

Chapter 3.19

RDF and OWL

Gian Piero Zarri

University of Paris IV/Sorbonne, France

INTRODUCTION

As Web-based content becomes an increasingly important knowledge management resource, Web-based technologies are developing to help harness that resource in a more effective way.

The current state of these Web-based technology—the “first generation” or “syntactic” Web—gives rise to well-known, serious problems when trying to accomplish in a non-trivial way essential management tasks like indexing, searching, extracting, maintaining, and generating information. These tasks would, in fact, require some sort of “deep understanding” of the information dealt with: In a “syntactic” Web context, on the contrary, computers are only used as tools for posting and rendering information by brute force. Faced with this situation, Berners-Lee first proposed a sort of “Semantic Web” where the access to information is based mainly on the processing of the semantic properties of this information: “... the Semantic Web is an extension of the current Web in which information is given well-defined meaning [emphasis added],

better enabling computers and people to work in co-operation” (Berners-Lee, Hendler, & Lassila, 2001, p. 35). The Semantic Web’s challenge consists then in being able to manage information on the Web by “understanding” its proper semantic content (its meaning), and not simply by matching some keywords.

BACKGROUND

The architecture proposed by Berners-Lee for the Semantic Web is reproduced in Figure 1. “Unicode” and “URI” make up the basis of this hierarchy. The Unicode Standard provides a unique numerical code for every character that can be found in documents produced according to any possible language, no matter what the hardware and software used to deal with such documents. Uniform Resource Identifier (URI) represents a generalization of the well-known Uniform Resource Locator (URL) that is used to identify a “Web resource” (e.g., a particular page) by denoting its primary access mechanism

(essentially, its “location” on the network). URI has been created to allow recording information about all those “notions” that, unlike Web pages, do not have network locations or URLs, but that need to be referred to in an RDF statement. These notions include network-accessible things, such as an electronic document or an image, things that are not network-accessible, such as human beings, corporations, and bound books in a library, or abstract concepts like the concept of a “creator.”

XML (eXtensible markup language) (see Bray, Paoli, Sperberg-McQueen, Maler, & Yergeau, 2004) has been created to overcome some difficulties proper to hypertext markup language (HTML); this last suffers from a number of limitations, from its lack of efficiency in handling the complex client/server communication of today’s applications to the impossibility of defining new tags to customize exactly the user’s needs. XML is called “extensible” because, at the difference of HTML, it is not characterized by a fixed format, but it lets the user design its own customized markup languages (a specific DTD, document type description) for limitless different types of documents; XML is a “content-oriented” markup tool. Basically, the syntactic structure of XML is very simple. Its markup elements are normally identified by an opening and a closing tag, like `<employees>` and `</employees>`, and may contain other elements or text; the elements must be properly nested and every XML document must have exactly one root element. Markup elements can be specialized by adding attribute/value pairs inside the opening tag of the element, like `<person name=“Jane”>`; taking into account the nesting constraint, a very simple fragment of XML document could then be represented as: `<employees><person name=“Jane”><id>99276</id></person></employees>`. To allow a computer interpreting correctly a fragment like this, it is necessary, however, to specify the semantics of the markup elements and tags used to make it; a simple way of doing this is to make use of a DTD.

A DTD is a formal description in XML Declaration Syntax of a particular type of document: for example, a DTD may specify that every person markup element must have a name attribute, and that it can have an offspring element called id whose content must be text. There are many sorts of DTDs ready to be used in all kinds of areas (e.g., www.w3.org/QA/2002/04/valid-dtd-list.html#full) that can be downloaded and used freely: some of them are MathML (for mathematical expressions), Sync Multimedia Integration Language (SMIL), Chemical Markup Language (CML), Open Software Description (OSD), Electronic Data Interchange (EDI), Platform for Internet Content Selection (PICS), and so forth. A more complete way of specifying the semantics of a set of XML markup elements is to make use of XML Schema (as mentioned in Figure 1): XML Schema (Thompson, Beech, Maloney, & Mendelsohn, 2001; Biron & Malhotra, 2001) supplies a more complete grammar for specifying the structure of the elements allowing, for example, to define the cardinality of the offspring elements, default values, and so forth.

MAIN FOCUS OF THE ARTICLE: RDF AND OWL

Moving up in the structure of Figure 1, we find now Resource Description Framework (RDF), an example of “metadata” language (metadata = data about data) used to describe generic “things” (“resources,” according to the RDF jargon) on the Web. An RDF document is basically a list of statements under the form of triples having the classical format: `<object, property, value>`, where the elements of the triples can be Universal Resource Identifiers (URIs), literals (mainly, free text), and variables. To follow a well-known RDF example (Manola & Miller, 2004), let us suppose we want to represent a situation where someone named John Smith has created a particular Web page. We will then make use of the RDF triple:

12 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/rdf-owl/25175

Related Content

Appreciative Sharing for Organizational Knowledge Work

Kam Hou Vat (2011). *Encyclopedia of Knowledge Management, Second Edition* (pp. 27-38).

www.irma-international.org/chapter/appreciative-sharing-organizational-knowledge-work/48955

Measuring Knowledge Management/Knowledge Sharing (KM/KS) Efficiency and Effectiveness in Enterprise Networks

Anirban Ganguly, Ali Mostashariand Mo Mansouri (2011). *International Journal of Knowledge Management* (pp. 37-54).

www.irma-international.org/article/measuring-knowledge-management-knowledge-sharing/59908

Modeling Operational Robustness and Resiliency with High-Level Petri Nets

Madjid Tavana, Timothy E. Buschand Eleanor L. Davis (2013). *Intelligence Methods and Systems Advancements for Knowledge-Based Business* (pp. 170-191).

www.irma-international.org/chapter/modeling-operational-robustness-resiliency-high/67723

Soil Quality Prediction in Context Learning Approaches Using Deep Learning and Blockchain for Smart Agriculture

Parvataneni Rajendra Kumar, S. Meenakshi, S. Shalini, S. Rukmani Deviand S. Boopathi (2023). *Effective AI, Blockchain, and E-Governance Applications for Knowledge Discovery and Management* (pp. 1-26).

www.irma-international.org/chapter/soil-quality-prediction-in-context-learning-approaches-using-deep-learning-and-blockchain-for-smart-agriculture/331226

Becoming a Swarm Catalyst

Juhana Kokkonen, Sami Paavolaand Yrjö Engeström (2013). *International Journal of Knowledge-Based Organizations* (pp. 57-70).

www.irma-international.org/article/becoming-swarm-catalyst/76325