Semantic Web Framework for E-Commerce Based on OWL

[1]O. N. Oyelade, [2]S.B. Junaidu, [3]A.A. Obiniyi,

[1,2,3]Department of Mathematics, Ahmadu Bello University

Zaria, Kaduna, Nigeria.

Abstract

The initial Web was called static web because only it allowed users to follow clickable links in a bid to view information. That primitive Web platform provided means for propagating electronic commerce (E-commerce). However, the Semantic Web Initiative and its supporting technologies further improved the Web, which have enhanced E-commerce. Some of the Semantic Web technologies that have been employed to enhance the primitive Web are Semantic Web technologies such as languages like XML, RDF/RDFS, OWL and also, query and rule languages like SPARQL, SQWRL and SWRL. In this paper, the goal is to demonstrate how data modeling with OWL and as well queried with SQWRL will enhance both more hits for products search of products in B2C scenario. Also, SWRL will be used to add entailments to the OWL ontology so that the weakness of SPARQL may be clearly shown compared to SQWRL, and by so doing, improve the efficiency of B2C. We seek to a mobile phone based Semantic Web application to implement the goal.

Keywords: *E-commerce, OWL, RDF, RDFS, SQWRL, and SPARQL.*

1.0 Introduction

Information gathering, retrieval and usage are a very crucial aspect in both social and business worlds for effective decision making. However, prior to the time of the Web (WWW) it was difficult for organizations to share information with other organizations. Tim Berners-Lee (1994), proposed the Web, a decentralized solution that helps organizations share information with the world. This Web grew within a very short time and it changed everything about our lives. Web technologies continue to improve, starting from the initial static web to the dynamic and interactive web. The Web has further developed into one in which information can be given well-defined meaning, better enabling computers and people to work in cooperation (Rubbani, 2007).

One major aspect of our lives that the Web has greatly enhanced is commerce, known as Ecommerce in the computer world. E-commerce has three major categories through which goods and services can be exchanged. These are Business to Business (B2B), Business to Consumer (B2C), and Consumer to Consumer (C2C). This paper focuses on implementing a Semantic Web application for B2C. The Semantic Web technologies are well tailored in such a way that machines can read the semantics of the information they convey. Some of these technologies are eXtensible Markup Language (XML), Resource Description Framework (RDF)/RDF Schema (RDFS), Web Ontology Language (OWL) and XML

Topic Maps (XTM) and other supporting technologies.

Ontology is the specification of a conceptualization (Gruber, 1993) or better put, it is the process of specifying the entities or classes, properties or relations that are peculiar to a domain in a formal way. Ontology language is a formal language used to encode the ontology (Maniraj *et al*, 2010) and some of these ontology languages include XML, RDF/RDFS, XTM, and OWL. The choice of these ontology languages depends on the domain of application. In this paper, we consider OWL as the desired ontology language. More so, models implemented by ontology languages are accessible through some query languages. For instance, XPath is used to query XML, SPARQL Protocol and RDF Query Language (SPARQL) and RDF Query Language (RQL) is used on RDF, while SPARQL can as well be used on OWL as well, however, it is more efficient to use Semantic Query Web-enhanced Rule Language (SQWRL) in querying OWL because it was actually designed to understand the semantics of OWL constructs (O'Connor *et al*, 2009) and this is one of the contribution of this work. Finally, some of these ontology languages have support for rules. This is because ontology may also contain some extra information about properties, and the restriction placed on these properties, and entailments generated by rules language (Antoniou *et al*, 2004). Semantic Web Rule Language (SWRL) is a supporting rule used with OWL just as other ontology languages also have their supportive rule languages.

2.0 Related Works

VijaLakshmi *et al* (2011) presented e-commerce data modeling using RDF and used SPARQL as a query language to query the ontology. The work developed ontology in the mobile phones domain so that consumers can access information from the ontology through a web application by entering the search words, then a search agent would query the underlying ontology. The producer on the other hand makes use of ontology agent to deploy or populate the underlying ontology that consumer will later access. An obvious limitation of this work is the use of RDF, which does not have more expressive constructs, as an ontology language and SAPRQL as a query language. In our paper, we seek to use OWL/SQWRL to both enhance data modeling and query hits.

Shamoug *et al* (2012) showed how they supported the reporting and decision making process in Humanitarian Response (HR) through Semantic Web technology and improve its efficacy. They centered the decision making in HR on asking questions that relates with WHO does WHAT, WHERE, WHEN, HOW and WHY (5WH) during Humanitarian Crises (HC). And the proposal was illustrated through a case study and its implementation with OWL/SWRL enabled technologies. The 5WH where used to generate the concepts that the ontology was built upon. SWRL is used to generate rules that an agent will reason over in its bid to choose the appropriate humanitarian organization that will be deployed in a conflict or crises location. This work utilizes OWL/SWRL in implementing a decision system, but however left out the use of

SQWRL, a fitting query language that can be used alongside OWL/SWRL. The work relates with our own work only in the use of OWL, SWRL.

As the work of VijaLakshmi *et al* use RDF, rather than OWL, as their modeling language, their systems lack the needed expressive power since, for example, their systems will yield ontologies that cannot generate entailments. Although Shamoug *et al* uses SWRL and OWL to implement their ontology and rule, they did not use SQWRL for querying the ontology. SQWRL would have afforded them the benefits of retrieving more useful results.

3.0 Materials and Methods

The expressivity of RDF and RDFS is very limited. RDF is limited to binary ground predicates and RDFS is limited to subclass hierarchy and property hierarchy with a domain and range definition of these properties (Antoniou *et al*, 2004). However OWL has more expressivity than RDF/RDFS. For example, OWL allows property range to apply to some classes while RDF/RDFS only allows property range to be specified for all classes. Secondly, there is no means to use RDF/RDFS to specify disjointness of classes, but this can be done with OWL. Thirdly building new classes from a Boolean operation on existing classes is possible with OWL but not with RDF/RDFS. Placing restrictions on how many distinct values a property may have is not expressible with RDF/RDFS except with OWL. And lastly, some particular features of properties, like a property or relation being transitive, symmetric, functional or inverse are not implementable with RDF/RDFS, hence OWL becomes the best choice to express this.

Consider the ontology captured in Figure 1. To specify in this ontology that a class called **Retailer** must at least be in a category, we implement it as captured in Figure 3. Figure 2 points out that class ***Retailer*** may or not be in any category. Figure 4 shows the changes in the ontology hierarchy when we eventually add this restriction. Figure 1 shows a visual representation of the ontology. Two classes (**Retailer** and **Category**) and their instances are visualized by the Figure. While Figure 2 is an extraction of the part of the ontology which modeled that **Retailer** is a class in the ontology.

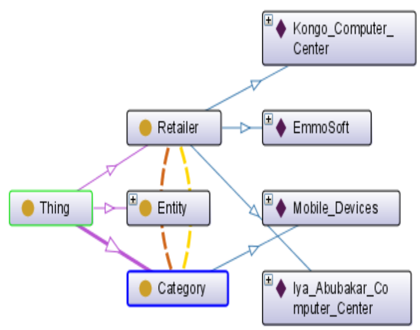

Figure 1: Ontology without restriction

```
<rdf:Description rdf:about="#Retailer">
  <rdf:type
rdf:resource=http://www.w3.org/2002/07/owl#Class
/>
 </rdf:Description>
```

Figure 2: OWL tags without restriction on class Retailer

ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784

Figures 3 and 4 shows the same ontology captured in Figures 1 and 2, but the major difference lies in some restrictions and object property features added to the original ontology using OWL constructs. These improvements will make it very difficult for SPARQL to retrieve information from the ontology whereas SQWRL will retrieve information from the same ontology using the same semantics of query. With respect to OWL syntax, one major difference between the Figures 2 and 3 is that, Figure 2 only states that **Retailer** is a class moreover, Figure 3 goes further to state the relation or predicate (**inCategory**) that the class **Retailer** has with another class called **Category**. The construct **owl:someValuesFrom** implies that at least one instance of class **Retailer** must have a **inCategory** relation with instance of class Category. Some of this level of expressivity in OWL are the reasons why SPARQL will retrieve a poor search hits compared to SQWRL, which understands OWL semantics very well. As a result of this expression carried out in the modified ontology that places restriction on the ***Retailer*** class, it enables a reasoner to be able to make intuitive derivation from the underlying ontology.

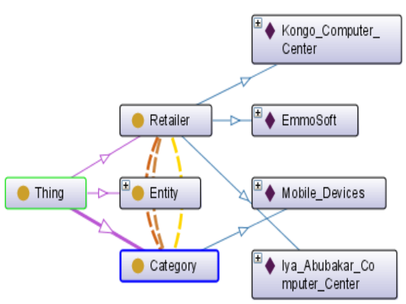

Figure 3: OWL tags with restriction placed on Retailer class.

```
<owl:Class rdf:ID="Retailer">
  <rdfs:subClassOf
rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:InverseFunctionalProperty
rdf:ID="inCategory"/>
      </owl:onProperty>
      <owl:someValuesFrom rdf:resource="#Category"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Figure 4: Ontology with restriction placed on Retailer class

We developed two ontologies for the purpose of the research reported in this paper. The first ontology which is the core ontology, serves as a registry for retailers to make their ontologies known to intending searchers. The core ontology gives information about retailers and, most importantly, it provides a link to the ontology of the retailers so that users' requests may be searched against the retailers' ontologies. The second ontology is the retailer ontology, where information such as the price, model, and quantity of an available product will be made available by every individual retailer for their prospective consumers. Figure 5 shows the architecture of the system.

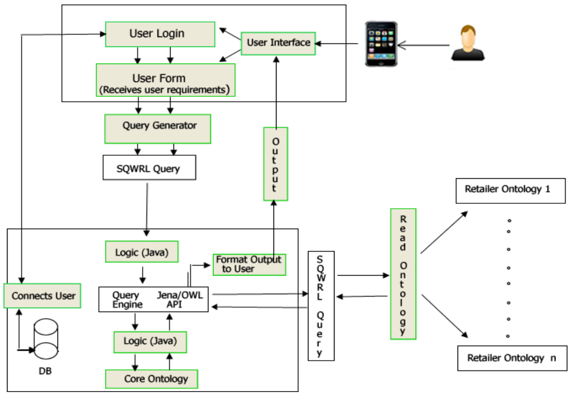

Figure 5: The Architecture of the Semantic Web Framework

3.1 Accessing Information from the Ontology

Data retrieval from ontology is of great interest and the choice of a language with which to do that must be taken with care, since the ontology can be implemented to have Closed World Assumption (CWA) or Open World Assumption (OWA). The CWA says information not known to be true is false and OWA says information not known to be true is unknown or cannot be false (Taofifenua, 2012). In CWA, only facts that are directly stated in the ontology are retrievable by the query language, while the OWA, what may be intuitively gainer as facts will be retrieved. Generally, there are so many query and rule languages that may be used with OWL. But this research focuses on SPARQL and SQWRL as query languages and SWRL as a rule language.

SPARQL is currently the *de factor* standard for querying RDF and since OWL can be serialized in RDF, SPARQL is used in querying it (Necula, 2012). However, SPARQL has no native understanding of OWL. It operates only on its RDF serialization and has no knowledge of the language constructs that those serializations represent. As a result, it cannot directly query entailments made by those constructs (O' Connor *et al*, 2009). When it comes to functions, there is no aggregated function implemented by SPARQL yet (Necula, 2012). There is thus a need for an expressive OWL query language that supports comprehensive querying of OWL (O' Connor *et al*, 2009). This paper proposes the use of SQWRL which is built on SWRL because our ontology is being deployed with OWL in its more powerful expressivity and also, this research demands the use of aggregated function. SQWRL takes on the antecedent part of SWRL and joins itself as the coincident part to form a query that will be executed on a rule engine. And Jess is the current rule engine. A current popular rule engine is Jess [http://www.jessrules.com/jess/index.shtml] which we make use of in this paper.

In particular, SWRL supports the development of querying tools that allows dynamic knowledge –driven access to relational data (O' Connor *et al*, 2009). It also complements OWL for the definition of ontology and they are written as antecedent and coincident. The antecedent is referred to as the rule body while the coincident is referred to as the rule head (Taofifenua, 2012). A typical example of the use of SWRL is;

Retailer(?ret) ^ sellsMobile(?ret, ?mob) -> ProductAvailable(?mob, ?ret)

The antecedent or left hand side of the rule states that if there is a retailer who sells a mobile device or phone, then we conclude in the coincident or the right hand side of the rule implies that such mobile phone product is available with this specific retailer.

Figures 2 and 3 where used to contrast the different expressivity of OWL. To demonstrate the efficiency of using SQWRL rather than SPARQL in querying OWL, an application was developed in the process of carrying out this research and this application was used in carrying out this comparison. We first execute a SPARQL query on the ontology in Figure 1 and at the same time, we implement a SQWRL that is equivalent to the SPARQL query on the same ontology in Figure 1. We observe that the two queries resulted in the same number of searches as shown by Figure 6.

However, when we run this same sets of queries (that is, the queries for SPARQL and SQWRL) on the second ontology captured by Figure 4, we observe that SQWRL yielded three (3) results while SPARQL resulted in zero (0) outputs as shown by Figure 7.

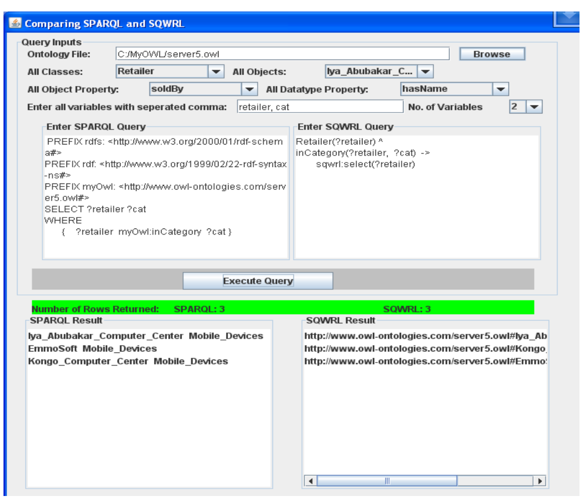

Figure 6: Querying ontology without restriction

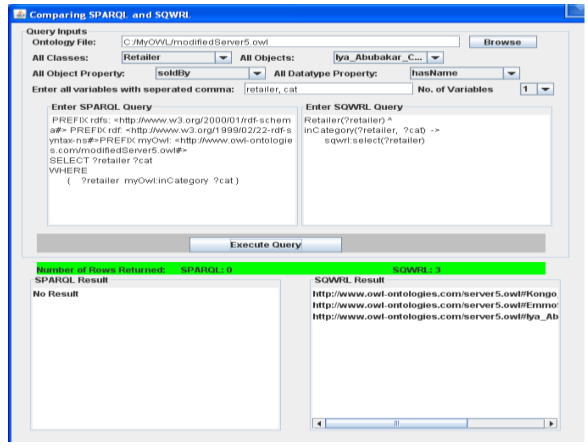

Figure 7: Querying ontology with restriction

The queries executed for these two snapshots are as shown below.

SPARQL	SQWRL
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> PREFIX rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns# PREFIX myOwl: <http://www.owl-ontologies.com/server5.owl#> SELECT ?retailer ?cat WHERE { ?retailer myOwl:inCategory ?cat }	Retailer(?retailer) ^ inCategory(?retailer, ?cat) -> sqwrl:select(?retailer)

A close look at the SQWRL query will show that the left hand side of the query is SWRL syntax and this is why it was earlier stated that SQWRL depends or is built on SWRL.

4.0 Implementation

For the implementation of this semantic application, we chose to deploy the application on mobile devices or phones, and specifically Android driven mobile device which is an open source software (Steele *et al*, 2011). Mobile devices are portable and are easily carried along by their users. And by so doing, users of the application will enjoy the ease of access to the application and also leverage on the ubiquity of computing. For this work, ease of access of the application and possible personalization features that may be added to subsequent improvement of this application was considerations that influenced the choice of Android driven devices.

Here the part of the framework deployed on the mobile device will function as client connecting to the server where these ontologies mentioned in the previous section are going to be queried. Figure 8 demonstrates how the user can enter the search request on the mobile device while Figure 9 shows the result of the search carried out and alongside the functionality for the user to either dial or SMS the retailer.

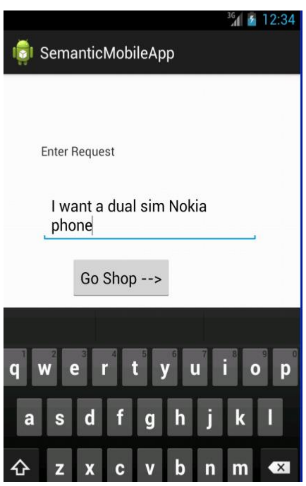

Figure 8: User preferences entry form

5.0 Results and Discussion

The application displayed in in Figures 8 and 9 was designed in such a way that when the user requests are entered into the search form, it generates, on the server side, a SQWRL and a corresponding SPARQL query which are channeled to a file. A sample of these automatically generated SQWRL and SPARQL queries were used to query the retailer ontology and the output of the result is as shown by Figure 10.

Figure 9: Search result with dialing and SMS menu

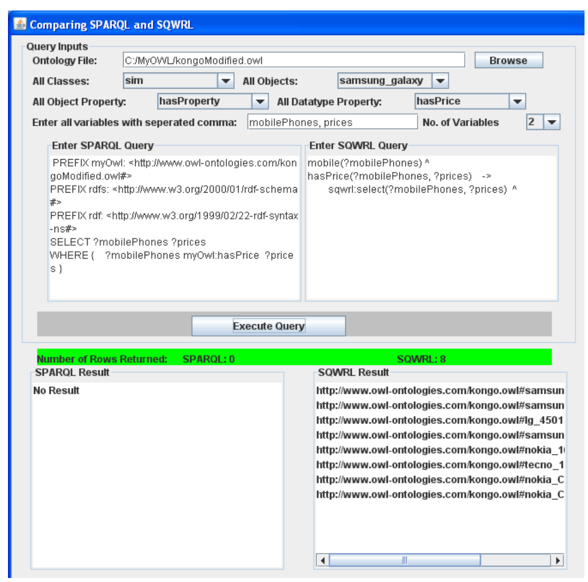

Figure 10: Comparing SPARQL and SQWRL.

This clearly reveals that we can realize more efficiency in terms of number of hits or search results in a B2C when the ontology is implemented with OWL and the query language used is SQWRL. The retailer ontology used in Figure 10 uses some more expressive OWL constructs. This was aimed at enriching the ontology so that inferences can be employed when reading through it, and so that SWRL can be used to generate entailments which will be added to the ontology, thereby creating more instances that are not out rightly stated in the ontology. This is why SPARQL, which is limited in understanding some of these OWL constructs, could not return result in Figure 10. Moreover, SQWRL returned eight (8) rows from the same ontology that SPARQL could not return anything.

6.0 Conclusion and Future Work

RDF/RDFS are good ontology languages, though it is quite primitive compared to more expressive constructs that OWL provides. Query languages for different ontology languages abound, but most of these query languages were originally tailored to excellently understand the semantics of a particular ontology language. In this paper particularly, we have focused on RDF and OWL as ontology languages and their respective befitting query langues SPARQL and SQWRL. A major advantage of OWL over RDF/RDFS is that it enables the coding deeper facts which in themselves can generate other facts that are not readily modeled in the ontology. when such facts are implemented in an ontology, it becomes difficult for SPARQL to be used to query the OWL ontology and thereby yield required result. Hence, this paper showed that product search on B2C can be improved upon when the ontology is modeled using OWL, and SWRL/SQWRL is used to add more rules to and query the ontology. This was shown in the Semantic Web application that was designed and deployed on a mobile device. Furthermore, an application was also developed to illustrate a scenario whereby same OWL ontology was been queried by SPARQL and SQWRL with the later yield result while the former langue yielded none. Making the process brokering and negotiating for the product, and paying the bills of the purchased product is the next point that this work should be focused on. Hence, this is left as a future work for subsequent consideration on this research.

7.0 Reference

[1] Shamoug, A., Radmila, J., and Shamimabi P., (2012). *Ontological Reasoning as a Tool for Humanitarian Decision Making*, Proceedings of the 9th International ISCRAM Conference – Vancouver, Canada, pp. 1.

[2] Andrea Liechti – Search Results Ranking for a Reputation Analysis Prototype. Pg. 8, 2012.

[3] VijayaLakshmi, B., GauthamiLatha, A., Srinivas, Y. and Rajesh, K. (2011). *Perspective of Semantic Web in E-Commerce*, International Journal of Computer Applications (0975 – 8887) Volume 25– No.10, pp. 53-54.

[4] Antoniou, G., and Frank V. H. (2004). *Semantic Web Primer*, MIT Press, Cambridge, Massachusetts, London, England, pp. 10-13.

[5] Rubbani, H. H., (2007). *Semantic Web Solutions*. IT University of Copenhagen, pp. 9.

[6] Steele, J. and Nelson T.(2011). *The Android Developer's Cookbok: Building Applications with the Android SDK*, pp.

[7] O'Connor, M., and Amar, D. (2009). *SQWRL: a Query Language for OWL*, Stanford Center for Biomedical Informatics Research, Stanford, pp 1.

[8] Taofifenua, O., (2012). *Ontology Centric Design Process, sharing a conceptualization.* A thesis work dated November 2012.

[9] Ribeiro-Neto, B., and Baeza-Yates, R., (2010). *Modern Information Retrieval: The Concepts and Technology behind Search*; 2nd revised edition, Addison-Wesley Educational Publishers Inc., pp. 9.

[10] Gruber, T. (1993). *A translation approach to portable ontologies*. Knowledge Acquisition, 5(2):199-220.

[11] S. B. Necula- Implementing the Main Functionalities Required by Semantic

[12] Search in Decision-Support Systems. INT J COMPUT COMMUN, ISSN 1841-98367(5):907-915, December, 2012. Pg 908.

[13] Maniraj, V., and Sivakumar, R., (2010). *Ontology Languages – A Review,* International Journal of Computer Theory and Engineering, Vol.2, No.6, pp. 887.

[14] Tim Berners-Lee (1994). *Weaving the Web*, The MIT Press Bookstore, Cambridge, pp. 9.

[15] Jess. Retrieved October *22, 2013 from* http://www.jessrules.com/jess/index.shtml

Authors Biography

Oyelade Olaide Nathaniel is a masters student of computer science in the Department of Mathematics Ahmadu Bello University, Zaria-Nigeria. His current research interests includes Semantic Web technologies, specifically ontology languages and their query and rule languages. Also, he has research interest in mobile computing. Oyelade earned his B.Sc. degree in computer science from University of Jos, Nigeria in 2010

Dr. A.A. Obiniyi received his Ph.D degree in Computer Science from Ahmadu Bello University (ABU), Zaria in Kaduna State of Nigeria in 2009. He is a member of Nigeria Computer Society, Internet Society and a Chartered member of Computer Professionals (Registration Council of Nigeria). He lectures in the Department of Mathematics of ABU and has many publications to his credit.

.

SB. Junaidu is professor of Computer Science at Mathematics Department of Ahmadu Bello University, Zaria. He earned his MSc. degree in Computer Science at Queen Marry & Westfield College, University of London in 1992 and PhD Computer Science at St. Andrews University Scotland, UK in 1998. He is currently the Director of Iya Abubakar Computer Center, Ahmadu Bello University, Zaria. His areas of research include Parallel Computing, Web Applications Development, Programming Languages, Computer Science Education, E-Learning.