The Potential of RDF/RDFa for eCommerce Applications

John Hines  
Hines IT

John Seydel  
Arkansas State University

Department of Computer & Information Technology  
Box 130  
State University, AR

(870) 680-8072

jhines@hinesit.com  
jseydel@astate.edu
The Potential of RDF/RDFa for eCommerce Applications

Abstract

Moving toward an intelligent web of self-describing information is a goal that has been around nearly since the inception of the World Wide Web, yet no consensus exists regarding how to achieve that goal. Recently, however, the movement has gained renewed momentum with the development of two potentially paradigm-shifting methodologies: the resource description framework (RDF/RDFa) and microformats (muF). This paper addresses the former by describing it and then considering barriers to implementation.

Introduction

The concept, or idea, behind Web 3.0 has recently been the topic of much debate. Different visionaries view the future Web (Web 3.0) as differently as anyone could predict the future. As early as 1999, Sir Tim Berners-Lee (Berners-Lee T., 1999) envisioned a new way to find data, through an intelligent Web that had semantic properties. Today, the Web has become more intelligent through the use of the Resource Description Framework (RDF/RDFa) (W3C, Bridging the Human and Data Webs, 2008). Web applications, such as Microsoft’s relatively new Bing search engine (Microsoft, 2009), are currently using RDFa data to power more intelligent searches. This research addresses the history, current uses, and advances in Resource Description Framework usage; also, any barriers to implementation that may exist will be examined.

Overview

The earliest form of RDF, called Meta Content Framework (MCF), was developed between 1995 and 1997 by Ramanathan V. Guha at Apple Computer. The research project was discontinued and Guha adapted MCF to include XML, thus creating the first version of the Resource Description Framework for Netscape (Andreessen, 1999). The Resource Description Framework, or RDF, was developed in order to ease the formatting of data. That is, the data about your resources as well as the data of others. RDF bundles this concept with the idea of making these statements in the form of triples, or, subject-predicate-object expressions (W3C, RDF/XML Syntax Specification, 2004). The subject denotes the resource that is being shared, the predicate denotes the traits or aspects of the subject, and the object denotes the thing that is being described. An example is provided in Figure 1.

```
"Arkansas State University is located in Jonesboro, Arkansas"
```

| Subject = Arkansas State University | Predicate = is located in | Object = Jonesboro, Arkansas |

Figure 1: The formation of a triple from a simple sentence.
A collection of these RDF statements intrinsically represents a labeled multidigraph, a directed graph which is permitted to have arcs with the same source and target destinations. The RDF-based data model is more suited to certain types of knowledge representation than more traditional relational models or other types of ontological models. RDF data is often cataloged in relational databases or native representations called “triple stores” (Harth & Decker, 2005). The World Wide Web Consortium (W3C) published the first specification of RDF’s data model and XML syntax as a recommendation in 1999 (W3C, Resource Description Framework, 1999). In 2004, a set of related specifications was published called the RDF/XML Syntax Specification (W3C, RDF/XML Syntax Specification, 2004). The subject-predicate-object triples used for RDF identification in applications of RDF, such as Really Simple Syndication (RSS) or Friend of a Friend (FOAF) implementations in social networking sites, are usually represented in the form of Uniform Resource Identifiers (URIs) that reference actual data on the Web, however, a URI that names a resource does not necessarily have to be able to be dereferenced. That is, a URI that begins by naming the Hypertext Transfer Protocol (HTTP) location of a resource does not have to be accessible by utilizing HTTP.

Cataloged RDF data has two common forms of serialization, or, processes of converting an object into sequences of bits so that data can be stored upon storage mediums; the first is an XML format (Swartz, 2004) and the latter, called Notation 3 (N3) (Berners-Lee T. , Notation 3 (N3), 2006), is a non-XML based format that was designed to be easier to author from memory and easier to follow when reading. The XML-type format is often called RDF as it was introduced among the RDF definition. The format is often confused with the RDF model which is known as serializing RDF as XML. The recommended usage is that RDF documents follow the 2004 W3C specification (Swartz, 2004). The non-XML format used in Notation 3 is based upon tabular notation which makes the underlying triples encoded in documents easily recognizable when compared the XML serialization of RDF.

Ultimately, knowledge collections that are gained by the use of RDF can undergo reification. Each statement, or more precisely each complete triple; can be treated as a resource about other resources. For example, the statement “John is a student at Arkansas State University” can be further reified to “Ryan has stated that John is a student at Arkansas State University”. The reification of statements is important to define the level of usefulness for each triple. In a reified RDF triple store, or RDF database, each resource will, most likely, be able to have at least three additional statements made about the original statement. One such statement that asserts that the subject is another resource, another asserting that the predicate is a resource and another that shows the object as another resource or a string literal. Some RDF implementations acknowledge that grouping statements according to different criteria (situations, contexts, or scopes) is sometimes useful (Klyne, Contexts for RDF Information Modelling, 2000) (Klyne, Circumstance, provenance and partial knowledge, 2002). A statement can be associated with a context identified by a URI in order to show an “is true in” type of relationship. In a dynamic way, statements can be grouped by their source, for example, grouping by the URI of an external RDF document, so that when changes are made to the source any statements that correspond can be changed in the model. Scopes of implementation are not limited to fully reified statements as some allow a single scope identifier that is not assigned a URI (Beckett, 2004).
To query RDF data, the current standard is through a structured query language called SPARQL.

“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.” (Prud'hommeaux & Seaborne, 2008)

As of January 2008, SPARQL has become the recommended specification for RDF queries. A SPARQL query is structured, in order, a prefix declaration for abbreviating URIs, a dataset definition stating what RDF documents are being queried against, a result clause to identify what results are desired, a query pattern specifying what to query for in the dataset, and any query modifiers for sorting or arranging results. An example follows in Figure 2.

```sparql
PREFIX space: <http://purl.org/net/schemas/space/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT * WHERE {
  ?subject rdfs:label ?label;
  rdf:type space:Discipline.
}
```

Figure 2: Example of a SELECT query, retrieving all variables.

Other methods of querying RDF data exist, such as SPARQL’s precursor called RDF Data Query Language (RDQL) (Seaborne, 2004), the RDF Query Language (RQL) (Karvounarakis & Christophides, 2008), or XML User Interface Language (XUL) (Mozilla, 2009), for example.

RDFa was first proposed by Mark Birbeck authored in a W3C note entitled XHTML and RDF, (Birbeck, 2004) which was then presented at the W3C’s 2004 Technical Plenary (Birbeck, 2004). The W3C note was included in the sixth public working draft of the XHTML 2.0 specification (W3C, XHTML Metainformation Attributes Module, 2004). It is generally assumed that RDFa was originally intended only for XHTML 2. The purpose of RDFa was to provide a way to add a metadata to any XML language. One of the first documents bearing the RDFa name is subtitled “A Collection of Attributes for Layering RDF on XML Languages” (W3C, A Collection of Attributes for Layering RDF on XML Languages, 2004).

The XHTML 2 Working Group produced a module to support RDF annotation within the XHTML 1 family (W3C, Modules to Support RDF Annotation of Elements, 2007) which included an extended version of XHTML 1.1 called XHTML+RDFa 1.0. Limited use of the XHTML+RDFa 1.0 DTD appeared on the web although the DTD was described as not representing an intended direction in terms of a formal markup language. The first public working draft of a document entitled “RDFa in XHTML: Syntax and Processing” (W3C, A Collection of Attributes and Processing Rules for Extending XHTML to Support RDF, 2007)
was introduced in October of 2007. This expanded upon the April draft; it contained rules for creating an RDFa parser, as well as guidelines for practical use. In October 2008, RDFa reached Recommendation status (W3C, A Collection of Attributes and Processing Rules for Extending XHTML to Support RDF, 2008) and an additional RDFa Primer document was updated in June 2008 (W3C, Bridging the Human and Data Webs, 2008).

Ontology Languages (OWL, FOAF, etc.)

As RDFS and OWL demonstrate, additional ontology languages can be built upon RDF. The Web Ontology Language (OWL) is a family of knowledge representation languages for authoring ontologies endorsed by the W3C (Smith, Welty, & McGuinness, 2004). These languages are based upon two compatible, for the most part, semantics. OWL DL and OWL Lite semantics, based upon Description Logics (Horrocks & Patel-Schneider), have well-understood computational properties. OWL Full uses a novel semantic model intended to provide compatibility with RDF Schema. OWL ontologies are most commonly serialized using RDF/XML syntax. OWL is considered one of the fundamental technologies underpinning the Semantic Web (OWG, 2009). The OWL Working Group started to extend OWL with several new features that were proposed in the OWL 1.1 member submission (SRI International; TopQuadrant; University of Manchester; webMethods, Inc., 2006) called OWL2. OWL2 has already found its way into semantic editors such as Protégé (Protégé, 2009) and Semantic Reasoners such as Pellet (Clark & Parsia, LLC., 2009) and FaCT++ (FaCT++, 2009).

Friend of a Friend (FOAF) is a machine-readable ontology describing people, their activities, and their relations to objects or other people. FOAF allows groups of people to describe social networks without the need for a centralized database. FOAF is a descriptive vocabulary expressed using RDF Resource Description Framework and OWL Web Ontology Language used to define relationships between people. Each profile has a unique identifier which is used when defining these relationships. These identifiers can be an email address, instant messenger identification or screen name, or a URI that locates a blog or homepage. In 2000, Libby Miller and Dan Brickley began the FOAF project (Project, 2009). The project outlines the methods and definitions of FOAF profiles. In 2007, Tim Berners-Lee authored an essay (Berners-Lee T., 2007) redefining the Semantic Web concept into something he calls the Giant Global Graph (GGG). The GGG is, as Berners-Lee states in his essay, “where relationships transcend networks or documents”. He considers the GGG to be on the same level with the Internet and World Wide Web, stating that "I express my network in a FOAF file, and that is a start of the revolution." The following example in Figure 3 was generated by FOAF-a-Matic (Dodds, 2009), an automated FOAF file generator:
Figure 3: An automatically generated FOAF file from the FOAF-a-Matic Website (Dodds, 2009).

**Existing RDF/RDFa Applications**

There are a large number of applications that already exist that utilize RDF/RDFa technologies. Some are well known while others are not so common, or even hidden from plain view. Social networking applications such as Wordpress blogs (Codex, 2009) and Facebook (Idehen, 2007) utilize RDF/RDFa data sharing, Really Simple Syndication (RSS) feeds, and use of FOAF profiles for their authors. Other applications exist such as Chandler, a personal information management tool for the desktop, network and security configuration ontologies such as SKOS, and Dublin Core for library and museum data.

Chandler is a personal information management software suite (Chandler, Chandler 1.0, 2008) designed for personal and small-group task management and calendaring. Chandler is licensed under the Apache 2.0 License (Chandler, Chandler is now licensed under the Apache 2.0 License, 2006) and is publically free software previously released under the GNU General Public License. Inspired by Lotus Agenda, Chandler features a "free-form" approach to information management. Lead developer of Lotus Agenda, Mitch Kapor (Kapor, 2009), was involved in the initial releases and project development of the Chandler suite. The Chandler suite is comprised of a cross-platform desktop application, the Chandler Hub Sharing Service, the Chandler Server application, the Chandler Quick Entry for iPhone and Chandler iGoogle widgets.
Really Simple Syndication (RSS) is a standardized collection of document formats used in the publication of frequently updated website portions such as news headlines and articles, blog entries, or audio and video files (Libby, 1999). An RSS document includes full or summarized articles and metadata about updated materials such as publish date and author. Web feeds, as these RSS documents are referred to, benefit content providers and authors by allowing content syndication automatically. Readers benefit by being able to stay updated, by subscription to a web feed, to the most current activities of their favorite websites and also have the ability to aggregate many web feeds into one application called an “RSS reader” or “aggregator”. These web feed aggregators can be browser-based or available on the desktop. Some others provide content to mobile devices as well. RSS documents are published using a standardized XML format that allows many applications to read the data. The end user will subscribe to the web feed by supplying a URI to their reader application, or, by clicking a link on the original hypertext document. The reader application will search all of the end user’s supplied URIs for changes to the referred works and display those changes to the end user.

The Dublin Core metadata element set builds upon RDF for cross-domain information and defines a naming convention in an attempt to standardize a method for easy retrieval of information. Dublin Core is in wide use to describe digital files such as audio, video, images, text, and mixed media in web pages. The Dublin Core implementation is defined by the ISO standard 15836 (ISO, 2009) and the NISO standard Z39.85-2007 (NSIO, 2007). Dublin Core utilizes the basis of XML and RDF in its implementation and is typically used for library and museum data. The Dublin Core has two levels of implementation; Simple and Qualified. The Simple Dublin Core is comprised of fifteen elements that are used to describe the properties of a web document. The fifteen elements are listed within Figure 4.

![Figure 4: The fifteen elements comprising the Simple Dublin Core.](image)

The Qualified Dublin Core adds additional terms to the Simple Dublin Core by the addition of element refinements to make the meaning of an element more specific. That is, elements must have a more restricted scope. The details of implementation of the Dublin Core, both qualified and simple are beyond the scope of this research. Those who seek more information about the Dublin Core can find information in The Dublin Core Metadata Registry website (DCMI, 2008).

The Simple Knowledge Organization System (SKOS) is a collection of formal languages intended for the representation of controlled vocabularies and to make them available to semantic web applications. In a nutshell, SKOS is intended to simplify more complex ontologies such as OWL. The construction of useful web ontologies is expensive in terms of cost, effort, and expertise. The extensibility of RDF makes possible further incorporation or extension of SKOS vocabularies into more complex vocabularies, including OWL ontologies.
Criticisms and Barriers

In 2001, RDF was criticized on the grounds of verbosity, reading and computational disadvantages, and the allowance of ambiguity (Shirky, 2003). In the Shirky research, Shirky states that the XML syntax for RDF is too verbose and suggests that a more grammatically concise representation can be found within Notation 3. Shirky comments further stating that the notation of triples (subject, predicate, and object) introduces reading and computational disadvantages. The triple is a virtually universal linguistic construct and has been handled by artificial languages such as PROLOG for decades. Shirky also states that “RDF’s ability to add an infinite amount of triples allows an author to negate ambiguity - this really is a matter of the authorship of the knowledge-representation, and not the format of the representation”. Other criticisms of RDF exist, although, many of the published criticisms are outdated, such as those presented in Shirky’s research. Today, the barriers to implementing RDF in documents exist in a slightly different scope.

The Open World Assumption (OWA) and Closed World Assumption (CWA) theories of formal logic can be applied to how RDF data is handled. The OWA states that the truth-value of a statement is independent of whether or not it is known by a single observation to be true. The CWA holds that any statement that is not known to be true is, simply, false by nature. To apply these definitions to RDF triples, one can make completely different assumptions about the predicate and object portions of a triple. The following figure (Figure 5) outlines such an example.

<table>
<thead>
<tr>
<th>Open World Assumption</th>
<th>Closed World Assumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statement: Joe is employed by Arkansas State University.</td>
<td></td>
</tr>
<tr>
<td>Question: Is John employed by Arkansas State University?</td>
<td></td>
</tr>
<tr>
<td>OWA Answer: Unknown</td>
<td>CWA Answer: No</td>
</tr>
</tbody>
</table>

Figure 5: A comparison of OWA and CWA of a possible uniformed RDF triple.

In the example above, the OWA does not assume, nor does it represent, any knowledge about the question being asked based upon the statement. The OWA is used in knowledge representation to codify the informal notion that, in general, no single observer has complete knowledge, and therefore, cannot make an informed statement. The OWA limits the kinds of inference and deductions an agent can make to those that follow from statements that are known to the agent to be true. In contrast, the CWA allows an observer to infer, from its lack of knowledge of a statement being true, anything that follows from that statement being false. This is explicitly true in methods of information retrieval such as Structured Query Language (SQL) statements. We must assume, syntactically, that simple data queries that retrieve no (zero) results are “unknown”, and not a definitive “No” answer. This does not work in the case of an inventory quantity request as is shown in Figure 6.

<table>
<thead>
<tr>
<th>Open World Assumption</th>
<th>Closed World Assumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question: How many of part number OU-812 are left in inventory?</td>
<td></td>
</tr>
<tr>
<td>Possible SQL Query: SELECT COUNT(*) FROM inventory WHERE part_num='OU-812'</td>
<td></td>
</tr>
<tr>
<td>OWA Answer: Unknown</td>
<td>CWA Answer: 0</td>
</tr>
</tbody>
</table>

Figure 6: A comparison of OWA and CWA of an inventory item quantity request.
In the above example the OWA fails to answer the question where the CWA gives an appropriate response. Neither RDF nor layering ontologies supply a method to interpret computational results in any intelligent form.

Concluding Remarks

When learning RDFa, the first logical step is to first learn the basis, or, RDF/XML. In order to take advantage of semantic web properties one does not have to use RDF/XML. Other methods, for example Notation 3, are available and useable. One of the major problems in RDF/XML is in its complexity. If a developer were to find difficulty learning RDF/XML then, by nature, all other ontologies and forms of RDF will become difficult as well. This notion may cast unwarranted disgust with adopting RDFa and shy developers away. Most examples of RDFa are academic and not real world implementations. This does provide the theory behind RDFa and what it can possibly be used for; however, to apply the theories in real world circumstances can become difficult in that interpretation of academic examples and tutorials will, normally, not directly apply to real world problems one may encounter while developing business web applications. XML serialization of RDF uses a striped syntax (Brickley, 2002), which users find confusing as the interpretation of elements changes depending on which stripe they are on in the document. One could argue that an RDF/XML document is more syntactic, or strict, than an XML document containing attribute and value pairs. The RDF is supposed to be able to handle different ontologies, vocabularies, and different versions of each than is possible with XML. Currently, there is no standard approach to dealing with different versions of RDF vocabularies. One major pitfall to RDF data sharing is simply found within human nature. The question of how to handle those chunks of data available through the RDF that simply are not true remains to be answered. If someone in the world published data that made false statements, how could RDF/XML provide a method of validation? If other RDF data parsing is referenced in this case, the false data would simply spread, thus completely negating the intended purpose of sharing RDF or Linked Data.

Finally, in order to truly fulfill the intentions of bringing the World Wide Web to a state where it is semantic in nature, raw data must be shared. All data including governmental, scientific, academic, and enterprise data must be made available, generally, to the public. Most businesses and owners of data stores consider their data as something that is owned, not shared. As long as data is kept secret and treated as a unique resource, the Web can never become truly semantic in nature. Technologies such as RDF/RDFa that were designed to accomplish the implementation of a Semantic Web become moot.

References


