Preface

A decade has passed since the term Semantic Web was first used in Tim Berners-Lee’s book “Weaving the Web.” It has also been a decade since this book’s editor founded Taalee, initially a Semantic Search company, fully knowing the power of modeling relationship as a first class model (as in RDF) and faceted search. The work presented at the 2000 keynote given at the first international event of Semantic Web (http://slidesha.re/sw-ib), in a patent filed in 2000 and awarded in 2001 (http://bit.ly/sw-p) and in an article (http://bit.ly/sw-ic) that describes the Semantic Web technology based application development platform as well as a variety of semantic search, browsing, personalization, interactive marketing (advertisement), and analysis applications for a variety of content within enterprises and on the Web. By the time you read this, it would also be a decade since the highly cited Scientific American article on Semantic Web by Tim Berners-Lee, James Hendler and Ora Lassila.

Some of the technologies, which involve minimal human training and experience to use, get adapted at a faster rate. One example is that of tablets. Another example is social networks. Both of these came about after Semantic Web was conceived of, and after RDF, the key underlying standard for Semantic Web data, was developed, and they were widely accepted within five years. However, technologies that require development of broader infrastructure, are complicated or have multiple components, require an ecosystem of trained programmers, best practices, and mature products and services, take at least a decade. I remember that it was 15 years after the first relational database management system came out that large corporations were seriously considering migrating their IMS and CODASYL based databases. In that context, Semantic Web’s adoption is quite reasonable.

At the core of it, Semantic Web technology involves three components. The first is development of domain/conceptual models or ontologies. These capture agreements regarding a domain of discourse--ranging from shared vocabulary to extensive factual knowledge. This component has seen extensive progress in a number of domains, with the domain of life sciences as perhaps the most prominent one. There are now over 300 ontologies in the repository maintained by the Nation Center of Biomedical Ontologies (bioprotal.org). There are also existing or emerging ontologies that are domain-independent or general purpose, including time, natural language, and provenance. Some, especially in the search community and those dealing with broad variety of content, argue that while ontologies can help with Enterprise content, they cannot scale to the Web. Others disagreed and believed it possible to develop simple models or identify relevant domain ontologies that are appropriate to interpreting and understanding content. You can find an example of the debate at http://bit.ly/s-search. The validation of the latter position came in the form of recent collaboration between the three Web search companies to create schema.org which provides schemas, or conceptual models, for several common domains.
The second component of Semantic Web approach is to semantically annotate any content. This essentially means adding semantic metadata where the semantics is conveyed through associating what is in the content with what is in described in the domain/conceptual model or ontology. For general web content that is described using HTML, better support for microdata and RDFa with HTML5 is making it easier to add or describe metadata. Followed by early progress in annotating documents found in enterprises and on the Web, there is now extensive growth in annotating social media data (such as Facebook’s Open Graph protocol) and e-commerce data (i.e., BestBuy’s use of the GoodRelations ontology). While my start up Taalee did semantic annotation of audio and video content over a decade ago, there was little progress until recently; an example can be found in discussions at a recent VW3C Video on the Web workshop (http://www.w3.org/2007/08/video/). As the Web 3.0 takes shape, the growth in the “Internet of the Thing” or “Web of the Thing”, including sensors data made accessible over the Web, is significantly outpacing that of textual content. Correspondingly we see the rise of Semantic Social Web, Semantic Sensor Web and other ecosystems emerging around new forms of largely non-textual data, each with its own form of techniques for annotating data, storing those annotations and accessing them. In 2010, W3C’s Semantic Sensor Networking has made significant progress in designing both a core ontology for Sensor, as well as in developing best practices for annotating sensor data.

Much of the initial Semantic Web data started with the annotations. However, parallel to the growth of conceptual models and ontologies, and their use for annotations, during last five year, we have seen even stronger growth of Semantic Web data in the form of Linked Open Data (LOD). This has involved representing unstructured data in the form of more structured data while capturing semantics-- such as representing Wikipedia as DBpedia. LOD has also started to become important knowledge as an anchor for semantics. Given that various sources of LOD are independent, although overlapping or related, there is an extensive need for semantic interoperability and integration needs. These are now being investigated in terms of robust sub-areas of ontology alignment.

In terms of the third component, analysis and reasoning, we have started to find ways to deal with large graphs - to find patterns, paths and subgraphs. There is a slow but steady progress in understanding and exploiting the power of relationships. An area that received concrete attention is that of the challenges in getting “same_as” right. Same_as has received the most attention in the context of LOD. This fundamentally gets to the issue of disambiguation and context- as things might be the same in one context, but not all.

This volume contains chapters organized in three sections. The first section is related to ontology development and applications primarily focused on using ontology for specific Semantic Web capabilities. The second section focuses on building semantic annotations and the tools that provide middleware and services that utilize ontologies and annotations. The third section includes chapters on semantic applications, starting with the use of semantics and Semantic Web technologies for improving retrieval, search, et cetera.

SECTION 1: ONTOLOGY DEVELOPMENT AND ONTOLOGY-BASED SERVICES

Section 1 begins with Chapter 1: Building Chemical Ontology for Semantic Web Using Substructures Created by Chem-BLAST, by Talapady N. Bhat. Having led the development of a few tens of ontologies, many of which are used in real-world applications, the author’s empirical observation is that ontologies developed to conceptualize and describe the natural world (e.g, life sciences) are usually lot
more complicated and intricate compared to ontologies to describe the domains, system and processes humans have created (e.g., sports, travel, entertainment). This chapter belongs to the first category, and primarily deals with biochemistry. It discusses an automated technique to create a structural ontology for compounds like ligands, co-factors and inhibitors of protein and DNA molecules using a technique developed from Perl scripts, which use a relational database for input and output, called Chem-BLAST (Chemical Block Layered Alignment of Substructure Technique). This technique recursively identifies substructures using rules that operate on the atomic connectivity of compounds. Substructures obtained from the compounds are compared to generate a data model expressed as triples. A chemical ontology of the substructures is made up of numerous interconnected ‘hubs-and-spokes’ is generated in the form of a data tree. This data-tree is used in a Web interface to allow users to zoom into compounds of interest by stepping through the hubs from the top to the bottom of the data-tree.

Chapter 2, A Pattern-Based Method for Re-Engineering Non-Ontological Resources into Ontologies, was written by by Boris Carmen Villazón-Terrazas, Mari Suárez-Figueroa, and Asunción Gómez-Pérez. Manually building ontologies that would involve humans to find and add each of the facts or component of its knowledgebase is not scalable. So a key to scalability is to find existing sources which are already based on ontological commitment, or consensus of the community of experts and users regarding the validity and interpretation of the facts that constitute the knowledgebase. If such sources are well structured, transforming or mapping them into the structure needed for ontological representation is easier, compared to the cases when unstructured sources. An even harder case to source ontological knowledge is presented by non-ontological resources (NORs), defined as “knowledge resources whose semantics have yet to be formalized by an ontology”. These NORs can be in a variety of heterogeneous forms, including textual corpora, classification schemes, thesauri, lexicon, and folksonomy. This chapter first characterizes NORs based on the type of inner organization of the resource, design data model used to represent the knowledge encoded by the resources, and resource implementation. The chapter then presents a comparative framework for re-engineering NORs. The core of the chapter presents a method that is based on so-called re-engineering patterns along with a software library that implements the transformations suggested by the patterns. The chapter also presents an evaluation framework.

Chapter 3, An Ontology-Based, Cross-Application Context Modeling and Management Service, was written by Annett Mitschick, Stefan Pietschmann, and Klaus Meißner. Context-awareness is important for both information providers (software agents, sensors, applications) and information consumers, including applications that use information in a specific context. While there is a body of work in context awareness and context-aware applications, this work takes the next step in building cross-application context management that provides context support for diverse applications, application plug-ins, software agents and sensors acting in the roles of producers and consumers. They use ontology-based approach to support what they term “domain profiles” to build their cross-application context management services, and demonstrate the use in three applications: personal multimedia document management, adaptive co-browsing and context-aware user interface mashups.

SECTION 2: ANNOTATION, MAPPINGS AND TOOLS

Section 2 begins with Chapter 4, Files are Siles: Extending File Systems with Semantic Annotations, by Bernhard Schandl and Bernhard Haslhofer. Nowadays, extensive applications of Semantic Web technologies have added to the challenges in Web and enterprise environments. One of the next frontiers, which
has already been started to be explored, is desktops, with an initial body of research in semantic desktops. Examples include incomplete information, broken links, or disruption of content and annotations. This chapter investigates a model authors call siles, which combine features of Semantic Web with files systems. The intent of this work is to show that siles model can provide a better infrastructure to build semantic desktops of the future. The chapter presents one component of this infrastructure, a virtual file system, which allows users to semantically annotate files and directories with RDF descriptors while keeping full compatibility to traditional hierarchical file systems. The chapter also provides the context of this work with respect to hierarchical file systems and metadata-centric file systems, and presents a prototype implementation and evaluates performance of typical access operations at file system and semantic metadata levels.

Chapter 5, Towards Controlled Natural Language for Semantic Annotation, was written by Brian Davis, Pradeep Dantuluri, and Hamish Cunningham. Semantic annotation is a core component of most Semantic Web approaches— it is how one uses controlled vocabulary or more formal representation as an ontology to associate meaning and common interpretations to words and objects on the Web. There are several approaches to achieve semantics annotations, including (a) manual annotation, semi-automatic annotation, and fully automatic annotations. Simple manual annotations found in unrestricted tagging is often of limited value due to little consistency and uncontrolled quality. Manual annotations, with respect to a formal ontology, can be time consuming as well as a complex job, especially if it involves the annotator to gain expertise in using a formal ontology. For a well-defined domain, which can have high quality ontology or ontologies when applied on certain high quality textual content, automatic annotations can give good and scalable results. A number of semi-automated approaches use natural language processing and/or machine learning techniques with ontologies described in a Semantic Web language (typically OWL). Such solutions require significant initial investment in building ontologies as well as sophistication required for developing the technical solution. This chapter focuses on empowering content creators with an approach that involves a lower learning curve and little initial investment. It advocates and demonstrates use of Controlled Natural Languages (CNLs) that are subsets of natural language whose grammar and dictionaries have been restricted in order to reduce or eliminate both ambiguity and complexity. The approach is demonstrated for meeting notes.

Chapter 6, A Tool Suite to Enable Web Designers, Web Application Developers and End-users to Handle Semantic Data, was written by Mariano Rico, Óscar Corcho, José Antonio Macías, and David Camacho. Web application development is no longer an easy task. A developer is faced with a large collection of technologies with significant increases in the complexity and novelty on the client side. While Semantic Web developers have often failed to pick up necessary skills to develop sophisticated but end user friendly interfaces, coercing traditional Web application developers to add Semantic Web technologies to their mix of skill is too burdensome for them. This work discusses a semantic template based strategy to provide Web designers ability to incorporate semantic data into their applications without getting too involved in Semantic Web technologies. This strategy seeks to provide collaborative features to create templates, allowing forms to enter user’s data that will be converted to semantic data and support semantic data transformation, all in a simpler albeit somewhat constrained ways compared to contemporary strategies. Evaluations include investigating how well it supports web designers in a wide range of competencies in client-side technologies, ranging from amateur HTML developers to professional Web designers.

Chapter 7, Adaptive Hybrid Semantic Selection of SAWSDL Services with SAWSDL-MX2, was written by Matthias Klusch, Patrick Kapahnke, and Ingo Zinnikus. Web services are a primary way
for making applications and tools available over the Web using standardized descriptions. Semantic annotations of services make services easier to find, match, map, and integrate. In 2007, W3C adopted SAWSDL as a recommendation for describing Semantic Web Services. This chapter presents what is termed "hybrid semantic matchmaker" for services described in SAWSDL. It incorporates three types of semantic matching, including logic-based, text-similarity-based and XML-tree edit-based structural similarity, and combines them using an adaptive matching approach by using a training set to help decide semantic relevance to rank component matching in the aggregated matching process.

SECTION 3: SEMANTIC APPLICATIONS

Section 3 begins with chapter 8, Enhancing Folksonomy-Based Content Retrieval with Semantic Web Technology, by Rachanee Ungrangsi, Chutiporn Anutariya, and Vilas Wuwongse. This work is at the intersection of Web 2.0 and Semantic Web for image content. Quality of so called folkonomy tagging in current systems such as Flickr is usually poor due to lack of common terminology, use of single word tagging, lack of synonym support, and lack of adequate context. This work discusses the SemFlickr system that supports semantic querying which involves a system that first retrieves relevant ontologies from among existing ontologies and finding term suggestions for use as tags from those ontologies. It takes the ontological relations among the given query terms to assign tag scores and then generates its ranked results. SemFlickr’s ability to boost image retrieval using semantics is demonstrated with respect to Flickr.

Chapter 9 is titled Semantic Search on Unstructured Data: Explicit Knowledge through Data Recycling, and it was written by Alex Kohn, François Bry, and Alexander Manta. Search is the most important Web application, and using semantics to improve search is something researchers have been investigating since the very early days of Semantic Web. Early examples include a semantic search engine called MediaAnywhere from Taalee, perhaps the earliest US Semantic Web company (details can be found in a patent awarded in 2001 titled “System and method for creating a Semantic Web and its applications in browsing, searching, profiling, personalization, and advertising), and a paper on Semantic Search in 2003 World Wide Web Conference. This chapter presents a semantic search system called YASA that uses ontology to improve information retrieval within an enterprise. The novel feature of YASA is that a large part of the ontology’s fact base is automatically built by recycling and transforming existing data. YASA is a fully implemented system that is evaluated in a pharmaceutical company with very positive results. The systems features include context-based personalization, faceted navigation, and of course, semantic search.

The tenth chapter, Ontology-Enhanced User Interfaces: A Survey, was composed by Heiko Paulheim and Florian Probst. Ontologies have been used as part of engineering numerous software systems. However, only a few of these systems are user interfaces. This chapter discusses a variety of the ways in which user interface developers can enhance their systems using ontologies. Examples include adapting user interfaces to a user’s needs and providing input assistance. It then outlines the corresponding requirements for the ontologies and their use in the context of enhancing user interface functions. This characterization serves two purposes: it allows for a better understanding of ontology-enhanced user interfaces, and it supports developers who want to use ontologies for a certain purposes in a user interface by pinpointing the relevant requirements. The survey ends with identifying new interesting research directions.
Chapter 11, Integrating Interactive TV Services and the Web through Semantics, was written by Vassileios Tsetsos, Antonis Papadimitriou, Christos Anagnostopoulos, and Stathes Hadjieffthymiades. The concept of interactive TV (iTV) has been around for a while, and Electronic Program Guides and program recommendation services can provide some useful capabilities. However, semantics and Semantic Web technologies have the promise to repurpose TV content more easily, support more advanced features, and even make access easier for end users. This chapter shows use of custom-made ontologies and rules, along with relevant standards in TV and Semantic Web areas, to support formal modeling of multimedia and user semantics, to create an iTV system that includes personalization and proactive content delivery to end users.

Chapter 12, Music Retrieval and Recommendation Scheme Based on Varying Mood Sequences, was written by Sanghoon Jun, Seungmin Rho, and Eenjun Hwang. This chapter deals with semantics in a broader sense, unlike other chapters; in this value, it does not use Semantic Web technologies. Music recommendation systems already utilize high-level musical features such as harmonics, beat, loudness, tonality, et cetera This chapter utilizes low-level features as base knowledge for music classification and recommendation, and develops an alternative strategy that involves measuring the similarity of music by using musical mood variation. Its recommendation system is based on applied artificial neural network algorithm to component ratio vectors of each music sequence and user preference rated playlist. It succeeds in achieving an average 70% classification accuracy.