RS2fx

# RDF by Structured Reference to Semantics

## An Approach to Emerge Semantic Web

By

**SADH | RATUL | SOURAV**

Department of Computer Science and Engineering
Khulna University of Engineering & Technology

# An Approach to Emerge Semantic Web

thesis report for the Course №: CSE 4000

by

| Khan Muhammad Nafee Mostafa | Samiul Hoque Sourav | Qudrat-E-Alahy Ratul |
|---|---|---|
| 0507007 | 0507035 | 0507037 |

Supervised by:
**Rushdi Shams**
Lecturer,
Department of Computer Science and Engineering
Khulna University of Engineering & Technology

Signature

A thesis report submitted in partial fulfillment of the requirements of the Khulna University of Engineering & Technology for the degree of Bachelor of Science in Computer Science and Engineering

April 2010

# Abstract

Current standard web documents are designed to be presented to humans. Machines have no idea about the information located in a web document. Semantic web is organized in a structured way so that it is meaningful to both machines and humans. In this thesis work, we suggest a framework that will process the web documents and produce machine readable format in RDF (Resource Description Framework) collaborated with the OWL (Web Ontology Language).

Our suggested framework, which we call RS2 (RDF by Structured Reference to Semantics), takes an HTML document as input, extracts the plain text from it. Natural language context of plaintext is then parsed to yield subject-object-predicate of each sentence. This data is used to lookup in the ontology and generate RDF graph which is the machine intelligible semantic equivalent to the original human recognized text.

# Acknowledgements

We express our gratitude to the Almighty and our family. We also thank Batch 2K5 of CSE and whole CSE department of KUET for being a beaming source of inspiration. We are grateful to Fazlul Hoque Hall tenants, who have been constant source of our lively life in KUET. Without the support of all these people this thesis could not have been carried out.

W3C and several other group of people have worked and been working in the field of Semantic Web. Our respects go to all of them. We would like to especially mention the names of, Stanford University for protégé, OpenLink Software for the Virtuoso Open-Source and off course Sir Tim Berners-Lee, the inventor of World Wide Web for coining the term ‘Semantic Web’.

We have used the natural language parser designed by Nazmul Hasnat Arka and Sadre-Ala Parvez of KUET CSE 2k5. We are very thankful to them.

Finally, we would also like to thank our supervisor Rushdi Shams for his academic guidance and support throughout this period.

# Contents

# List of Figures

# List of Tables

# An Approach to Emerge Semantic Web

# Chapter 1

# Introduction

## 1.1 Background

World Wide Web[*] is a rapidly evolving technology and arguably the greatest technological success in the history. From the start, dated back in the 1990, when web was just a collection of html[†] documents it gathered a lot of attention from wide variety of people and continually introduced new and newer features. Gradually most of its contents turned into web application rather than being simple web pages and now is regarded as a great information source, communication media and the social network. Even so, it has been kept dumb since its birth and thus turned into a clutter of valuable resources where users often fail to find desired content and give up by blaming the whole internet system.

Web today is in its second version, often called *Web 2*.0 or the *Social Web*, which connects people. In future web will have senses in order to connect knowledge. This version, comprising of *Semantic Web,* will be called *Web 3.0*. [1]

The term *Semantic Web* was first coined by Tim Berners-Lee, the inventor of web and according to him and others, *"The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation".* [2]

Although, web is evolving from one version to another and increase own version number quite quietly, current web documents will certainly not fit into *Semantic Web* without modifications. To provide sense in the web, web documents should also be sensible to the machines as well as to human beings. RDF[‡] is a well-established technology for providing machine

[*] Later in this text simply referred to as *Web*

[†] Hypertext Markup Language

[‡] Resource Description Framework

recognizable meaning to the web documents. These RDF documents constitute some URI[*]s and the resources are described in OWL[†] documents. RDF incorporates with RDFS[‡] that describes the RDF vocabulary. RDF, RDFS, OWL and other documents that will constitute Semantic Web are necessarily XML documents.

Web is now compiled of html documents in a scattered way. Introduction of new technologies will obviously cause a shift in the internal structure of Web. Evidently, a great challenge of emerging Semantic Web is to convert the existing human-readable documents into *machine intelligible* ones. Hence, a major target of our quest is to develop tools for opting in html web documents into Semantic Web data. To do so, we are suggesting a framework titled as ***RDF by Structured Reference to Semantics***[§] **(RS2)**, which will read web document, NL parse it and map them into Semantic Web entities based on existing ontology to prepare an RDF graph.

In 1989, Tim Berners-Lee, based on earlier features of hypertext wrote a proposal [3] that has eventually turned into web in 1990. In the early stage web was a system of interlinked hypertext documents accessed via the Internet. In those days, the web consisted of web sites and static web documents only. People may browse to an address, open some web document or may download some images. From then every day, the web grew larger and once people started to think differently. Many web developers thought that the web might have some user interactions. By this time several technologies like JavaScript[**], PHP[††], ASP[‡‡], JSP[§§] etc. came in action and the web document format HTML was also improved with the introduction of CSS[***] [4]. Thus, web developers started to develop web services that can act interactively with the users and web documents turned into web applications rather than static pages.

[*] Universal Resource Identifiers

[†] Web Ontology Language

[‡] RDF Schema

[§] RDF by Structured Reference to Semantics *abbreviated as* RSRS *or* $(RS)^2$ = RS2

[**] JavaScript's official name is ECMAScript and ECMA-262 is the official standard.

[††] PHP Hypertext Preprocessor, *an open source web service language*

[‡‡] Active Server Pages *from Microsoft*

[§§] Java Server Pages, *from Sun Microsystems, with the advent of Java*

[***] Cascading Style Sheets

The evolution of web is fueled by standardization of technologies and especially by the formation of W3C[*] in 1994. HTML documents were not standardized before HTML 2.0 [5] came in 1995. After several modifications, current stable standard reached is HTML 4.01 [6] . HTML is even undergoing enhancement and latest HTML 5 is in 'Working Draft' as of 04 March 2010 [7].

HTML is a scripting language like XML[†]. To conform it to XML, W3C made a reformulation of HTML 4 in XML 1.0 in 26 January 2000 and revised in 1 August 2002 [8]. This was named XHTML[‡].

With the advent of JavaScript web pages turned user interactive. Then, AJAX[§] was introduced to facilitate asynchronous requests to web servers and enabled clients to frequently link with the servers. Meanwhile PHP, JSP, ASP and other server-side programming languages geared up web applications. Soon, the aspects of web were changed by the introduction of social web services.

In the early 2004, web community started to experience new dimension of services, widely known as social web. Some of these social web applications are Flickr [9], Wikipedia [10], digg [11], Facebook [12] and many others.

The term *Web 2.0* was first coined by Tim O'Reilly in 2005 [13]. This term was used to give significance to the then ongoing trends of web applications. Web 2.0 is the dynamic social web where users can create and share contents, connect with people and collaboratively provide and attain services.

While Web 2.0 targeted people, Web 3.0 emphasizes on machine knowledge and sense. Although, specific nature of Web 3.0 yet to be precisely defined the outline has become clear over past few years. Semantic Web has the semantics in a document underlying any web document. This semantic can refer to some objects in ontology and ignite some 'sense' to the machine. This sense can be realized by the web application developers to provide intelligent services. Rest of this text will emphasize on Semantic Web and approaches to emerge it.

[*] World Wide Web Consortium

[†] Extensible Markup Language

[‡] Extensible Hypertext Markup Language

[§] Asynchronous JavaScript and XML

## 1.2 Problem Statement

Human beings communicate with each other on a very high level language commonly known as *Natural Language (NL)*[*]. Human brain organizes a *knowledge graph* maintaining some hierarchy. It also has a *lexicon* which can map any known NL word to corresponding node in its own knowledge graph. This knowledge graph contains the *concept* and *description* of any intelligible word or phrase. Upon encountering any NL sentence and successfully recognizing each words and mapping them into knowledge graph human beings conceives the *semantics*[†] of the sentence.

Humankind do not store all its knowledge in brain, rather each human being has its own *concept graph* or *ontology*[‡] while most of their descriptive knowledge are stored and transmitted in natural languages. NL itself can be presented as speech or text. Texts are books, handwritings, notes, printed documents, web documents and so forth. Certainly, web is contemporarily the largest repository and exchange medium of human knowledge. Still, even with the advent of astonishing web technologies, web itself cannot sense what it is storing or transmitting. Of course, the obvious problem is the incompatibility between the knowledge representation structure of humankind and machines.

A long cherished dream of humankind is to avail an agent that can conceive, gather and enrich knowledge and act upon its knowledge. Web can be the promise to fulfill the dream. In regard to this objective it is necessary to define some methodology in which web will be able to process knowledge i.e. there

[*] **NL,** *Natural Language;* from here on will be abbreviated as *NL* in this text

[†] *Semantics* is the study of meaning, one of the major areas of linguistic study [74]
*Linguists have approached it in a variety of ways. Members of the school of interpretive semantics study the structures of language independent of their conditions of use. In contrast, the advocates of generative semantics insist that the meaning of sentences is a function of their use. Still another group maintains that semantics will not advance until theorists take into account the psychological questions of how people form concepts and how these relate to word meanings.* [73]

[‡] Ontology is the study of beings in existence [74]
*In computer science and information science, ontology is a formal representation of the knowledge by a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain, and may be used to describe the domain.*
*According to a protégé publication* [75], *an ontology is a formal explicit description of concepts in a domain of discourse (classes (sometimes called concepts)), properties of each concept describing various features and attributes of the concept (slots (sometimes called roles or properties)), and restrictions on slots (facets (sometimes called role restrictions)). Ontology together with a set of individual instances of classes constitutes a knowledge base.*

shall be some way for machines to comprehend semantics. As part of Semantic Web activity [14], W3C has defined some technology and the Semantic Web Stack *(see 2.3.1)*. Implementation of the stack and these technologies will lead web to emerge in to next generation of it, the so called Semantic Web.

Knowledge graph is actually a clutter of a *concept graph* and several description graphs. W3C has suggested that, concept graph or ontology will be defined in an OWL (Web Ontology Language) file. Description graphs are represented with RDF (Resource Description Framework). Both of them are presented as XML tree.

For any complete Semantic Web system there should be ontology stored in OWL. We primarily would work with manually developed OWL graphs. Eventually, any web document has an underlying description graph, constituting the machine intelligible semantic expressible as an RDF graph. Our task is to elicit the RDF graph of an existing web document.

A web document is a collection of NL sentences. A sentence in fact is an SPO[*] expressible as a function,

$$predicate(subject, object)$$

Likewise, a node in RDF graph is a triple of SPO,

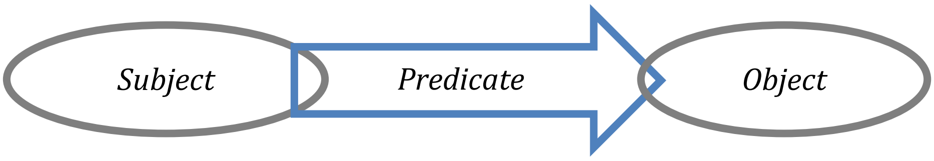


*Figure 1-1: an RDF triple as SPO*

We are going to suggest a framework that will take input of existing web document and give output of underlying RDF (i.e. an NL to RDF translation framework). Output RDF will be derived in terms of an existing OWL ontology.

For developing such a framework we need an *NL Parser* that will generate SPO of sentences and a *mapper* that will map *NL entities* to *Semantic Web entities*.

A prototype application is needed to demonstrate the suggested framework.

[*] *SPO* = Subject Predicate Object

## 1.3 Motivation

Today the web is a great collection of information and data. People from around world everyday elicit knew knowledge and express them in various forms through the web. One can find thousands of text documents in any subject of interest; let it be about global economics or just about paperclip collection hobby; any individual will encounter a bulk of web pages staring around and trying to get read. But does he or she know which specific document is to read? Actually s/he needs someone who will suggest a small list of documents.

For another instance, let there be web documents that says peperoni is a kind of pizza topping. There are notion of many other pizza tipping also. But, there is no web document that explicitly lists all pizza toppings. You need to list all type of pizza toppings; again you need someone who will read all these piles of web documents and elicit a list of pizza toppings.

Now, a question already has arose, “who will answer questions for you from gathering data from web?” The answer is, “World Wide Web itself”!

There are two instances above that urge that we need an intelligent being that will conceive and comprehend knowledge for humankind from web. This has motivated many web experts to come with the idea of intelligent web to relieve web users from the frustration of being clueless in piles of files. Web will be intelligent if web documents become intelligible or some semantics are added to web documents. To introduce semantics into web, five approaches are taken:

1. Document Tagging and metadata
2. Statistical Approach
3. Linguistic processing of documents
4. Semantic Web
5. Intelligent web

W3C’s Semantic Web activity suggests accompanying NL web documents with machine intelligible semantic graphs (RDF, OWL etc.). The main bottleneck in emergence of Semantic Web is that, *most of the published content isn’t structured to allow for logical reasoning* [15]. As the *Semantic Web is about giving information a well-defined meaning* [2]*, better enabling computers and people to work in cooperation,* these documents need to be converted into machine-processable forms. As of March 2010, there are more than 200 million hostnames with about 100 million active websites. All of these sites constitute billions of webpages. It is quite impossible to hand code all of html documents into machine intelligible RDF files.

Web, today, is like a *horde* of valuable documents with humankind's precious knowledge left unorganized in a very scattered fashion. In the process of generating RDF graphs from web documents using single centralized[*] ontology these billion of documents will become machine intelligible document and also will be linked in an organized structure, like a *squad* of knowledge.

In order to generate RDF graph from HTML file we will focus on,

- Extract Natural Language text from an XHTML consistent web document
- Parse each NL sentence with grammar into a parse tree
- Yield Subject Predicate and Object (SPO) from the parsed tree of NL sentence
- Look up Semantic Web entities into a mapper for each NL entity in a SPO
- Generate RDF triple from retrieved Semantic Web SPO

## 1.4 Thesis organization

The thesis is organized as follow,

In this chapter, the basic idea of Semantic Web is introduced and the motivation behind the work is mentioned.

In the next chapter, background studies, current state of Semantic Web, versions of web, related technologies, existing works, related works and research methodology is discussed.

Chapter three describes the suggested RS2 framework in detail.

[*] Centralized in the sense, that all services will be built on knowledge processed from same set of ontology; this ontology can even be stored in a distributed way with replication or whatever way seem just

# Chapter 2

# Semantic Web Today

## 2.1 Introduction

The emerging Semantic Web is viewed as the next version of existing World Wide Web, Web 3.0. It is preceded by current Web 2.0 and expected to be succeeded by Web 4.0. In this chapter a comparative study of web versions are placed accompanying views of major web experts.

Semantic web promises to infuse the internet with a combination of metadata, structure, and various technologies so that machines can derive meaning from information, make intelligent choices, and complete tasks with reduced human intervention. To do so, some technology specifications have been standardized and some are yet to be standardized. In section *2.3* we have briefly discussed the Semantic Web stack, related technologies including RDF, OWL, SPARQL and other specifications.

The idea of Semantic Web has been conceived over a decade of effort from various people. They have worked out general development environments, RDF triple storing systems, programming environments and other tools as contribution to the emergence of Semantic Web. There are several web applications which are undergoing research and development in order to provide Semantic Web services. These include twine, freebase, Swoogle, DBpedia, Powerset and several more. Several researches are also carried out to convert HTML documents into RDF; this chapter will also describe one in brief in section *2.4.3*.

## 2.2 Web versions

In section 1.1, we have already asserted how web has evolved from simple documents to highly extensible application network. Since Tim O'Reilly coined the term Web 2.0 in 2005 [13], in response to the evolutionary shift of web, the notion of web version has greatly being discussed by some group. Web 2.0 is said to be fully emerged in 2004 and Web 3.0 is in early stage of emerging. Experts also expect Web 4.0 to come after emergence of Web 3.0.

*Four Web versions are described in brief as:*

***Web 1.0*** – That Geocities & Hotmail era was all about read-only content and static HTML websites. People preferred navigating the web through link directories of Yahoo! and dmoz.

***Web 2.0*** – This is about user-generated content and the read-write web. People are consuming as well as contributing information through blogs or sites like Flickr, YouTube, Digg, etc. The line dividing a consumer and content publisher is increasingly getting blurred in the Web 2.0 era.

***Web 3.0*** – This will be about Semantic Web (or the meaning of data), personalization, intelligent search and behavioral advertising among other things.

***Web 4.0*** – This will be ubiquitous that will connect intelligence, semantics and social web activities.

*Figure 2-1* below shows evolution of web 1 and 2 along with the expected evolution of web 3 and 4. Major features of each version are written in the area denoting the respective version in the graph.

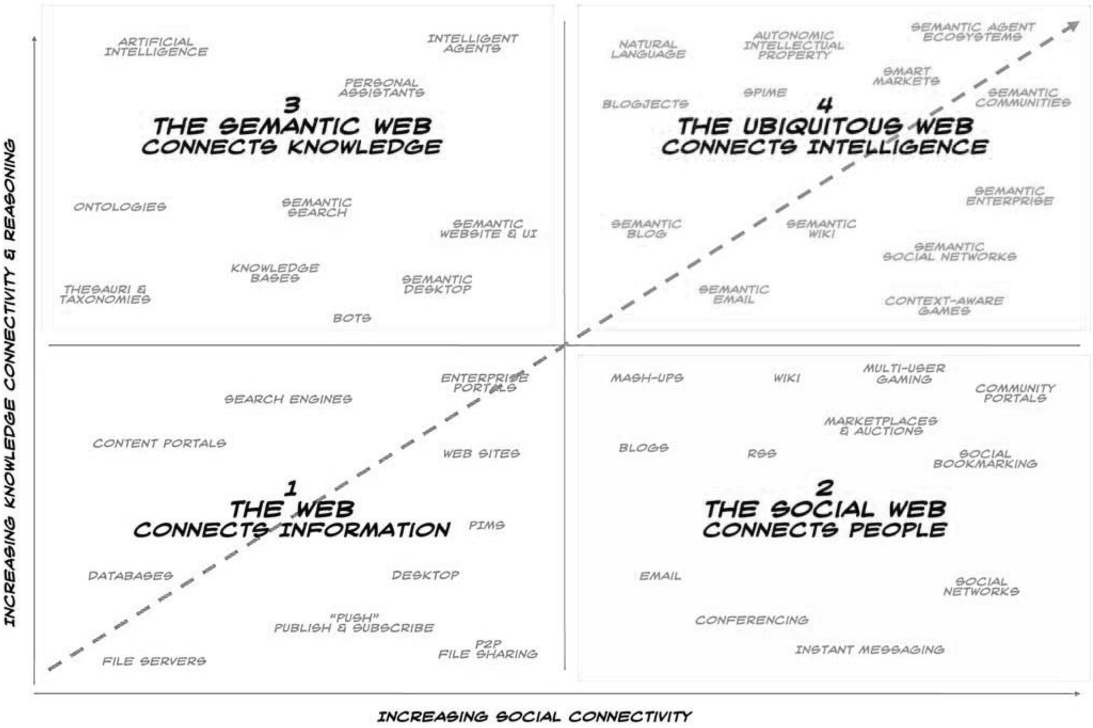


*Figure 2-1: Evolution of internet. [Source: Nova Spivak, Radar Networks; John Breslin, Deri; & Mills Davis, Projectiox]*

A comparison of first three versions of web is given below,

| Web 1.0 | Web 2.0 | Web 3.0 |
|---|---|---|
| “The mostly read only web” | “the widely read-write web” | “the portable personal web” |
| 45 million global users(1996) | 1 billion+ global users (2006) | ? |
| Focused on companies | Focused on communities | Focused on individuals |
| Home pages | Blogs | Consolidating dynamic content |
| Owning content | Sharing content | Semantic Web content |
| Online static document | Community effort | Widget, mashup |
| HTML, portals | XML, RSS | User behavior |
| Web form | Web application | Personalized web |
| Directories (taxonomy) | Tagging (“folksonomy”) | User engagement |
| Page view | Cost per click | ? |
| Advertising | Word of mouth | “Advertainment” |

*Table 2-1: Comparison of Web versions. This table neatly sums up the main differences between Web 1.0, Web 2.0 and Web 3.0.*

Let us now concentrate on how major experts in this field are thinking about Semantic Web,

Tim Berners-Lee, the expert who is responsible for the core principles that originated the World Wide Web, coined the term “Semantic Web”, and he seems to advocate the idea of converting the Web into a huge database, where we can make very sophisticated and complex queries.

> *"I think maybe when you've got an overlay of scalable vector graphics - everything rippling and folding and looking misty – on Web 2.0 and access to a semantic Web integrated across a huge space of data, you'll have access to an unbelievable data resource." —Tim Berners-Lee [16]*

Yahoo founder, Jerry Yang thinks that with the new generation of tools for the creation of content and online applications there will be a blurring in the distinction of professionals, semi professionals and consumers. At the TechNet Summit in November 2006, he stated:

> *“…you don't have to be a computer scientist to create a program. We are seeing that manifest in Web 2.0 and 3.0 will be a great extension of that, a true communal medium…the distinction between professional, semi-professional and consumers will get blurred, creating a network effect of business and applications.” — Jerry Yang [17]*

Nova Spivak from Radar Networks believes that the semantic Web will play a central role in the new generation, although he recognizes that there will be other important technologies that should have a great impact as well [18] Spivak also suggests that the versioning should be used to refer to decades of development on the Web, instead to specific sets of features that define them [19]. Finally when Google's CEO, Eric Schmidt, was asked to define Web 2.0 and Web 3.0 he said:

> *"…If I were to guess what Web 3.0 is, I would tell you that it's a different way of building applications... My prediction would be that Web 3.0 will ultimately be seen as applications which are pieced together. There are a number of characteristics: the applications are relatively small, the data is in the cloud, the applications can run on any device, PC or mobile phone, the applications are very fast and they're very customizable. Furthermore, the applications are distributed virally: literally by social networks, by email. You won't go to the store and purchase them...." [20]*

Netflix founder, Reed Hastings goes for a definition related to the amount of bandwidth available; he thinks the increase of bandwidth will allow for the full video Web.

> *"Web 1.0 was dial-up, 50K average bandwidth, Web 2.0 is an average 1 megabit of bandwidth and Web 3.0 will be 10 megabits of bandwidth all the time, which will be the full video Web, and that will feel like Web 3.0" — Reed Hastings [17]*

These are just a few examples of the different perceptions of experts in the information technology industry. It is obvious that the goal of the World Wide Web is to enable universal information access, which considers the delivery of content under different usage environments.

## 2.3 Overview of Related Technologies

Semantic Web is completely a new set of technologies. For working with Semantic Web or Web 3.0 a clear overview of these technologies are necessary. This section briefly introduces the reader with related technologies including the semantic web stack, RDF, OWL, SPARQL, XML, XML namespace URI etc. References included in this section will be helpful for further reading on these technologies.

### 2.3.1 Semantic Web stack

The semantic layer cake is rather famous for describing the illustration of the key Semantic Web enabling technologies. Building one upon another from bottom to top, these technologies can help us realize the full Semantic Web vision. It is known as *Semantic Web Stack* [21].

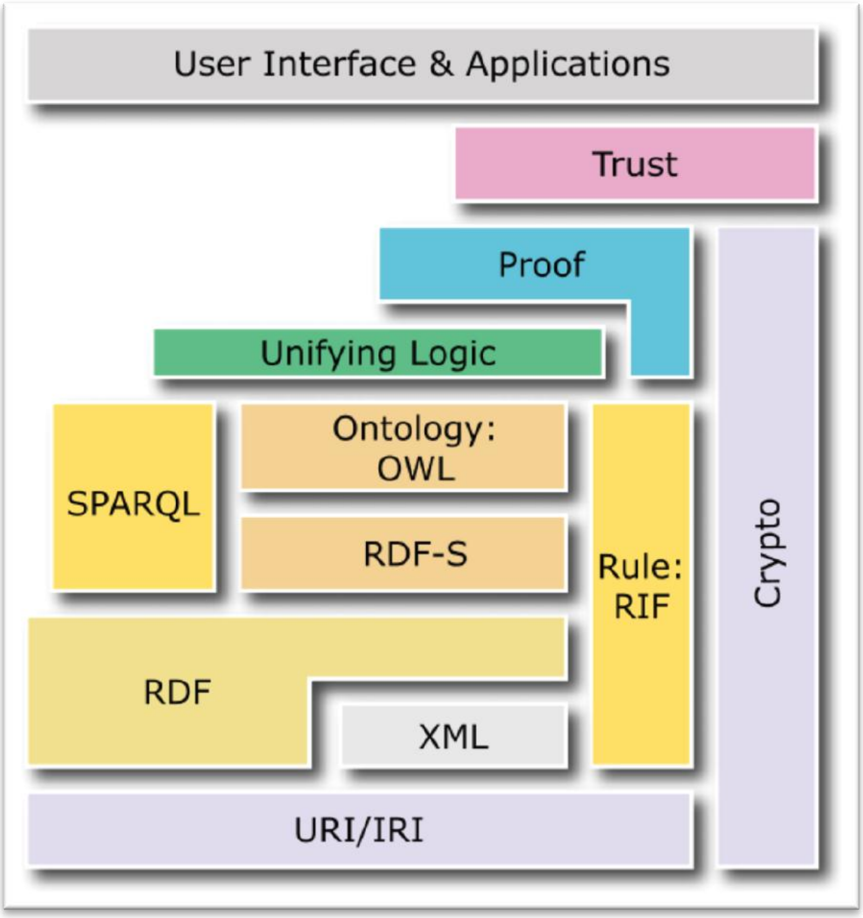


*Figure 2-2: Semantic Web Stack*

The Semantic Web Stack is an illustration of the hierarchy of languages, where each layer exploits and uses capabilities of the layers below. It shows how technologies that are standardized for Semantic Web are organized to make the Semantic Web possible. It also shows how Semantic Web is an extension (not replacement) of classical hypertext web. The illustration was created by Tim Berners-Lee. The stack is still evolving as the layers are concretized.

From the developers' view, the Stack can be fragmented in three parts. From the bottom up to the XML are well known *Hypertext Web Technology*. From the XML up to the OWL the technologies are known as *Standardized Semantic Web Technology*. It is not clear how the top of the stack will implemented, known as *Unrealized Semantic Web Technologies*. This section concentrates on brief discussion of these technologies.

### 2.3.2 Hypertext Web Technologies

*These are the Stack's bottom layer technologies. These technologies are of course standardized for a long time. These are well-known hypertext technology. These existing technologies can be implemented in semantic web technology without changing the basic principle.*

**URI** is acronym for Uniform Resource Identifier [22]; a compact string of characters used to identify or name a resource. The URL* to a web site (e.g. http://www.semanticfocus.com) is a popular example of an URI [23].

**IRI** is acronym for Internationalized Resource Identifiers [24]. IRIs are a generalization of Uniform Resource Identifiers (URIs) that admits the use of Unicode for expressing symbols that are not supported in URIs yet. This is becoming important part in internet technology.

**Unicode** is the universal standard encoding system and provides a unified system for representing textual data [25]. One million characters can be encoded to specify any character in any language without a single escape sequence or control code. Before Unicode, there were several different encoding systems, which made communication and integration across borders very difficult. Now it is so much easier.

**XML** is acronym for Extensible Markup Language [26] standardized by W3C [27]. It enables the creation of document composed of structured data. With XML, we have a standard way to compose information so that it can be more easily shared. It is a set of rules for encoding the documents electronically. XML's design goals emphasize simplicity, generality, and usability over the Internet. This textual format allows Unicode. [28]

**XML namespaces** are used for providing uniquely named elements and attributes in an XML document. They are defined in Namespaces in XML, a W3C recommendation. An XML instance may contain element or attribute names from more than one XML vocabulary. If each vocabulary is given a namespace then the ambiguity between identically named elements or attributes can be resolved. [28]

**XML Schema** describes the structure of XML documents just like DTDs, only better. An XML Schema is known as an XML Schema Definition (XSD). XSD provides the way to define rules (like guidelines) so that people and machines can understand them, adhere to them, and integrate with them.

* Uniform Resource Locator

**XML Query** (also known as XQuery) is a standardized language for combining documents, databases, Web pages and almost anything else. It is very widely implemented, powerful, and easy to learn. XQuery is replacing proprietary middleware languages and Web Application development languages. XQuery is replacing complex Java or C++ programs with a few lines of code. [28]

### 2.3.3 Standardized Semantic Web technologies

*These are the middle layer technologies. These technologies are standardized by W3C to enable building semantic web application.*

**RDF** is **R**esource **D**escription **F**ramework — a W3C recommendation [29]. RDF is a graphical formalism for representing metadata and for describing the semantics of information in a machine- accessible way. It also provides simple data model for based on triple. The RDF can be represented by these triple *<subject, predicate, object>.*

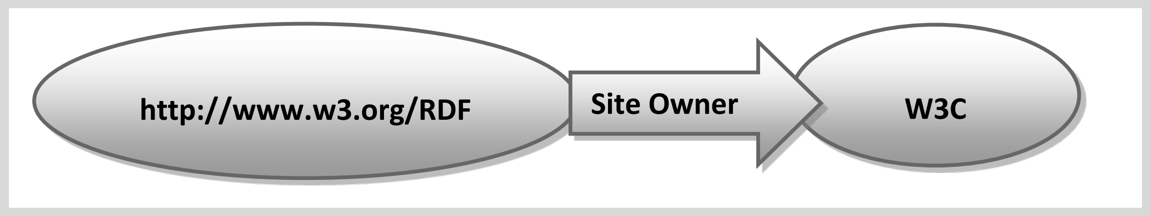


*Figure 2-3: an RDF graph*

This relationship can be represented as,

```
<http://www.w3.org/RDF, Site owner, W3C>
```

In the concept of RDF, properties themselves can be a URI; the subject of one statement can be the object of another statement. Thus, RDF gives us a mechanism for annotating data and resource.

**RDFS** is an acronym for RDF Schema [30]. RDF Schema extends RDF with a schema vocabulary that allows us to define basic vocabulary terms and the relations between those terms. RDF Schema does not provide actual application-specific classes and properties. Instead, RDF Schema provides the framework to describe application-specific classes and properties. Classes in RDF Schema are much like classes in object oriented programming languages. This allows resources to be defined as instances of classes, and subclasses of classes.

**OWL** stands for **W**eb **O**ntology **L**anguage [31]. The OWL Web Ontology Language is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics. OWL has three increasingly expressive sublanguages: OWL Lite, OWL DL, and OWL Full. Full.

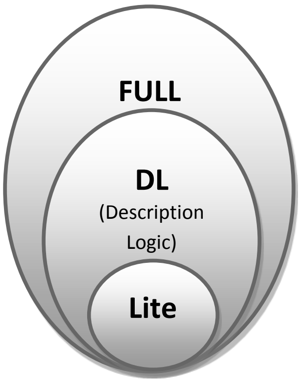


*Figure 2-4: three dialects of OWL*

*OWL Lite* supports those users primarily needing a classification hierarchy and simple constraints.

*OWL DL* supports those users who want the maximum expressiveness while retaining computational completeness and decidability. OWL DL includes all OWL language constructs, but they can be used only under certain restrictions. OWL DL is so named due to its correspondence with *description logic*, a field of research that has studied the logics that form the formal foundation of OWL.

*OWL Full* is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. OWL Full allows an ontology to augment the meaning of the pre-defined (RDF or OWL) vocabulary. It is unlikely that any reasoning software will be able to support complete reasoning for every feature of OWL Full.

**SKOS** (Simple Knowledge Organization System) is, strictly speaking, a vocabulary built on RDFS and OWL, with specifications [32] recently developed. However, it shares similar application areas as those other languages. SKOS lets us express classification systems such as taxonomies and thesauri in the RDF model when RDFS and OWL's logical strictures (used directly) might be too strong. It offers a straightforward migration path from existing knowledge organization systems to Semantic Web technologies.

**SPARQL** is acronym for **S**PARQL **P**rotocol **A**nd **R**DF **Q**uery **L**anguage [33] [34]. It is a query language for RDF [35]. SPARQL Protocol is described in two ways: first, as an abstract interface independent of any concrete realization, implementation, or binding to another protocol; second, as HTTP and SOAP bindings of this interface. SPARQL allows a query to consist of *triple patterns, conjunctions, disjunctions,* and *optional patterns.* RDF, language of Semantic Web, is a directed, labeled graph data format for representing information in the Web. SPARQL can be used to express queries across diverse data sources, whether the data is stored natively as RDF or viewed as RDF via middleware. SPARQL contains capabilities for querying required and optional graph patterns along with their conjunctions and disjunctions. SPARQL also supports extensible value testing and constraining queries by source RDF graph. The results of SPARQL queries can be results sets or RDF graphs.

### 2.3.4 Unrealized Semantic Web technologies

*Top layers contain some technologies that are yet not been standardized or contain just ideas that should be implemented in order to realize Semantic Web.*

**RIF** stands for **R**ule **I**nterchange **F**ormat [36]. It is a proposed component for Semantic Web and being developed by RIF Working Group [37] . Yet the rule is not standardized. The World Wide Web Consortium is developing it as a potentially recommended format for the interchange of rules in rule-based systems on the semantic web. The goal is to create an interchange format for different rule languages and inference engines. (*See also* [38])

**SWRL** (**S**emantic **W**eb **R**ule **L**anguage) is a proposal for a Semantic Web rules-language, combining sublanguages of the OWL Web Ontology Language (OWL DL and Lite) with those of the Rule Markup Language [39].

**Cryptography** is important to ensure and verify that semantic web statements are coming from trusted source. This can be achieved by appropriate **digital signature** of RDF statements.

**User interface** is the final layer that will enable humans to use semantic web applications.

## 2.4 Related Work

Several groups of people have been working on Semantic Web for more than a decade. As the output of their effort, many nice tools have been developed that can be used to work with Semantic Web features. Some web sites are also trying to come out with Semantic Web features. Short introduction to some of these are given in sections 2.4.1 and 2.4.2 respectively. Section 2.4.3 describes a contemporary research for generating RDF from HTML. Details of these contexts are out of the scope of this thesis report. So, short introduction is presented here along with some references that contain detail information.

### 2.4.1 Existing tools for working with Semantic Web

**General Development Environments**

***Protégé*** [40] is an extensible, platform-independent environment for creating and editing ontology. It has OWL plugin (called Protégé-OWL) to edit RDF and OWL ontology as well as SWRL rules, a visual editor for OWL (called OWLViz), storage back-ends to Jena and Sesame, as well as an OWL-S plugin, which provides some specialized capabilities for editing OWL-S descriptions of Web services.

**The *Semantic Web Client Library*** [41] represents the complete Semantic Web as a single RDF graph. The library enables applications to query this global graph using SPARQL. To answer queries, the library dynamically retrieves information from the Semantic Web by dereferencing HTTP URIs and by following RDFS. The library is written in Java and is based on the Jena framework.

**RDF Triple Store Systems**

***Sesame*** [42] is an open source RDF database with support for RDF Schema inferencing and querying. It offers a large scale of tools to developers to leverage the power of RDF and RDF Schema. It is a Java framework; there is also a python wrapper.

***Oracle Spatial 11g*** [43] includes an open, scalable, secure and reliable RDF management platform. Based on a graph data model, RDF triples are persisted, indexed and queried, similar to other object-relational data types.

**Programming Environments**

***Jena*** [44] Java RDF API and toolkit is a Java framework to construct Semantic Web Applications. It provides a programmatic environment for RDF, RDFS and OWL, SPARQL, and includes a rule-based inference engine. It also has the ability to be used as an RDF database via its Joseki layer. Jena's SDB layer offers an RDF Triple Store facility with SPARQL interface on top of other database systems

***Rowlex*** [45] .NET library and toolkit built to create and browse RDF documents easily. It abstracts away the level of RDF triples and elevates the level of the programming work to (OWL) classes and properties.

**RDF Generator**

***Zemanta API*** [46] is a web API that delivers relevant tags, links, categories and pictures from unstructured data/content. It is semantic standards compliant, with RDF output and ability to disambiguate to entities from Linking Open Data.

### 2.4.2 Semantic Web Implementation Approaches

So far there are several partial approaches to implementation of Semantic web are available. Several of them are discussed below.

**Twine** is an application that helps people organize, share and discover information around their interests. All information in Twine is expressed in a set of triples (RDF). When two triples refer to the same object, they become linked; in this way start to build a semantic graph. In short, Twine uses triples to access a tremendous breadth and depth of information about any given subject. It uses another important technology called OWL (Web Ontology Language) to define properties and classes, and to determine their uses. The system employs natural language processing and machine learning to extract concepts from written text in user data and express it using RDF triples tied to a semantic taxonomy based on concepts.

**TrueKnowledge** combines natural language analysis, an internal knowledge base and external databases to offer immediate answers to various questions. Instead of just pointing to web pages where the search engine believes it can find answer, it will offer an explicit answer and explain the reasoning patch by which that answer was arrived at.

**Powerset** had done a good job of creating a rich semantic layer on top of Wikipedia. It is a searching a fixed subset of Wikipedia using conversational phrases rather than keywords. It brings a new, rich semantic dimension via natural language query processing to Wikipedia that greatly improves the search and reading experience. Its natural language search technology is based on its own proprietary indexing.

**Tripit** is an app that manages travel planning. It extracts useful information from these mails and makes a well-structured and organized presentation of your travel plan. It pulls out information from Wikipedia for the places that someone visits.

**Adaptive Blue** allows web site publishers to add semantically charged links to their site. Smart Links are browser 'in-page overlays' (similar to popup) that add additional contextual information to certain types of links. AdaptiveBlue supports a large list of top web sites, automatically recognizing and augmenting links to those properties.

**Freebase** is an open shared database of the world's knowledge and a massive, collaboratively edited database of cross-linked data. It provides an interface that allows non-programmers to fill in structured, or 'meta-data', of general information, and to categorize or connect data items in meaningful or 'semantic' ways. It supports queries from web developers wanting to build applications around them. It also solicits people to contribute their knowledge to the database, governed by a community of editors. It offers a Creative Commons license so that it can be used to power applications, on an open API.

**Swoogle** is a search engine for Semantic Web documents, terms and data found on the Web. It employs a system of crawlers to discover RDF documents and HTML documents with embedded RDF content. Swoogle reasons about these documents and their constituent parts, records, and indexes meaningful metadata about them in its database. It provides services to human users through a browser interface and to software agents via web services.

### 2.4.3 A New Semantic Web Services to Translate HTML Pages to RDF

Although, researches have been carried out about Semantic Web, only one published work has been found that described about HTML to RDF conversion.

Debajyoti Mukhopadhyay, Rituparna Kumar, Sourav R. Majumdar and Subhobroto Sinha of Web Intelligence & Distributed Computing Research Lab, Techno India Group, West Bengal University of Technology has proposed a method of translating HTML pages to RDF presented in a publication titled “A New Semantic Web Services to Translate HTML Pages to RDF” [47]. This section discuss about this research.

In this approach, HTML pages are crawled and then the html tags are stripped off. Then, sentences are delaminated, tokenized and POS Tagged. Then it was passed through an NL Processor and Synsets. Then the dependency is serialized. The process algorithm [47] is, given in figure below,

```
INPUT: A Web page (P), written in natural text or HTML.
OUTPUT: The serialized or RDF format of the processed
data of (P).

Step1 Parse the natural text corpora
Step2 IF the parsing is successful, THEN GOTO Step5
Step3 Log unknown class or classes which throw parsing
exceptions to log
Step4 Process automatically or notify expert to handle
the unknown classes or exceptions, GOTO Step8
Step5 Normalize the parse tree.
Step6 Process and tag or identify and correlate
entities recognized.
Step7 Serialize the processed data to RDF.
Step8 End
```

*Figure 2-5: Process Algorithm of "A New Semantic Web Services to Translate HTML Pages to RDF"*

The architecture [47] of the proposed method is given by the figure,

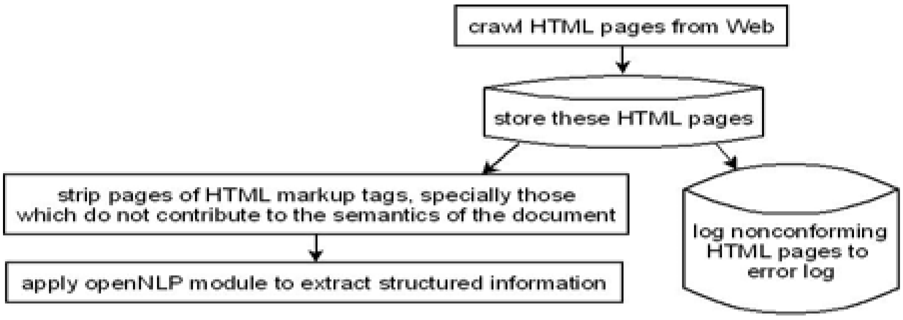

crawl HTML pages from Web
store these HTML pages
strip pages of HTML markup tags, specially those which do not contribute to the semantics of the document
log nonconforming HTML pages to error log
apply openNLP module to extract structured information

*Figure 2-6: Architecture of "A New Semantic Web Services to Translate HTML Pages to RDF"*

The approach described here is a nice one and can elicit some semantics of natural language text. It produces an RDF. But these RDF graphs do not refer to with any URI and it is not related to or built with respect to any ontology. Semantic Web definitely depends on ontology. RDF without URI and ontology can be processed in some extent, but real semantic ability is not achieved here. W3C recommends that, RDF entities relate to globally addressable ontology, preferably with an URI reference.

## 2.5 Research Methodology

Web is in a process of emerging Semantic Web. Semantic Web need ontology and RDF graphs to hold machine intelligible semantics. On the other hand, current web documents are not stored in this approach. Today there are more than 100 million active web domains and each of them has many web pages. For the emergence of Semantic Web, if we have to rewrite all the documents manually then, it will be almost impossible to hope that, Semantic Web will ever emerge. Semantic Web stack is already been defined (see 2.3.1) by W3C. Specification for the Semantic Web data, alternative to contemporary web document, is achieved; W3C has also defined methods for storing and working with ontologies. Yet, we do not see a lot of web services to come with Semantic Web capabilities, because there is no well-established automated way to convert html to RDF. Hence, we are going to propose an approach to convert html to RDF graphs.

Figure 2-7 below, describes the research methodology in short.

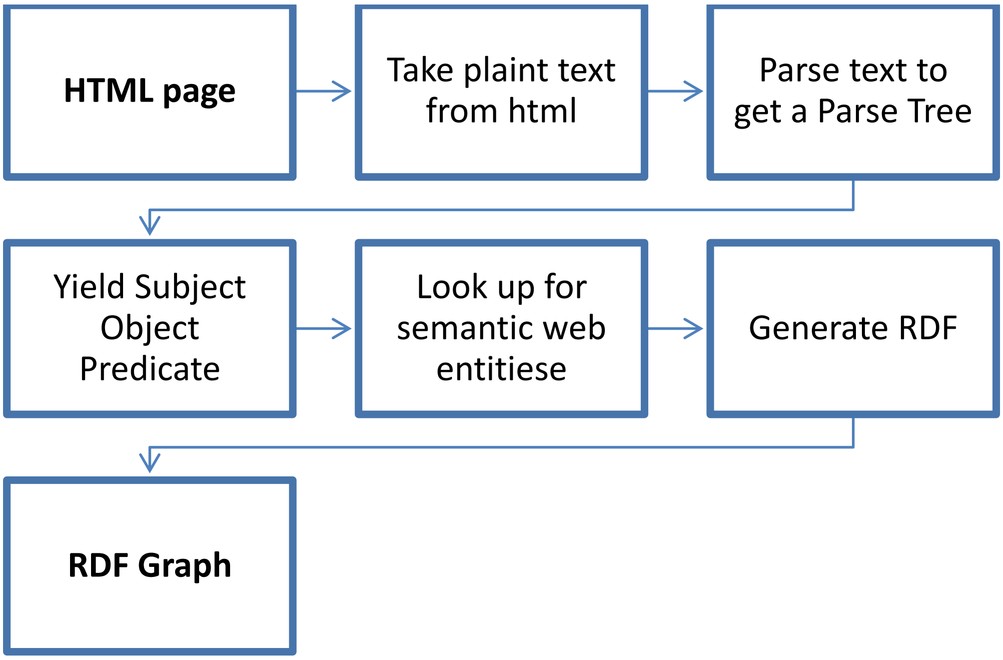

HTML page
Take plaint text from html
Parse text to get a Parse Tree
Yield Subject Object Predicate
Look up for semantic web entitiese
Generate RDF
RDF Graph

*Figure 2-7: Research Methodology for an Approach to Emerge Semantic web*

Our suggested framework is titled RDF by Structured Reference to Semantics or RS2. Target of this framework is to make RDF graph from html webpage. To do this, RS2 extracts the plaintext from html web document and parse natural language into a structured syntax tree. This tree is again analyzed to understand semantics from the syntax. Semantics of each sentence is represented by a tuple of Subject-Predicate-Object. This tuple is then gone through mapping and relating of Semantic Web references. After all these works, we will get a set of RDF triple for the text being processed. Then, web document's semantic information is stored in an RDF graph for providing Semantic Web features.

Chapter 3 (RDF by Structured Reference to Semantics) on page 24 has detailed description of the framework.

## 2.6 Summary

This chapter is of particular importance as it provides the reader with some basic knowledge about the research background. As Semantic Web is not an existing but an emerging technology, the notion of it is not too familiar to everyone. This chapter helps comprehending the current status of Semantic Web and related technologies. In this chapter, we have described about some implementation approaches of Semantic Web, reviewed a related work and briefly introduced our research methodology.

# Chapter 3

# RDF by Structured Reference to Semantics

## 3.1 Introduction

> *In addition to the classic "Web of documents" W3C is helping to build a technology stack to support a "Web of data," the sort of data you find in databases. The ultimate goal of the Web of data is to enable computers to do more useful work and to develop systems that can support trusted interactions over the network. The term "Semantic Web" refers to W3C's vision of the Web of linked data. Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies such as RDF, SPARQL and OWL*
>
> *—W3C Semantic Web Standards*

To emerge Semantic Web, one main bottleneck is to convert existing web documents from human readable natural language texts to machine intelligible web data. We suggest that existing html documents will necessarily not be eliminated[*] but RDF graph will be generated from these documents. We suggest a framework that will take input of html document and generate an RDF graph based on the semantics yielded from NL parsing the text and mapping them into existing ontology. We call this framework as *RDF by Structured Reference to Semantics* or ***RS2***[†].

We used an NL parser for English grammar and also developed a prototype application with a tiny ontology which now covers English Premier League. This chapter will concentrate on describing the framework and the application.

[*] In fact html will persist for text formatting and document presentation

[†] Stylized abbreviation of *RDF by Structured Reference to Semantics*, RSRS = (RS)$^2$

## 3.2 The RS2 framework

RS2 framework suggests a serialized approach, as depicted in the figure below.

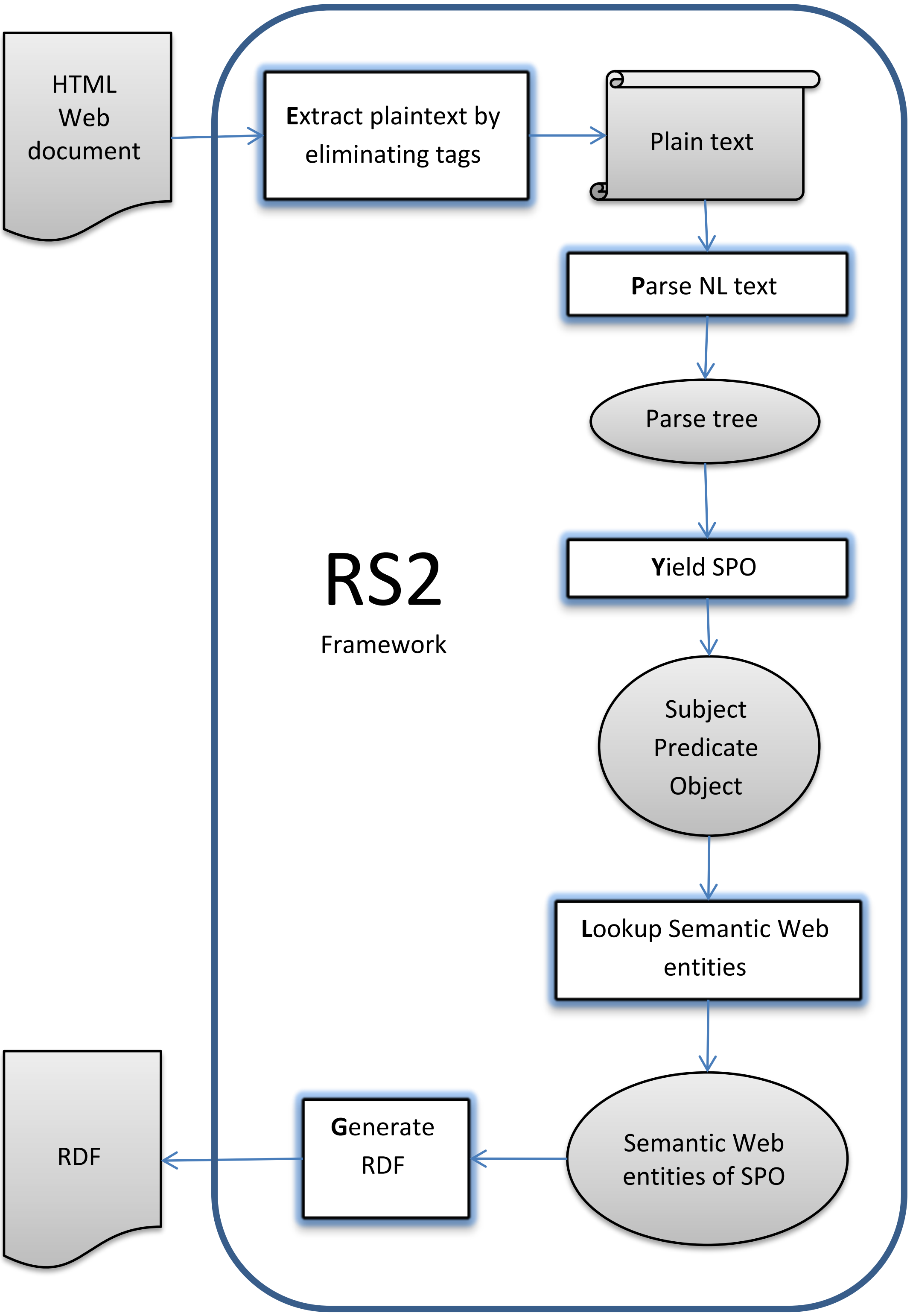


*Figure 3-1: a flow chart showing the top level functions and elements of the RS2 framework*

RS2 framework has five main phases:

$$Extract - Parse - Yield - Lookup - Generate$$

Before going deep into the details of RS2 framework let us first have a brief overview of the purposes of each phase.

**Extract** natural language text by eliminating html tags

Existing html web documents hold some browser-recognized presentation and text formatting information along with the human readable natural language text. We only need the NL text only, not the formatting information. Hence, in the very first phase all html tags are stripped of and a plaintext is extracted.

*Input in this phase is an html file and output is plaintext*

**Parse** NL sentences according a grammar into a parse tree

NL text is obviously of a language, and each language has its own grammar and lexicon. An NL parser is to be employed to understand natural language and generate preliminary syntax tree. This syntax tree will be useful in understanding the underlying semantics

*Input in this phase is plaintext and the output is parse tree*

**Yield** atomic Subject-Predicate-Object tuples

Most NL sentences are actually tuple of subject, predicate and object, often noted as a SPO i.e. mathematically a sentence is a function of subject and object while the predicate is defined by the function. In Semantic Web's primary language, RDF, the primitive unit of sensible knowledge is a triple of SPO.

*Input in this phase is parse tree of a sentence and output is SPO tuple*

**Lookup** for semantic equivalent to each constituent in an SPO tuple

SPO tuple generated in the previous stage is still in NL. To generate machine intelligible SPO, the predicate is looked up for equivalent Semantic Web entry in a predicate mapper. Then, subject and object are prepared as suggested by the lookup.

*Input in this phase is NL SPO tuple and output is a set of Semantic Web entities holding the semantics of original SPO*

**Generate** Resource Description Framework graph

In lookup phase all Semantic Web data are gathered. Now it is time to get the RDF graph and let machines comprehend human knowledge.

*Input in this phase is raw semantic data and output is RDF graph*

Extract, Parse, Yield, Lookup and Generation — these five phases altogether form the RS2 framework. In short RS2 takes input of html and gives output of RDF. In realization of this framework, it will necessitate several other elements:

- An lexicon — an XML graph that holds all the words and phrases used in a language
- Mapper — an XML graph that maps NL Phrase to Semantic Web reference (URI)
- Ontology — the concept map on which semantics are comprehended

The figure below depicts an RS2 framework in action,

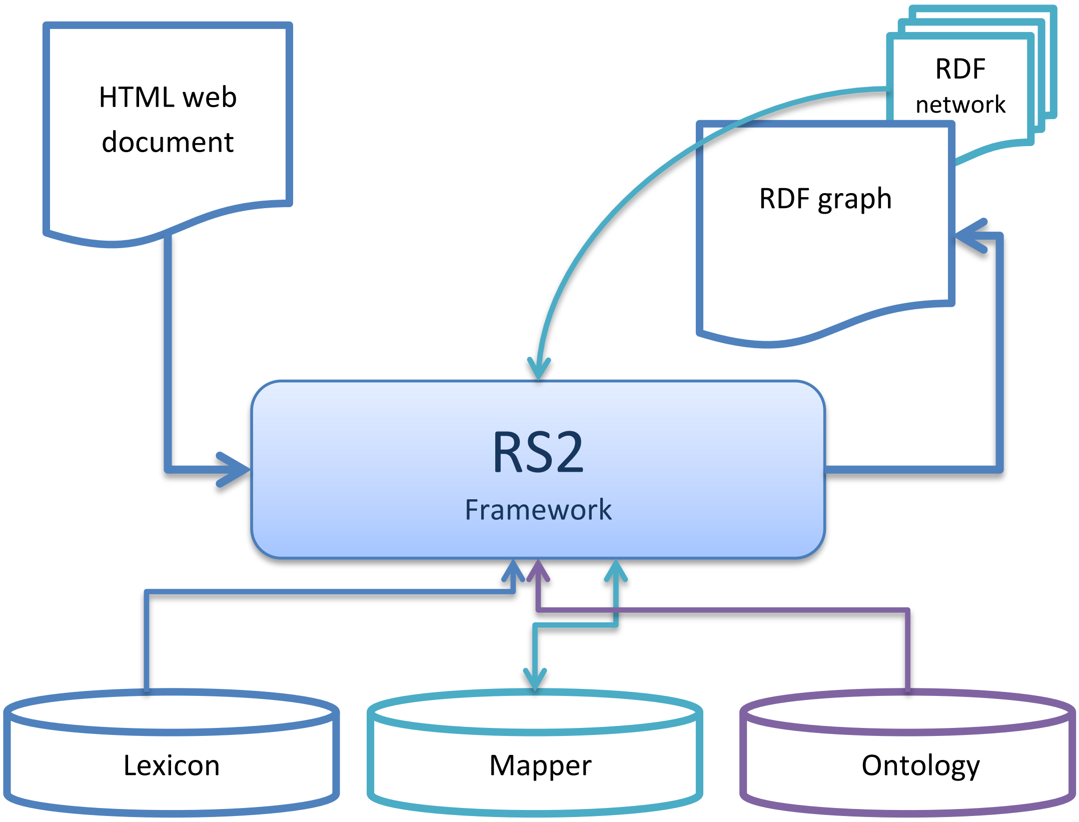


*Figure 3-2: RS2 framework in action.* RS2 framework reads NL words from lexicon, maps NL SPO to Semantic Web entity by reading from mapper, in terms RS2 may write new term in mapper. it also reads from ontology to generate RDF

In next sections of this chapter we will focus on the five phases of RS2 framework and describe them along with practical implementation scenario.

## 3.3 Extraction of plaintext from html

Each web document contains lots of unnecessary character. It may contain some non-ASCII character. To maintain the controls of the web document these non-ASCII characters are used. Most used non-ASCII character is no-break space (ASCII 160). But to Natural language does not contain ASCII not more than 127. So these unnecessary characters are removed for NLP.

To convert the web document the extinction of the document is checked. To ensure the web document the extension can be checked. It may be .Html or .ASPX or .PHP file. The documents Html validation is checked. It should obey all the XHTML validation. As we know, the XHML is the cleaner and stricter version of HTML. From valid XHTML document text can be extracted using regular expression. HTML tags are matched using regular expression and the matched text is eliminated. We used the following expression,

$$< (.|\backslash n)+? >$$

All the tag is removed and the plain text is stored. Trim operation is done to the plain text to remove the new line (\n), carriage return (\r), tab (\t), empty character (\0)and space (\s).

As we done the RDF conversion operation on each line of the document, so each line is parsed by detecting punctuations (? Or. ). After parsing each line special character and non-ASCII valued character is removed. Special character and non-ASCII can be detected by using the regular expression as follows:

$$[\backslash \mathrm{u0000} - \backslash \mathrm{u007F}]$$

After removing these special characters the trim operation is done again to ensure the valid format of the sentence.

Finally the array of sentence is checked so the array does not return any empty string.

## 3.4 Parse

Human beings comprehend plaintext written in some natural language. Web documents contain plaintext included into many formatting information. In previous phase, plaintext from the html web document is extracted. This plaintext is yet not intelligible to machine. The goal of this phase is to parse the text, written in natural language, into some structure that can be further processed for elicit underlying semantics.

> *'A natural language parser is a program that works out the grammatical structure of sentences, for instance, which groups of words go together (as phrases) and which words are the subject or object of a verb.'*
>
> *— The Stanford Natural Language Processing Group [48]*

For parsing sentences from natural language, several steps are to undergo:

- Separate each sentence, we will parse a sentence at once
- Separate words in the sentence
- POS tagging, find parts of speech of each word from the lexicon
- Try to parse the sentence with a grammar by recognizing parts of speech as input symbols
- If parsed successfully return parse tree (syntax tree)

In next few subsections we will describe how to accomplish these steps for English grammar*.

* Although it is not necessary that, RS2 parse only English sentences, we have chosen this language because there are a lot of studies about NLP for English. Any other approach will fit into RS2 framework if it takes input of sentences and generate parse tree for each sentence.

### 3.4.1 Separation of sentences

English sentence are generally delaminated by punctuations:

- A period '.' followed by an whitespace
- An interrogation mark '?'
- An exclamation mark '!'

We have thus split the sentence using these delimiters.

### 3.4.2 Split words of a sentence

This part is quite simple, just split the sentence by whitespaces after trimming *(eliminate any trailing or ending whitespace and repetitive whitespaces)* the sentence.

### 3.4.3 POS tagging

In this step the parts of speech of every word in a sentence is tagged by looking up the information into a lexicon or thesaurus.

*We have partitioned the lexicon in three sections:*

1. CAPPA (conjunctions, auxiliary verbs, prepositions, pronouns, articles)
2. Name — includes proper nouns and names (case sensitive)
3. VANA (verb, adverb, noun, adjective)

The lexicon is stored as a XML file that has a root `lexicon`, each word is entered as an element named `word`. An element, word has two necessary attributes, `value` (indicates the word) and `pos` (parts of speech of the word).

The skeleton of XML file for lexicon is

```
<lexicon>
  <word value="word1" pos="pos1" />
  <word value="word2" pos="pos2" />
  <word value="word3" pos="pos3" />
  … … …
  … … …
  … … …
</lexicon >
```

*Figure 3-3: skeleton of the lexicon XML*

In optional thesaurus xml, a `word` element have child element called `synonyms`, which store the synonyms of the word. Skeleton for such xml is given below

```
< thesaurus>
<word value="…" pos="…">
      <synonyms>
      …, …, …
      </synonyms>
</word>
</thesaurus>
```

*Figure 3-4: skeleton of the thesaurus XML*

A word s first checked for parts of speech in entries for conjunctions, auxiliary verbs, prepositions, pronouns and articles with a non-case-sensitive way. If it is not in this part, it is checked in the entries of names and proper nouns, this check is case sensitive. If the word is yet not been found, it is next checked into the entries of verb, adverb, noun, adjective. Even if this word is not matched, it means the word is not in the lexicon and tagged as `iname` (is it a name?) and considered same as noun.

Many words act both noun and verb; we tag them as `nounverb` and empirically decide to use as either noun or verb by analyzing the sentence. If a `nounverb` is encountered, it is checked that there is already a verb been found. If there is no other verb then it will be used as verb else as noun.

### 3.4.4 Try to parse

So far, there are several NL parsers being developed. We have chosen the parser developed by *Nazmul Hasnat Arka and Sadre-Ala Parvez of KUET* [49] and enhanced it for our purpose.

The parser used called predictive-backtracking NL parser, which first tries to parse in predictive parsing, if it encounters an error then trace back to an optional production. This parser is built for English grammar defined in table. It takes input of parts of speech tokens for understanding the syntax of the sentence. It stores all successful paths in the process of parse and generated a parse tree or syntax tree as a top down traverse. Details of this parsers structured is not in the scope of our research. Hence we are just going to include short informative descriptions of the parser, in appendix A.

### 3.4.5 Return parse tree

If the sentence is parsed successfully then the parser generates a parse tree and passes it to next phase.

For example, the sentence

*Chelsea football club located in London*

Is parsed and the following tree is generated,

```
A->CFB
C->aE
E->aE
E->aE
E->z
F->MG
M->cN
N->z
G->I
I->J
J->fK
K->CL
C->aE
E->z
L->z
B->z
```

*Figure 3-5: parse tree for the sentence “Chelsea football club located in London”*

This parsed tree will be analyzed to yield the Subject, Predicate and Object.

## 3.5 Yield

The parser generates a parse tree. This parse tree is analyzed for finding out the predicate and associated subject and object.

Algorithms for getting these are included below,

The first parsed noun phrase is the **subject**, we can get it by

- lookup for rule in parse tree that starts with Noun Phrase;
- traverse from that production until null;
- return traversed string;

**Predicate** is yielded from the verb phrase. We extracted a portion of verb phrase depending on the structure of verb phrase. Generally, predicate in natural language that conform to a Semantic Web predicate, do not go further than the preposition, followed by the object. So, the parsed traversal, from Verb phrase to the preposition is returned as predicate. In case, there is an article in verb phrase, the predicate will then be traversed up to the article.

The **object** is a noun phrase within the verb phrase. So, we parse the verb phrase with the starting symbol for verb phrase and get the noun phrase by traversing the parse tree of verb phrase from the non-terminal NP or noun phrase.

For example, the sentence parsed in *figure 4—5* yields subject object predicate as,

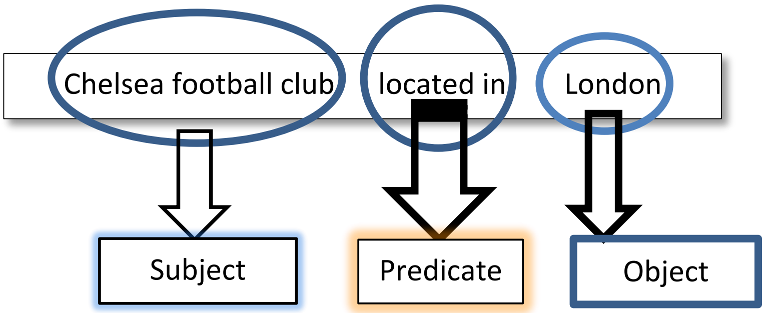


*Figure 3-6: Subject object and predicate*

From these tuple of SPO, we will generate RDF in sections 3.5 and **Error! Reference source not found.**

## 3.6 Lookup

SPO tuple generated in YIELD stage is not compatible to formulate a RDF semantic. Generated SPO by Natural Language Parsing must be attached to syntax. This semantics is defined as a mapping on the Natural Language SPO to the RDF Triple in a machine accessible way.

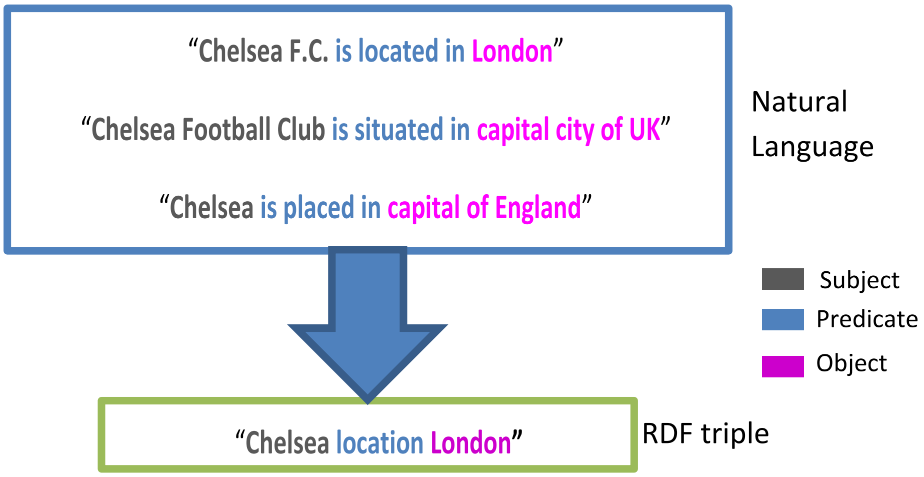

"Chelsea F.C. is located in London"
"Chelsea Football Club is situated in capital city of UK"
"Chelsea is placed in capital of England"
Natural Language
Subject
Predicate
Object
"Chelsea location London"
RDF triple

*Figure 3-7: Mapping from Natural Language SPO to RDF Triple.*

In Figure 3-7 above it is shown that, subject, predicate and object in natural language refer to same semantic but their lexical structure may be different. The subjects Chelsea F.C, Chelsea Football Club and Chelsea refer to a football club which is named after Chelsea. Same confusion is occurred for predicate and object. A machine cannot distinguish between different lexical structured SPO which are referring to same instance.

In order to avoid these anomalies a Lookup procedure is applied to convert the natural language SPO to RDF triple. An xml SPO mapper is used to map to RDF triple, which can be understood by the machine. Now it is an ease for a machine to determine the semantic of the SPO.

```
<predicateMapper>
  <predicate nl="is located in" owl="location" />
  <predicate nl=" is situated in" owl="location" />
  <predicate nl=" is placed in" owl="location" />
  … … …
  … … …
</predicateMapper>
```

*Figure 3-8: Predicate mapper XML file.*

In Figure 3-8, a natural language predicate is enclosed with predicate tag and every predicate tag has two attribute. First one is ‘`nl`’ attribute, contains the value of natural language predicate. Second and last one is ‘`owl`’ attribute, contains the value of the predicate which is accessible by the machine and is defined in an owl graph described in Appendix B (Ontology of English Premier League).

## 3.7 Generate

The generation of RDF triple is depicts by the figure below,

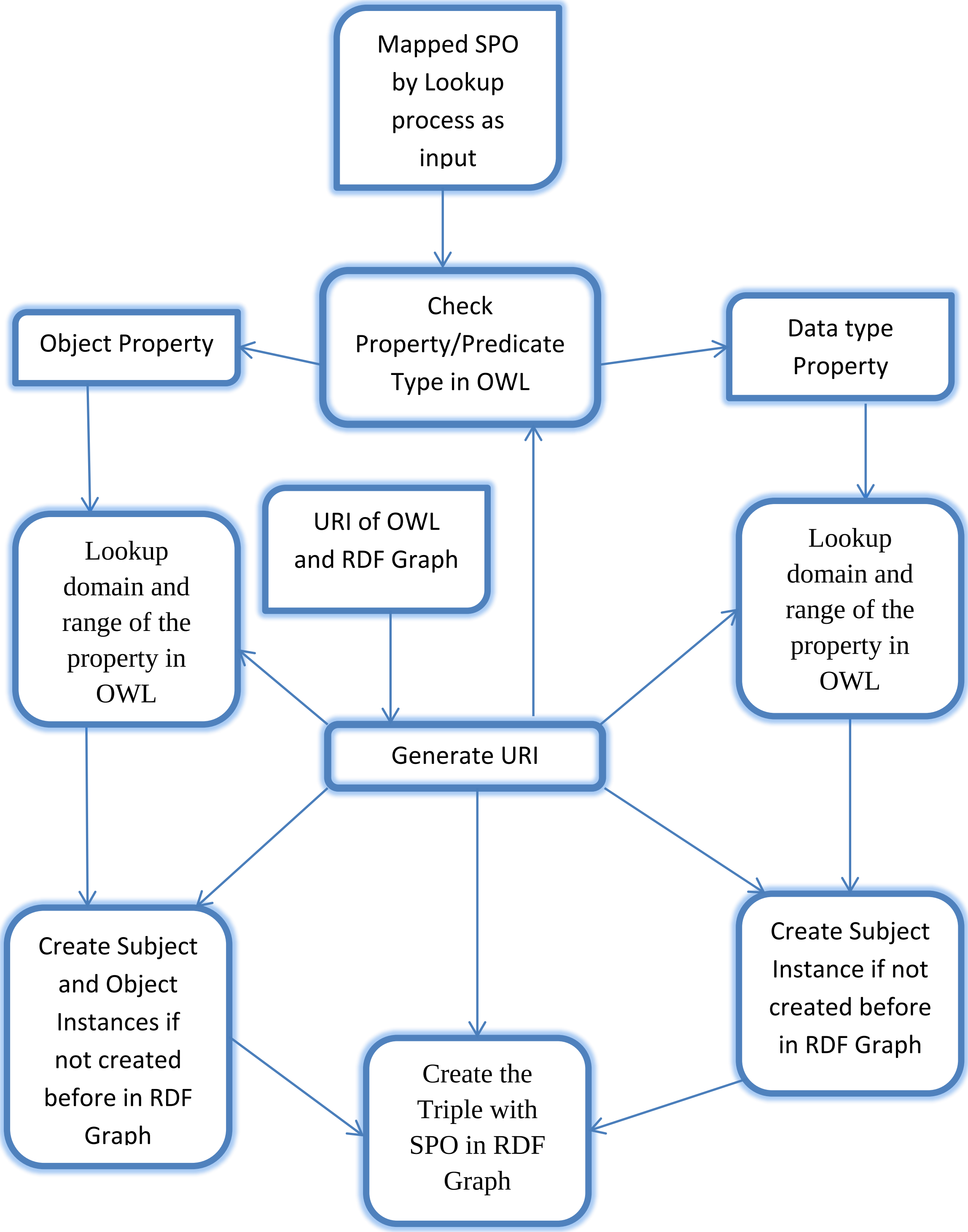

Mapped SPO by Lookup process as input
Check Property/Predicate Type in OWL
Object Property
Data type Property
Lookup domain and range of the property in OWL
URI of OWL and RDF Graph
Lookup domain and range of the property in OWL
Generate URI
Create Subject and Object Instances if not created before in RDF Graph
Create the Triple with SPO in RDF Graph
Create Subject Instance if not created before in RDF Graph

*Figure 3-9: Generate RDF from SPO*

### 3.7.1 Generate URI

Our OWL graph has URI `http://www.owl-ontologies.com/football.owl#` and there is a Base URI `http://www.owl-ontologies.com/` for every RDF graph. The base URI is same for every RDF graph. Now URI of RDF graph, subject, object is generated by following method —

RDF graph URI= Base URI+ HTML File name + '/'

Subject URI= RDF graph URI + '#' + subject

Object URI== RDF graph URI + '#' + object

URI of predicate is generated from the URI of owl graph.

Property URI=owl file URI + property

### 3.7.2 Check Property/Predicate Type in OWL

Now the type of the predicate is checked in the owl file. We checked in owl graph that a property is either `Object Property` or `Data type Property`. It is done by the following query —

```
SPARQL SELECT ?propertyType from
<http://www.owl-ontologies.com/football.owl#> WHERE{
<http://www.owl-ontologies.com/football.owl#property>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?propertyType
};
```

### 3.7.3 Look Up Domain and Range in OWL

Properties verify general facts about the members of classes and specific facts about individuals. In order to find out the domain and range following query is executed in SPARQL endpoint —

```
SPARQL SELECT ?dom from
<http://www.owl-ontologies.com/football.owl#>
WHERE{<http://www.owl-ontologies.com/football.owl#property>
<http://www.w3.org/2000/01/rdf-schema#domain> ?dom};
SPARQL SELECT ?range from
<http://www.owl-ontologies.com/football.owl#>
WHERE{<http://www.owl-ontologies.com/football.owl#property>
<http://www.w3.org/2000/01/rdf-schema#range> ?range};
```

### 3.7.4 Create Subject and Object Instances

An instance is introduced by declaring it to be a member of a class. In previous stage the class of the subject and object is found as domain and range respectively. Now the instance of subject and object is declared in the RDF graph.

```
SPARQL INSERT into graph <http://www.owl-
ontologies.com/html_file_name/> {<http://www.owl-
ontologies.com/html_file_name/#subject>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type >
<http://www.owl-ontologies.com/football.owl#domainClass>};

SPARQL INSERT into graph <http://www.owl-
ontologies.com/html_file_name/> {<http://www.owl-
ontologies.com/html_file_name/#object>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type >
<http://www.owl-ontologies.com/football.owl#rangeClass>};
```

For data type property object is not declared in RDF graph.

### 3.7.5 Create Triple with SPO in RDF Graph

At last the triple is created in RDF graph. Data type properties have no range class and its range can be string, date, integer, float etc. So following query is executed for data type property —

```
SPARQL insert into graph <http://www.owl-
ontologies.com/html_file_name/> {<http://www.owl-
ontologies.com/html_file_name/subject> <http://www.owl-
ontologies.com/football.owl#property> ‘data’@en };
```

For object type property the query is —

```
SPARQL insert into graph <http://www.owl-
ontologies.com/html_file_name/> {<http://www.owl-
ontologies.com/html_file_name/subject> <http://www.owl-
ontologies.com/football.owl#property> <http://www.owl-
ontologies.com/html_file_name/object> };
```

## 3.8 Prototype demo

We have built an application as small demonstration of using RDF and OWL. In this application user can see the related concept map on the top of the web page. The concept cloud is shown by making query in the OWL graph.

On the other hand at the bottom of the web application there is a text box. User can put there query in the text box and an answer set is returned. As for Interrogative sentence strong NLP Algorithm is not found so we used semi natural language for making query. User should the question mark sign (?) about which s/he wants to know about.

For Example If user brows about the Chelsea Football Club, then at the concept cloud he can see the topic of English Premier League, Liverpool, players of Chelsea and so on. If the user makes query as: "? Stadium name of Chelsea" that means "What is the stadium name of Manchester United?" it returns the answer Old Trafford.

To do this the predicate is mapped. Then the Subject URI is retrieved from its label tag. Then query is done to determine the objects label using subjects URI and Predicates URI.

This is a small demonstration of using RDF and OWL for retrieving knowledge. But This OWL and RDF graphs play the key role to the field of Semantic web.

## 3.9 Summary

In this chapter, we have described an approach to generate RDF graph from existing HTML document with the help of ontology and a lexicon. We have named our framework *RDF by Structured Reference to Semantics* or *RS2.* It has five major phases: *Extract, Parse, Yield, Lookup* and *Generate.* These phases are described along with schematic diagrams of the framework and illustrative examples. We have built a demo application for illustrative and explanatory purpose — it is also mentioned in brief.

# Chapter 4

# Conclusion

This chapter concludes the discussion presented so far. The goal of this chapter is to summarize our discussion, draw out generalized lessons, solutions from the implementation and outline future research in the area.

## 4.1 Thesis Summary

Web of today is collection of html documents which do not have structure. Without structure a document cannot be processed for machine intelligible semantics. It is necessary for machines to understand web documents for providing intelligent services and enhanced features. For giving structure to web documents these documents are to be processed in a way that every sentence of these documents is organized in subject-predicate-object representation. These subjects and objects and relationship between them are described as a vocabulary in a hierarchy of classes and relationships. A tuple of subject-predicate-object found in natural language is then perceived as an RDF triple by structuring them in accordance to the ontology. Our research suggests the RS2 framework to elicit underlying semantics from html document and generate RDF graph to embed data to the web document.

In chapter 1, we introduced the idea and motivation of Semantic Web and gave a brief history of the evolution of World Wide Web. The problem in emerging Semantic Web and our approach for transforming html to RDF was mentioned.

In chapter 2, we presented the state of Semantic Web today, discussed and compared among the versions of web, stated what people are thinking on web evolution. Overview of some related technologies including Semantic Web stack and its components is given. We mentioned about existing works for Semantic Web development. Some Semantic Web implementation approaches and an overview of a related work are also briefed. Research methodology was also included at the end of this chapter.

In chapter 3, we described our approach *RDF by Structured Reference to Semantics* which take html input and give output of RDF graph with help of lexicon, ontology and mapper in Extract-Parse-Yield-Lookup-Generate phases.

## 4.2 Limitations

### 4.2.1 Limitation of NL parser

The parser we used cannot parse compound sentences. One reason of this is that, the grammar has some flaws. If Grammatical flaws can be eliminated the parser will be able to parse more types of sentences.

A good thing is that, the change of grammar can be implemented by simply editing the XML parse table. Although our parser cannot parse all compound sentences, it seems necessary to note that most of text in web constitutes simple sentence. The lexicon has some problem; if it is enriched and corrected more sentences will be parsed successfully.

### 4.2.2 Limitation of ontology

The ontology we are using is now domain specific and not rich enough. This ontology can be compared to an infant, who has just started to learn. To eliminate this limitation, an application can be built on RS2 framework that will report to user about any term that it does not know. It will ask to user about any new knowledge and can enrich the ontology by getting knowledge from user input and learn about more and more domains.

## 4.3 Future work

RS2 framework will help the emergence of a unified giant global graph of linked data which can enable many features of Semantic Web.

RS2 will help convert the giant collection of html documents to RDF graphs of data and applications can be built with the help of RDF graph occupied in this method.

## 4.4 Summary

In this thesis we have tried to eliminate one of the greatest bottlenecks of the emergence of Semantic Web. We have suggested a framework that will take input of HTML web document and give output of RDF graph of linked data. This will help us convert the web from the horde of documents into the squad of data.

# Bibliography

[1] Jim Hendler, "Web 3.0 Emerging," *Computer*, vol. 42, no. 1, pp. 111-113, Jan. 2009. [Online] Available: http://doi.ieeecomputersociety.org/10.1109/MC.2009.30 [Accessed: 2010, Apr.]

[2] Tim Berners-Lee, James Hendler, and Ora Lassila, "The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities," *Scientific American*, vol. 284, no. 5, pp. 28-37, May 2001. [Online] Available: http://www.scientificamerican.com/article.cfm?id=the-semantic-web [Accessed: 2010, April 07]

[3] Tim Berners-Lee. (1989, March) "The original proposal of the WWW, HTMLized" *www.w3.org* [Online] Available: http://www.w3.org/History/1989/proposal.html [Accessed: 2010, April 07]

[4] W3C, Håkon Wium Lie , and Bert Bos. [CSS Working Group]. (1996, Dec 17) "Cascading Style Sheets, level 1 Specification" [REC-CSS1-20080411](11 Apr 2008) *W3C* [Online] Available: http://www.w3.org/TR/CSS1/ [Accessed: 2010, Apr.]

[5] T. Berners-Lee and D. Connolly. [Network Working Group]. (1995, Nov.) "Hypertext Markup Language - 2.0" [RFC 1866](Obsoleted by: 2854) *The Internet Engineering Task Force* [Online] Available: http://tools.ietf.org/html/rfc1866

[6] Dave Raggett, Arnaud Le Hors, and Ian Jacobs. [HTML Working Group]. (1999, Dec. 24) "HTML 4.01 Specification" [ISO/IEC 15445](W3C HTML Activity) *W3C* [Online] Available: http://www.w3.org/TR/html401/ [Accessed: 2010, Apr.]

[7] Ian Hickson and David Hyatt. [HTML Working Group]. (2010, March 04) "HTML5: A vocabulary and associated APIs for HTML and XHTML" (W3C Working Draft) *W3C* [Online] Available: http://www.w3.org/TR/html5/ [Accessed: 2010, Mar. 4]

[8] HTML Working Group. [W3C HTML Activity]. (2002, Aug. 01) "XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition)" (A Reformulation of HTML 4 in XML 1.0) *W3C* [Online] Available: http://www.w3.org/TR/xhtml1/ [Accessed: 2010, Apr.]

[9] Yahoo! Inc. *Flickr* [Online] Available: http://www.flickr.com/

[10] Wikimedia Foundation. *Wikipedia* [Online] Available: http://www.wikipedia.org/

[11] Digg Inc. (2004, Dec.) *Digg* [Online] Available: http://digg.com/ [Accessed: 2010, Apr.]

[12] Facebook Inc. (2004, Feb.) *Facebook* [Online] Available: http://www.facebook.com/ [Accessed: 2010]

[13] Tim O'Reilly. (2005, Sep. 30) "What Is Web 2.0" *O'Reilly Media* [Online] Available: http://oreilly.com/web2/archive/what-is-web-20.html

[14] World Wide Web Consortium. (1994-2010) "W3C Semantic Web Activity" *W3C* [Online] Available: http://www.w3.org/2001/sw/ [Accessed: 2010, Apr.]

[15] Alexander Mikroyannidis, "Toward a Social Semantic Web," *Computer*, vol. 40, no. 11, pp. 113-115, Nov. 2007. [Online] Available: http://doi.ieeecomputersociety.org/10.1109/MC.2007.405 [Accessed: 2010, Apr.]

[16] Victoria Shannon, "A 'more revolutionary' Web," *International Herald Tribune*, May 2006. [Online] Available: http://www.nytimes.com/2006/05/23/technology/23iht-web.html?_r=1&scp=1&sq=A+'more+revolutionary'+Web%22.&st=nyt [Accessed: 2010, Apr.]

[17] Dan Farber and Larry Dignan. (2006, Nov 15) "TechNet Summit: The new era of innovation" *ZDNet blog* [Online] Available: http://blogs.zdnet.com/BTL/?p=3959

[18] Nova Spivak. (2007, Sep. 14) "Gartner is Wrong about Web 3.0" *Minding the Planet* [Online] Available: http://www.novaspivack.com/technology/gartner-is-wrong-about-web-3-0 [Accessed: 2010, Apr.]

[19] Nova Spivak. (2007, Oct 04) "Web 3.0 -- The Best Official Definition Imaginable" *Minding The Planet* [Online] Available: http://www.novaspivack.com/technology/web-3-0-the-best-official-definition-imaginable [Accessed: 2010, Apr.]

[20] Linden Research, Inc. (2003, June) *Second Life* [Online] Available: http://secondlife.com

[21] Steve Bratt. (2007, Jan. 30) "Semantic Web, and Other Technologies to Watch" *W3C* [Online] Available: http://www.w3.org/2007/Talks/0130-sb-W3CTechSemWeb/#%2824%29 [Accessed: 2010, Apr.]

[22] T. Berners-Lee, R. Fielding, and L. Masinter. [Network Working Group]. (2005, Jan.) "Uniform Resource Identifier (URI): Generic Syntax" *The Internet Society* [Online] Available: http://labs.apache.org/webarch/uri/rfc/rfc3986.html [Accessed: 2010, Apr.]

[23] Dan Connolly and Tim Bernern-Lee. [URI Interest Group]. (1993) "Naming and Addressing: URIs, URLs,." (Revision: 1.58 of Date: 2006/02/27 15:15:52) *W3C* [Online] Available: http://www.w3.org/Addressing/ [Accessed: 2010, Apr.]

[24] M. Duerst and M. Suignard. [Network Working Group]. (2005, Jan.) "Internationalized Resource Identifiers (IRIs)" [RFC 3987]*The Internet Engineering Task Force* [Online] Available: http://www.ietf.org/rfc/rfc3987.txt [Accessed: 2010, Apr.]

[25] Unicode, Inc. *The Unicode Consortium* [Online] Available: http://unicode.org/

[26] W3C. (2010, Mar. 14) "Extensible Markup Language (XML)" *W3C* [Online] Available: http://www.w3.org/XML/ [Accessed: 2010, Apr.]

[27] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, and François Yergeau. [XML Core Working Group]. (2008, Nov. 26) "Extensible Markup Language (XML) 1.0 (Fifth Edition)" *W3C* [Online] Available: http://www.w3.org/TR/2008/REC-xml-20081126/ [Accessed: 2010, Apr.]

[28] Cody Burleson. (2007, Oct 04) "Introduction to the Semantic Web Vision and Technologies - Part 2 - Foundations" *Semantic Focus* [Online] Available: http://www.semanticfocus.com/blog/entry/title/introduction-to-the-semantic-web-vision-and-technologies-part-2-foundations/ [Accessed: 2010, Apr.]

[29] Frank Manola and Eric Miller. [RDF Core Working Group]. (2004, Feb. 10) "RDF Primer" (one document in the set of six, intended to jointly replace the original Resource Description Framework specifications) *W3C* [Online] Available: http://www.w3.org/TR/rdf-primer/ [Accessed: 2010, Apr.]

[30] Dan Brickley and R.V. Guha. [RDF Core Working Group]. (2004, Feb. 10) "RDF Vocabulary Description Language 1.0: RDF Schema" *W3C* [Online] Available: http://www.w3.org/TR/rdf-schema/ [Accessed: 2010, Apr.]

[31] Sean Bechhofer, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein. [Web Ontology Working Group]. (2004, Feb. 10) "OWL Web Ontology Language Reference" (Updated: 2009, Nov. 12) *W3C* [Online] Available: http://www.w3.org/TR/owl-ref/ [Accessed: 2010, Apr.]

[32] Alistair Miles and Sean Bechhofer. [Semantic Web Deployment Working Group]. (2009, Aug. 18) "SKOS Simple Knowledge Organization System" *W3C* [Online] Available: http://www.w3.org/TR/skos-reference [Accessed: 2010, Apr.]

[33] W3C. "SPARQL Current Status" *W3C* [Online] Available: http://www.w3.org/standards/techs/sparql [Accessed: 2010, Apr.]

[34] SPARQL Working Group. (2010, Apr. 22) "SPARQL Working Group Wiki" *W3C* [Online] Available: http://www.w3.org/2009/sparql/wiki/Main_Page [Accessed: 2010, Apr.]

[35] Eric Prud'hommeaux and Andy Seaborne. [SPARQL Working Group]. (2008, Jan. 15) "SPARQL Query Language for RDF" *W3C* [Online] Available: http://www.w3.org/TR/rdf-sparql-query/ [Accessed: 2010, Apr.]

[36] Harold Boley, Gary Hallmark, Michael Kifer, Adrian Paschke, Axel Polleres, and Dave Reynolds. (2009, Oct. 1) "RIF Core Dialect" (W3C Candidate Recommendation) *W3C* [Online] Available: http://www.w3.org/TR/rif-core/ [Accessed: 2010, Apr.]

[37] RIF Working Group. [RIF Working Group]. (2010, Feb. 9) "RIF Working Group" *W3C* [Online] Available: http://www.w3.org/2005/rules/wiki/RIF_Working_Group [Accessed: 2010, Apr.]

[38] Michael Kifer, "Rule Interchange Format: The Framework," in *Web Reasoning and Rule Systems*. Karlsruhe, Germany: Springer Berlin / Heidelberg, Oct. 2008, vol. 5341/2008, ch. 1, pp. 1-11. [Online] Available: http://www.springerlink.com/content/e7v2802743688216/ [Accessed: 2010, Apr.]

[39] Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, and Mike Dean. (2004, May 21) "SWRL: A Semantic Web Rule Language" *W3C* [Online] Available: http://www.w3.org/Submission/SWRL/

[40] Protégé Project, Stanford University, CA, USA. *protégé* [Online] Available: http://protege.stanford.edu/

[41] Chris Bizer, Tobias Gauß, Richard Cyganiak, and Olaf Hartig. [Freie Universität Berlin]. (2009, Nov. 18) "Semantic Web Client Library" (Querying the complete Semantic Web with SPARQL.) *Lehrstuhl für Wirtschaftsinformatik* [Online] Available: http://sites.wiwiss.fu-berlin.de/suhl/bizer/ng4j/semwebclient/ [Accessed: 2010, Apr.]

[42] Arjohn Kampman and Herko ter Horst. [OpenRDF.org]. (1997-2009) "Sesame" *Aduna* [Online] Available: http://sourceforge.net/projects/sesame/ [Accessed: 2010, Apr.]

[43] Oracle. "Oracle 11g Spatial" *Oracle Technology Network* [Online] Available: http://www.oracle.com/technology/tech/semantic_technologies/index.html

[44] "Jena – A Semantic Web Framework for Java" (Managed by 'The HP Semantic Web Team' until October 2009 and now a commercially neutral community project) *OpenJena* [Online] Available: http://openjena.org/ [Accessed: 2010, Apr.]

[45] "Relaxed OWL EXperience" *NATO C3 Agency* [Online] Available: http://rowlex.nc3a.nato.int/ [Accessed: 2010, Apr.]

[46] Zemanta, Ltd. (2006-2009) "Zemanta API" *Zemanta* [Online] Available: http://www.zemanta.com/api/

[47] Debajyoti Mukhopadhyay, Rituparna Kumar, Sourav R. Majumdar, and Subhobroto Sinha, "A New Semantic Web Services to Translate HTML Pages to RDF," presented at the *10th International Conference on Information Technology (ICIT 2007)*, Rourkela, India, Dec. 2007, pp. 292-294 [ISBN: 0-7695-3068-0]. [Online] Available: http://doi.ieeecomputersociety.org/10.1109/ICIT.2007.22 [Accessed: 2010, Apr.]

[48] The Stanford Natural Language Processing Group. (2002-2010) "The Stanford Parser: A statistical parser" *The Stanford NLP Group* [Online] Available: http://nlp.stanford.edu/software/lex-parser.shtml [Accessed: 2010, Apr.]

[49] Nazmul Hasnat and Sadre-Ala Parvez, "Corpus Based Multi-document Text Summarization Using Sentence Similarity Detection Technique," Dept of Computer Science and Engineering, Khulna University of Engineering & Technology, Khulna, BD, Undergraduate Thesis Report 2010.

[50] Tim Berners-Lee, Sandro Hawke, and Dan Connolly. (2004, May 16) "Semantic Web Tutorial Using N3" *W3C* [Online] Available: http://www.w3.org/2000/10/swap/doc/Overview.html [Accessed: 2010, Apr.]

[51] Tim Berners-Lee, *Weaving the Web*. London, UK: Orion Publishing Group, Ltd, 1999.

[52] Andrew Newman. (2007, Mar. 14) "A Relational View of the Semantic Web" *O'Reilly XML.com* [Online] Available: http://www.xml.com/pub/a/2007/03/14/a-relational-view-of-the-semantic-web.html [Accessed: 2010, Apr.]

[53] Michel Klein, "Tutorial : The Semantic Web (XML, RDF, and Relatives)," *IEEE INTELLIGENT SYSTEMS*, pp. 26-28, Mar. 2001.

[54] Michel Klein, "XML, RDF, and Relatives," *IEEE INTELLIGENT SYSTEMS*, vol. 16, no. 2, pp. 26-28, March/April 2001.

[55] Dave Beckett. [RDF Core Working Group]. (2004, Feb. 10) "RDF/XML Syntax Specification (Revised)" (W3C Recommendation) *W3C* [Online] Available: http://www.w3.org/TR/REC-rdf-syntax/ [Accessed: 2010, Apr.]

[56] RDF Working Group. (2004, Feb. 10) "Resource Description Framework (RDF) "*W3C* [Online] Available: http://www.w3.org/RDF/ [Accessed: 2010, Apr.]

[57] Joshua Tauberer. [originally written by Tim Bray in 1998 and updated by Dan Brickley in 2001]. (2006, July 26) "What Is RDF" *O'REILLY XML.com* [Online] Available: http://www.xml.com/pub/a/2001/01/24/rdf.html [Accessed: 2010, Apr.]

[58] A. Swartz. [Network Working Group]. (2004, Sep.) "application/rdf+xml Media Type Registration" [RFC 3870] [Online] Available: ftp://ftp.rfc-editor.org/in-notes/rfc3870.txt [Accessed: 2010, Apr.]

[59] Dave Beckett. (2005, Sep. 23) "Dave Beckett's Resource Description Framework (RDF) Resource Guide" *Planet RDF* [Online] Available: http://planetrdf.com/guide/ [Accessed: 2010, Apr.]

[60] Joshua Tauberer. (2005-2010) *rdf:about* [Online] Available: http://www.rdfabout.com/

[61] Shelley Powers, *Practical RDF*.: O'Reilly Media, Print: July 2003 | eBook: Feb. 2009.

[62] Michael Denny. (2004, July 14) "Ontology Tools Survey, Revisited" *O'Reilly XML.com* [Online] Available: http://www.xml.com/pub/a/2004/07/14/onto.html [Accessed: 2010, Apr.]

[63] Michael K. Smith, Chris Welty, and Deborah L. McGuinness. [Web Ontology Working Group]. (2004, Feb. 10) "OWL Web Ontology Language" (W3C Recommendation) *W3C* [Online] Available: http://www.w3.org/TR/owl-guide/ [Accessed: 2010, Apr.]

[64] W3C OWL Working Group. (2009, Oct. 27) "OWL 2 Web Ontology Language Document Overview" (W3C Recommendation) *W3C* [Online] Available: http://www.w3.org/TR/owl2-overview/ [Accessed: 2010, Apr.]

[65] Lee Feigenbaum and Eric Prud'hommeaux. [Cambridge Semantics]. (2009, June 09) "SPARQL By Example" *Cambridge Semantics* [Online] Available: http://www.cambridgesemantics.com/2008/09/sparql-by-example/ [Accessed: 04, 2010]

[66] Kendall Grant Clark, Lee Feigenbaum, and Elias Torres. [W3C RDF Data Access Working Group]. (2008, Jan. 15) "SPARQL Protocol for RDF" *W3C* [Online] Available: http://www.w3.org/TR/2008/REC-rdf-sparql-protocol-20080115/ [Accessed: 2010, Apr.]

[67] Eric Prud'hommeaux and Andy Seaborne. [RDF Data Access Working Group]. (2008, Jan. 15) "SPARQL Query Language for RDF" *W3C* [Online] Available: http://www.w3.org/TR/rdf-sparql-query/ [Accessed: 2010, Apr.]

[68] Leigh Dodds. (2005, Nov. 16) "Introducing SPARQL: Querying the Semantic Web" *O'REILLY XML.com* [Online] Available: http://www.xml.com/pub/a/2005/11/16/introducing-sparql-querying-semantic-web-tutorial.html [Accessed: 2010, Apr.]

[69] Richard Cyganiak. [Digital Media Systems Laboratory, HP Laboratories Bristol]. (2005, Sep. 25) "A relational algebra for SPARQL" [HPL-2005-170 ](Technical Report) *HP Labs* [Online] Available: http://www.hpl.hp.com/techreports/2005/HPL-2005-170.html [Accessed: 2010, Apr.]

[70] Robert Barta. (2005, June 1) "TMQL: A Brief Introduction" *O'Reilly XML.com* [Online] Available: http://www.xml.com/pub/a/2005/06/01/tmql.html [Accessed: 2010, Apr.]

[71] Peter Mikhalenko. (2005, June 22) "Introducing SKOS" *O'Reilly XML.com* [Online] Available: http://www.xml.com/pub/a/2005/06/22/skos.html [Accessed: 2010, Apr.]

[72] O'Reilly Media, Inc. (2010) "Semantic Web" *O'REILLY XML.com* [Online] Available: http://www.xml.com/semweb/

[73] *Britannica Concise Encyclopædia*, 15th ed. Chicago, Illinois, U.S.: Encyclopædia Britannica, Inc., 1768–present.

[74] Lord Quirk, Della Summers, Adam Gadsby, Michal Rundell, Sue Engineer, Nick Ham, Phil Scholfield, Ingrid Freebairn, Chris Fox, Patrick Gillard, Ted Jackson, Stella o'Shea, and Wendalyn Nichols, *Longman Dictionary of Contemporary English*, 3rd ed. Essex, England: Longman Dictionaries, 2000.

[75] Natalya F. Noy and Deborah L. McGuinness. (2007, May 08) "Ontology Development 101: A Guide to Creating Your First Ontology" *The Protégé Ontology Editor and Knowledge Acquisition System* [Online] Available: http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html

[76] Pham Thi Thu Thuy, Young-Koo Lee, Sungyoung Lee, and Byeong-Soo Jeong, "Exploiting XML Schema for Interpreting XML Documents as RDF," presented at the *2008 IEEE International Conference on Services Computing Vol. 2*, July 2008, vol. 2, pp. 555-558 [ISBN: 978-0-7695-3283-7; DOI 10.1109/SCC.2008.93]. [Online] Available: http://doi.ieeecomputersociety.org/10.1109/SCC.2008.93

[77] Jim Hendler, "Linked Data, Web 3.0 and the Semantic Web," , Dec. 2005 [ISBN: 978-0-7695-3401-5]. [Online] Available: http://doi.ieeecomputersociety.org/10.1109/SKG.2008.102

[78] Danny Ayers, "Delivered Deliverables: The State of the Semantic Web, Part 1," *Internet Computing*, vol. 13, no. 1, pp. 86-89, Jan. 2009.

# APPENDICES

## A. Predictive Backtracking Natural Language Parser

### A.1 Productions for input symbols

This parser first generates input tokens by checking parts of speech

```
a → noun | name | iname
b → pronoun
c → verb
d → adjective
e → adverb
f → preposition
g → conjunction
h → article
i → degree
j → gerund
k → auxiliary verb
```

N.B.: *iname is used to denote those words not found in lexicon. It is high probable that this word is then a Proper Noun.*

## A.2 The grammar

The production rule is defined as,

**S → $N_P$ $V_P$ | $V_P$ | S Conj S**
**$N_P$ → (det) *n* | pron |(det) adj *n*| $N_P$ $V_P$| $N_P$ Conj $P_P$ | gerund| $P_P$ $N_P$**
**$V_P$ → V | V $N_P$ | V $P_P$ | $V_P$ Conj $V_P$**
**$P_p$ → prep $N_P$ | prep ($V_P$) | $P_P$ Conj $P_P$**
**V → V adv |verb | aux (v)**
**Adj → (deg) adj (adj)| adj Conj adj**
**Conj→ Conj adv | adv Conj | conj**

Here,

**S** is sentence
**$N_P$** is Noun Phrase
***n*** is Noun
**pron** is Pronoun
**det** is determinant
**adj** is adjective
**V** is verb
**aux** is auxiliary verb
**conj** is conjuction

## A.3 Production table

Production table for the grammar of B.2

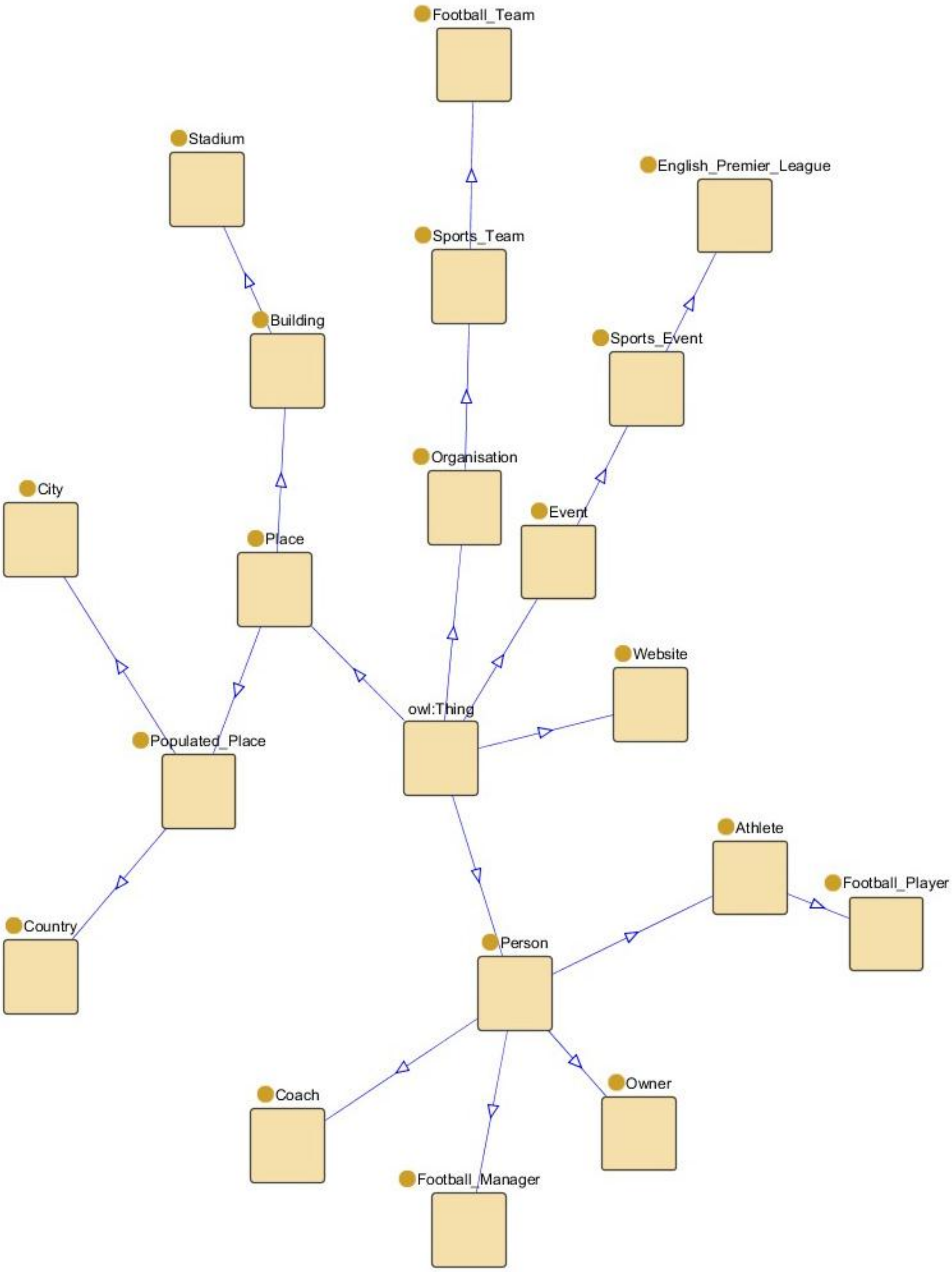
| input<br>non-terminal | a | b | c | d | e | f | g | h | i | j | k | l |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A = S | CFB | CFB | FB | CFB | err | CFB | err | CFB | CFB | CFB | FB | err |
| B = S’ | | | | | RAB | | RAB | | | | | Z |
| C = NP | aE | bE | | OaE | | JE | | hD | OaE | jE,JE | | |
| D = $N_{P1}$ | aE | | | OaE | | | | | OaE | | | |
| E = $N_P$’ | aE | | z | | RCE,z | JE,z | RCE,z | | | z,JE | z | z |
| F = $V_P$ | | | MG | | | | | | | | MG | |
| G = $V_{P1}$ | CI | CI | | CI | I | I,CI | I | CI | CI | I,CI | | Z |
| H = $V_{P2}$ | N | N | N,MN | N | N | N | N | N | N | N | N,MN | N |
| I = $V_P$’ | | | | | RFI,z | J | RFI,z | | | z | | z |
| J = $P_P$ | | | | | | fK | | | | jJL | | |
| K = $P_{P1}$ | CL | CL | | CL | | CL,L | L | CL | CL | CL,L | FL,L | L |
| L = $P_P$’ | | | z | | RJL,z | z | RJL,z | | | z | z | z |
| M = VP | | | cN | | | | | | | | kH | |
| N = V’ | z | z | z | z | eN,z | z | z | z | z | z | z | z |
| O = AdJ | | | | dP | | | | | idP | | | |
| P = Adj1 | Q | | | OQ | Q | | Q | | OQ | | | |
| Q = Adj’ | z | | | | ROQ,z | | ROQ,z | | | | | |
| R = Conj | | | | | eRS | | gS | | | | | |
| S = Conj’ | z | z | z | z | eS | z | | z | z | z | z | |

Parse table is defined and stored into XML. Following figure show a snippet of the xml file, with all production from B,

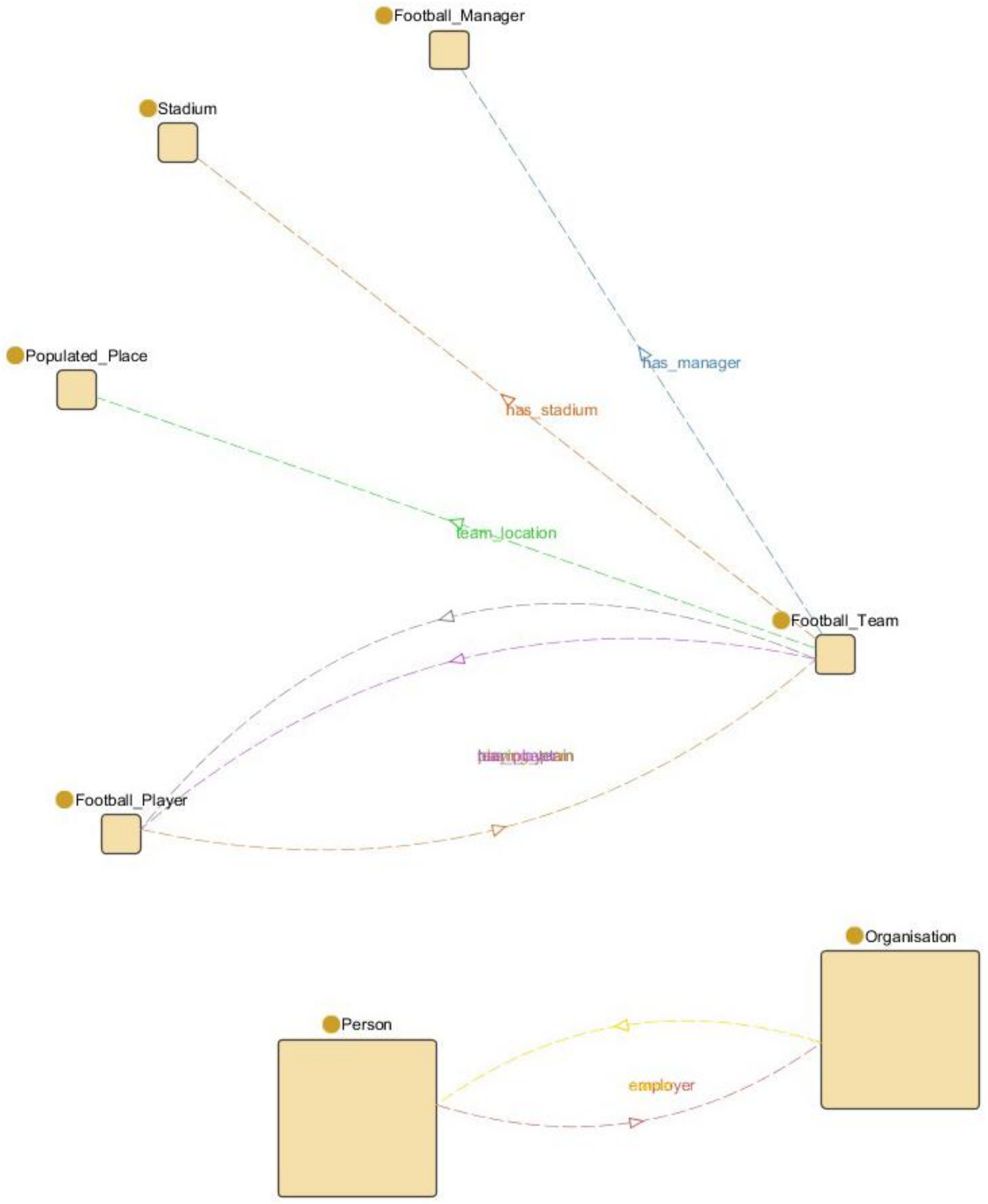
```
<parseTable>
      <nonTerm id="A">
            <production input="a">CFB</production>
            <production input="b">CFB</production>
            <production input="c">FB</production>
            <production input="d">CFB</production>
            <production input="e">error</production>
            <production input="f">CFB</production>
            <production input="g">error</production>
            <production input="h">CFB</production>
            <production input="i">CFB</production>
            <production input="j">CFB</production>
            <production input="k">FB</production>
            <production input="l">error</production>
      </nonTerm>
      <nonTerm id="B">
      … … …
      </nonTerm>
</parseTable>
```

# B. Ontology of English Premier League

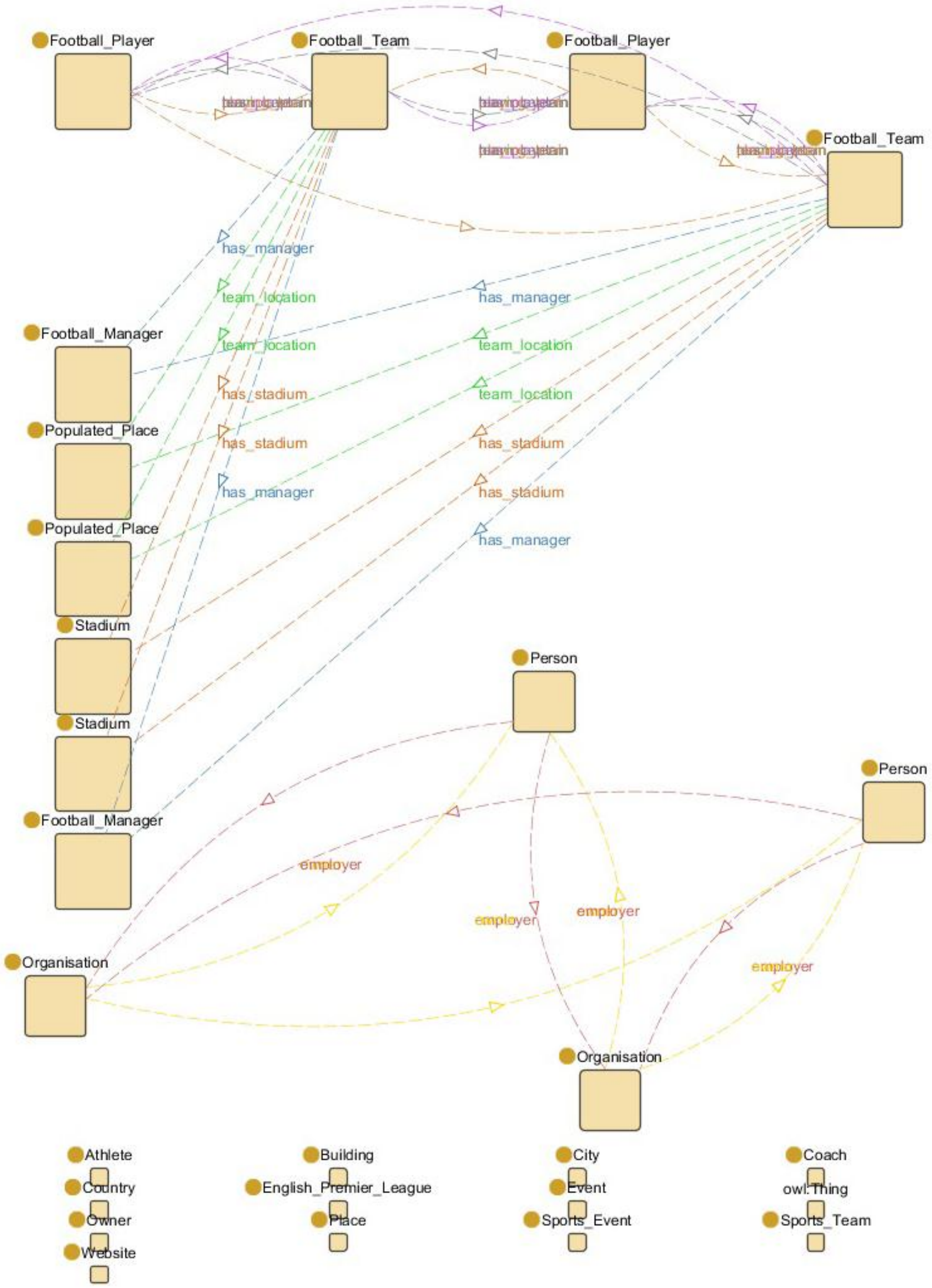
## B.1 Ontology graph



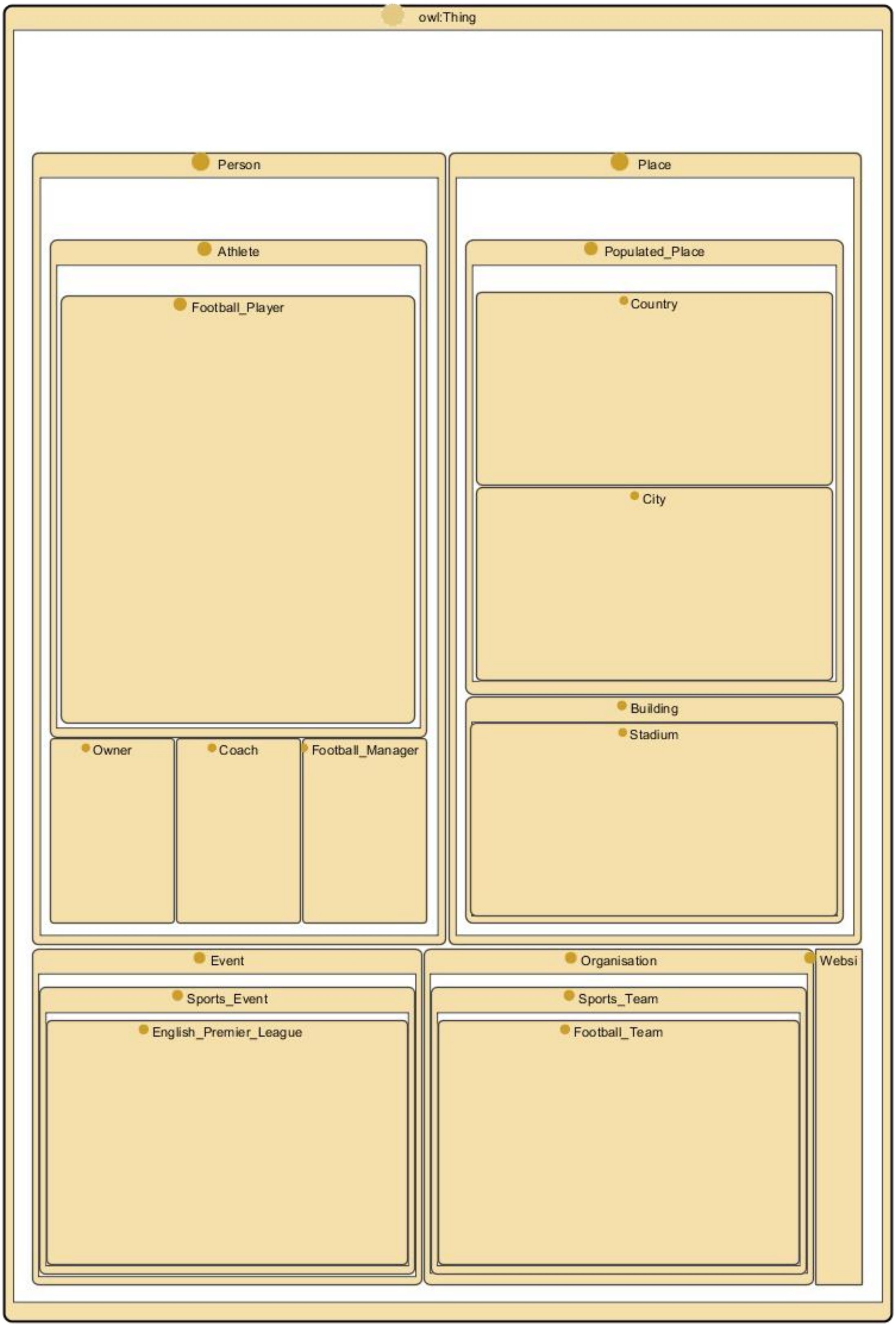
## B.2 Domain and range of predicates



## B.3 Nested composite view of the ontology



## B.4 Nested tree map of the ontology graph


owl:Thing
Person
Athlete
Football_Player
Owner
Coach
Football_Manager
Place
Populated_Place
Country
City
Building
Stadium
Event
Sports_Event
English_Premier_League
Organisation
Sports_Team
Football_Team
Websi