semantics. *Intelligent Systems, IEEE* (see also IEEE Expert), *17*(1), 78-86.

Sternberg, R. J. (2003). *Cognitive psychology* (3rd ed.). Thomson Wadsworth.

Stork, D. G. *The end of an era, the beginning of another? HAL, Deep Blue and Kasparov*. Retrieved June 25, 2005, from http://www.research.ibm.com/deepblue/learn/html/e.8.1.shtml

Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge engineering: principles and methods. *Data & Knowledge Engineering, 25*, 161-197.

Su, Z., Jiang, J., Liu, T., Xie, G. T., & Pan, Y. (2004). Market intelligence portal: An entity-based system for managing market intelligence. *IBM Systems Journal, 43*(3), 534.

Sycara, K., Paolucci, M., Ankolekar, A., & Srinivasan, N. (2003). Automated discovery, interaction and composition of Semantic Web services. *Web Semantics: Science, Services and Agents on the World Wide Web, 1*(1), 27-46.

Trastour, D., Bartolini, C., & Preist, C. (2003). Semantic Web support for the business-to-business e-commerce pre-contractual lifecycle. *Computer Networks, 42*(5), 661.

Tsalgatidou, A., & Pilioura, T. (2002). An overview of standards and related technology in Web services. *Distributed and Parallel Databases*, *12*, 135-162.

Uschold, M. (2003). Where are the semantics in the Semantic Web? *AI Magazine, 24*(3), 25-36.

Uschold, M., King, M., Moralee, S., & Zorgios, Y. (1998). The enterprise ontology. *The Knowledge Engineering Review*, *13*, 31-89.

Varile, G. B., & Zampolli, A. (Eds.). (1997). *Survey of the state of the art in human language technology*. Pisa: Giardini.

W3C. (2004). *Web services architecture requirements*. W3C Working Group Notes.

Wang, H., Huang, J. Z., Qu, Y., & Xie, J. (2004). Web services: Problems and future directions. *Web Semantics: Science, Services and Agents on the World Wide Web*, *1*(3), 309-320.

Wason, P. C., & Johnson-Laird, P. N. (1972). *Psychology of reasoning: Structure and content*. Cambridge, MA: Harvard University Press.

Williams, R. N. (2000). Epistemology. In A. E. Kazdin (Ed.), *Encyclopedia of psychology* (Vol. 3, pp. 225-232). Washington, DC: American Psychological Association.

Wittgenstein, L. (1974). *Tractatus logico-philosophicus* (Revised ed., D. F. Pears & B. F. McGuinness, Trans.). London: Routledge and Kegan Paul.

Zhang, J., Chung, J.-Y., & Chang, C. K. (2004). *Migration to Web services oriented architecture: A case study*. Paper presented at the ACM Symposium on Applied Computing.

Zhao, J. L., & Cheng, H. K. (2005). Web services and process management: A union of convenience or a new area of research? *Decision Support System*, *40*(1), 1-8.