

OWL and SWRL: Scope, Propensity and Future of Web Based Distributed Multi Agent Application

A.A. Obiniyi, O. N. Oyelade E. F. Aminu

Abstract - The web is overloaded with data and most of the search available search engines accomplish their retrieval task syntactically. The Semantic Web however promotes avenue for the separately stored web data to be intelligently reasoned over during data retrieval. Intelligent systems are coupled around intelligent data retrieval, reasoning and inference making over a knowledge base. In this paper, we explore and present the scope and usability of OWL – Semantic web data repository – and SWRL, considering their relevance to the field of artificial intelligence. We also points out the trends in the development of ontology languages with emphases on Ontology Web Language (OWL). We propose that distributed mobile agents would largely influence the design of high performing web applications solving difficult problems. Furthermore, we show that efficiency of such multi agent systems may be realized by agglomerating nodes in nearness within the system. Finally, we highlight some ideas that reveal what the future of complex web applications stand to gain by using distributed agents which collectively perform a task.

Keywords- Ontology, OWL, SQWRL, SPARQL, Rule Engine, SWRL and Artificial Intelligence.

1 Background of Study

The term ontology was derived from the Greek word with **onto** meaning *being* and *logy* or *logos* meaning science (Lawson, 2004). Philosophically, ontology means the study of the existence of a thing. However, in Semantic Web, ontology is a specification about conceptualization (Gruber, 1993) which provides a platform for knowledge sharing and reuse mostly in the study of artificial intelligence. Much effort that have been geared towards improving semantic web technologies- that is, the technologies that enhances machine readable content or content that machines can

derive meaning from – have yielded some standards such as Web Ontology Language (OWL) and Web Services Languages such as Web Service Description Language (WSDL), Universal Description, Discovery Integration (UDDI) and Simple Object Access Protocol (SOAP). One of the goals of Semantic Web initiative is to provide an infrastructure for intelligent agents and web services, and also to provide the aggregation of distributed data through ontologies and reasoning. This infrastructure is based on formal domain models (ontologies) that are linked to each other on the Web. And these linked ontologies provide application with shared knowledge base and understanding (Knublauch, 2005). Hence, it may be said that ontologies are developed for the purpose of collecting terms or concepts in a particular domain and then relating those terms together in a structural and organized way so that such information may be used to store related information and as well to enable information

- Dr. A. A. Obiniyi lectures in the Department of Mathematics of Ahmadu Bello University, Zaria, Kaduna, Nigeria. E-mail: aaobiniyi@gmail.com
- O.N. Oyelade is holds a masters in computer science from Ahamadu Bello University, Zaria, Nigeria. E-mail: olaide_oyelade@yahoo.com
- E. F. Aminu lectures at Federal University of Technology, Minna, Niger, Nigeria and holds masters in computer science from Ahmadu Bello University, Zaria. E-mail: enesifa@futminna.edu.ng

sharing that is aided by machines – intelligent agents.

Ontology development is realized using different ontology languages. Figure 1 captures the semantic web cake which gives associative representation of some semantic web languages. Some ontology languages briefly discussed in this paper includes Research Projects Agency, **DARPA Markup Language (DAML)** and **Ontology Inference Layer (OIL)** - DAML+OIL - and, RDF, OWL Lite, OWL Description Language (DL) and OWL Full. However, in this paper, we concentrate on OWL – an ontology language. We highlight their degree of usefulness of each of this ontology languages and what relation exist amongst them.

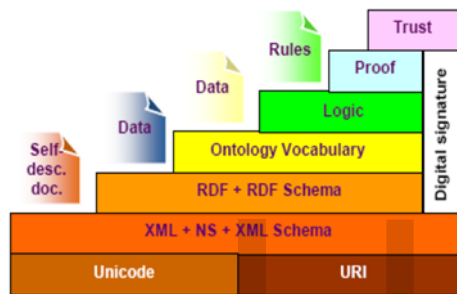


Figure 1: Semantic Web Technology Stack (Medić and Golubović, 2010)

Ontology query languages are cogent in the discussion of ontology design languages. Sparql Protocol and RDF Query Language (SPARQL) is a query language that is meant for querying ontology implemented using Resource Description Framework and Resource Description Framework Schema (RDF/RDFS), though these query languages have been found to be useful in querying ontology implemented in OWL, but its set back lies in its lack of understanding of the semantics of OWL some constructs. SPARQL does not have a complete understanding of OWL's semantics (Martin et al, 2009). A more primitive query language for OWL is SQWRL (Semantic Query Web Rule Language). SQWRL provides few, though powerful features and operators that enable one to query information from any OWL ontology.

Some of the available popular search engines around are Google. A peculiar challenge that necessitated the improvement on information retrieval, courtesy of the semantic Web initiative,

was the due to lack of precision of search result returned by keyword based search engines. The sheer scale of web data and its decentralized nature continues to pose difficulty in actually harnessing such ever amassing online data for searches. This distributed and continuously building up data tends the web towards what is often referred to as information overload (Nováčak, 2010). The workability of keyword search engines is to make their search base on strings or syntactic matching of user search input as against the underlying database. Expectedly, this may result in large search result containing both relevant and

irrelevant data source to the user. But Semantic Web initiative proposes a semantic approach in implement web search through the use of semantic web based search engines. We strongly perceive that query executed using semantic web approaches will result in (possibly) few results (though depending on the size of the underlying ontology) and the result will mostly be semantically related with what the user demands for.

The vision of the Semantic web is to argument the syntactic Web with semantic markup, so that resources are more easily interpreted by programs or intelligent (Pan *et al.*, 2006). The syntactic web is human driven. And the Semantic web enables agents or machines to crawl or move over the web and intelligently or reasonably read the web, take necessary decision or suggest solution to complex problems as in decision making systems. In an event that there are large dataset to query, an intelligent agent can reduce the time of our search algorithm to a barest minimum. Consequently, this necessitates the need to study ontology languages, their trends and related technologies.

Finally, the study of OWL development is motivated by the challenges faced in the task of the interoperability of data from different sources. Most of the available data stores are heterogeneous in nature, posing limitation to interoperability of applications in leveraging a wealth of data managed by other applications. The Semantic Web technologies are to a large extent been employed in data integration. The use of ontologies for knowledge sharing, heterogeneous database integration, and semantic interoperability has been

long realized (Necula, 2012) and proven more profitable and efficient for mashup applications.

The remaining part of this paper is structured to cover the scope of OWL, its usage in AI systems, and the promise it holds in developing future autonomous system.

2 The Scope of OWL

Data modeling have been an area of interest ever since the field of artificial intelligence began to promote the concept knowledge representation. Different measures have been taken to model and manage data. Some of these include relational database management systems such as Oracle and MySQL database management systems. But the Semantic Web provides machine readable pattern for data modeling. Extensible Markup Language (XML) is first in line of these data modeling languages, though it could be closely said to be related to Hyper Text Markup Language (HTML) which is seen more as a structuring and styling language than a semantic-oriented language. HTML may be used for data structuring, though not formally acceptable, but it can be observed that there is a slight use of it in data modeling (Ghobadi et al, 2011). XML WRAPper (XWRAP), a semi-automatic wrapper-generator, that is used to build on the structural meaning of specific HTML tags (such as tables and headings tags) and how they can be used in data modeling (Ghobadi et al, 2011). XML is more *semantic* than HTML – in fact it will be right to say HTML is not *semantic* all. Data modeling in XML provides us with a data that is structurally related with little or no semantic relations among those data, except that the data are hierarchical arranged out in such that a well written supportive application may be able to derive some semantics from the data modeled with XML.

However, Semantic Web vision which is been promoted by the World Wide Web Consortium, helps to provide better ontology languages that adds semantics and structuring to data been modeled with them. And this Semantic Web initiative, pioneered by Tim Berners-Lee, have greatly helped in advancing the Semantic Web

technologies to be more *semantic* and agents supportive just as the author have dreamt of before.

RDF is another data modeling language in semantic web. Data representation in RDF is done in an object-attribute-value triple. This simply implies that data is model in the form of a statement. RDF has been given the syntax of XML, though it adds more *semantics* to a data modeled with it than XML usually does. For instance, to express the statement Professor “John is a Lecturer who teaches Advance Programming”. In XML, we could represent it as

```
<lecturer name="Prof John">
  <teaches>Advance
  Programming</teaches>
</lecturer>
```

However, this data representation is not *semantic* enough, though it reflects a hierarchical representation of information. RDF on the other hand would represent the same data as shown.

```
<rdf:Description rdf:about="John">
  <rank>Professor</rank>
  <teaches>Advance
  Programming</teaches>
</rdf:Description>
```

This clearly tells us that there is a resource which is identifiable by the name **John** and he is a **Professor** and as well **teaches** a course called **Advance Programming**. RDF is a language that allows one to model data by describing them using the concept of resources. RDF Schema (RDFS) is used to specify a particular domain upon which the data modeled by RDF is based, and RDFS provides modeling primitives for expressing information (Antonioni et al, 2004). Furthermore, research efforts geared towards the provision of more powerful ontology language resulted in what we call DAML+OIL. The W3C group saw DAML+OIL as a good point to improve on the pursuit of developing and standardizing a more powerful ontology language. This move has led to what is referred to as OWL. Ontology languages allows users to write explicit formal conceptualizations of domain models and the main requirements are; a well-defined syntax, a formal semantics, convenience of expression, sufficient

expressive power and efficient reasoning support (Antoniou et al, 2004). Observe that if part of the requirement of ontology languages is to provide efficient reasoning support, then we sense why a technically the fields of intelligent agents and Semantic Web are inseparable. Several reasoning and rule engines have been developed and used over ontologies written with some of these ontology languages. Some of these reasoners are Fact plus plus (FACT++), Jess rule engine and Renamed Abox and Concept Expression (RACER).

OWL, an ontology language, has three flavors, namely OWL 1 DL, OWL 1 Lite and OWL 1 Full. OWL Lite is the lightest of these flavors with provision for basic constructs that can be used to develop an ontology. It has constructs for making Classes, Subclasses, Collections, Object and Data properties, and Restrictions. However, some OWL constructs that are usable in other variants of OWL are not used in OWL Lite. Some of which includes owl:unionOf and owl:complementOf. On the other hand, OWL DL is more complex than other flavors of OWL. It supports those users who want maximum expressivity while retaining computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). OWL DL includes all OWL constructs, though it can be used under certain restrictions. Finally, OWL Full is another flavor of OWL. For users who want to leverage on the entire power of OWL, OWL Full is the best option. This is because it has all the constructs of OWL and does not have the restrictions of OWL DL. Consequentially, this makes OWL full not to render a computational guarantee. According to W3C, the relationship that exist between these flavors is thus: Every legal OWL Lite ontology is also a legal OWL DL ontology and every legal OWL DL ontology is a legal OWL Full ontology and every valid OWL Lite conclusion is a valid OWL DL and every valid OWL DL conclusion is a valid OWL Full conclusion (<http://www.daml.org/2000/10/daml-ont.html>). OWL 1 DL and OWL 1 Lite corresponds to description logic SHOIN and SHIF respectively, while OWL 1 Full correspond to an undecidable logic that contains SHOIN. Figures 2 and 3 illustrate

graphical illustration and text file respectively of a typical ontology model for a domain of university.

OWL 2 is an extension of OWL 1. It was made with the intention of providing ontology developers with a language that ultimately enhance machine readability and it is exchanged in RDF/XML format. Three sublanguages that exist in OWL 2 are OWL 2 QL, OWL 2 RL and OWL 2 EL. This OWL 2 profiles have their peculiarity and specific applicable scenarios. OWL 2 QL is has quite some limited expressivity power compared to the other variants. It is suitable for applications that model large number of instances in the ontology. And basically, it enables the compositions of query that are make up its answer from other databases – such as relational database. The QL its name indicates the *query-centric* nature it provides – enabling query to be translated into relational query pattern as in SQL. Reasoning adds more rule based expressivity and efficiency to data model in ontology. Hence, when developers are willing to achieve more efficient application performance at the detriment of obtaining the full expressivity of OWL, then OWL 2 RL fits well into this. Rule engines such as Jess may be used alongside OWL 2 RL for the implementation of the rules engendering the reasoning functionality. The RL in the profile name indicates that rule language can be used in implementing reasoning. And lastly, OWL 2 EL owes its profile acronym to description logic's concept of existential quantification. It is relevant to applications that need more classes and or properties.

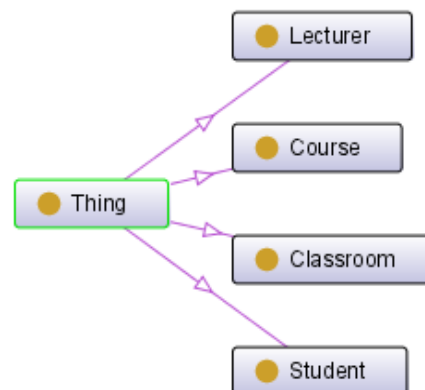


Figure 2: Hierarchical Description of the Class


```

<owl:ObjectProperty rdf:about="&journal;offers">
  <rdfs:range>
    <owl:Restriction>
      <owl:onProperty rdf:resource="&journal;offers"/>
      <owl:someValuesFrom rdf:resource="&journal;Course"/>
    </owl:Restriction>
  </rdfs:range>
  <rdfs:domain>
    <owl:Restriction>
      <owl:onProperty rdf:resource="&journal;offers"/>
      <owl:someValuesFrom rdf:resource="&journal;Student"/>
    </owl:Restriction>
  </rdfs:domain>
</owl:ObjectProperty>
    
```

Figure 3: The OWL file

3 Rules Implementation on OWL

There exist different rule languages, however, in this section; we concentrate on Semantic web rule Language (SWRL). SWRL enables developers to write rules that may be seen as OWL constructs and it enables reasoning over OWL individuals (O'Connor *et al.*, 2008). It provides some powerful features such as the one that makes it support built-ins (Horrocks *et al.*, 2004). It was designed to be the rule language of the semantic web and includes syntax of Horn-like rules. An OWL file literally contains a SWRL file and it adds to the existing axioms of OWL. And SWRL rules are implemented as the instances of OWL built-in class called `swrl:Imp` (Mei,). SWRL rules when added to OWL file provides Reasoners with a more expressive ontology from which they can reason out both what is a fact and what is an inference drawn out. Some of the reasoners used alongside rule engines are Hermit, Fact++ and Pelletes.

So, writing an SWRL to be used on ontology will entail something like:

$$A(?v) \wedge p(?v, c) \wedge \dots \rightarrow B(?v)$$

Where *A* and *B* belongs to a set of Classes model in the OWL file, *c* denotes constants and *p* belongs to a set of properties in the OWL file. For example, we could write a rule which states that if a student is a male and offers COS 205, then he is eligible to live in Duke Hostel, then we write it thus:

$$Student(?s) \wedge offer(?v, COS205) \rightarrow DukeHall(?s)$$

It must be noted that this is a very simple rule and a more complex and compounding one could be written so as to derive a stronger inference.

4 Intelligent Agents Systems and OWL

What sounds like science fictitious decades ago are now been carried out with breakthroughs made in information technology. Imagine proposing intelligent agents that can run around on the web doing complex and time consuming tasks for the users. This will sound like been too ambitious. However, semantic web ontologies are now been annexed as data modeling base for inference making applications in to actualize autonomous systems.

In section 3.0, we discussed the use of rules in OWL while section 2.0 highlighted the scope of OWL. Some constructs available with OWL makes it more expressive and logical in such a sufficient way that renders it convenient for an agent to make intuition or inference from the model data. For example, consider the `owl:Restriction` construct which defines condition for the values of a property or relations. It provides three categories of restrictions placed on the value of a property. And these are Quantifier Restriction, Cardinality Restriction and Value Restrictions. The first restriction includes the existential and universal restriction, the second restriction `minCardinality` (at least one), `maxCardinality` (at most) and `Cardinality` (equal). And the third category of restriction demands that the value of a property must have a specified single value or drawn from a collection of values. All these restrictions form an anonymous class consisting of individuals that meets the condition of the restriction. Now, these categories of restriction have their place in enabling intelligent agent to make inference over an underlying ontology. More so, we have stated in Section 3.0 that SWRL is part of an ontology file whose axioms extends the axioms of the ontology itself. Hence, it is clear that an ontology enrich with well-constructed rule sets will provide an intelligent agent a powerful knowledge base that improves its reasoning and learning capability.

Much of have been done in harmonizing the semantic web and the field of artificial intelligence. Semantic web applications have been coupled with lone agents or cooperating agents –multi agent systems (MAS). Technically, miniaturization of ontology files is a convention ontology developers seek to stick to. And in such situations, multiple agents are coordinated together to control the modularized ontology files of the entire systems. Distributed system is a phenomenon employed in high performance computing. The concept promotes solving complex problem within a short time by modulating the problem into chunks that each node in the distributed system can handle. Communication gap is been handled by a protocol known as message passing interface (MPI). Likewise, we could assemble a complex semantic web system that draws up its web services from various service providers possibly from different locations on the globe. An agent is programmed to man a particular service and task with managing and rendering such service. Now, ontology is invariably interpreted as logical knowledge base. Basically, ontology may be seen as a set of vocabulary, the semantic connections, instances and rules set that both enrich the ontology and as well improve inference making. Hence, harmonizing the modularized ontology and the agents forms a high performing web application that leverages the coordinated services of distributed semantic web knowledge based and agents to perform such complex task within a short time. We illustrate this concept in equation 1.

$$Wb(m) = \alpha_0 + \sum_{i=1}^n (O_i + A_i) * d_i \dots \text{eqtion 1}$$

The function $Wb(m)$ denotes the web application that *gathers* the services of the distributed agent. And it calculates the overall cost of implementing providing the onerous service amassed from coordinating each of the nodes in the distributed system. Hence, we call $Wb(m)$ *gather cost*. α_0 stands for the waiting time experienced at the gathering point. While O_i represents the summation of the cost of updating the ontology, querying the ontology by the agent, A_i indicates the cost of executing the reasoning faculty of the intelligent agent. d_i is the distance between each node.

We observe d_i is a major determinates in the weight of $Wb(m)$ and also, as the value of n increases, the corresponding value of *gather* function also increases. Hence, we suggest that at the gathering point – were the result of the entire distributed system is rendered to the user – another agent, possibly named *INTUI* intuitively learn the proximity between nodes and then assigns an agent to jointly handle the tasks of such nodes whose proximity value is either equal to or less than a given threshold denoted by δ . Hence, as the number of agents are been slashed down by the increased number of nodes counted by *INTUI*, we argue that the overall value of $Wb(m)$ also lowers. We maintain that the same high performance obtained by $Wb(m)$ can still be relatively upheld even with this agent reduction effect. Therefore, we rewrite equation as thus:

$$Wb(m) \approx \lim_{n \rightarrow \infty} \left(\frac{\alpha_0 + \sum_{i=1}^n (O_i + A_i) * d_i}{2} \right) \leftrightarrow \{ \exists P | P \in P \} \dots \text{equation 2}$$

P= all proximity gained from close nodes.

Equation 1

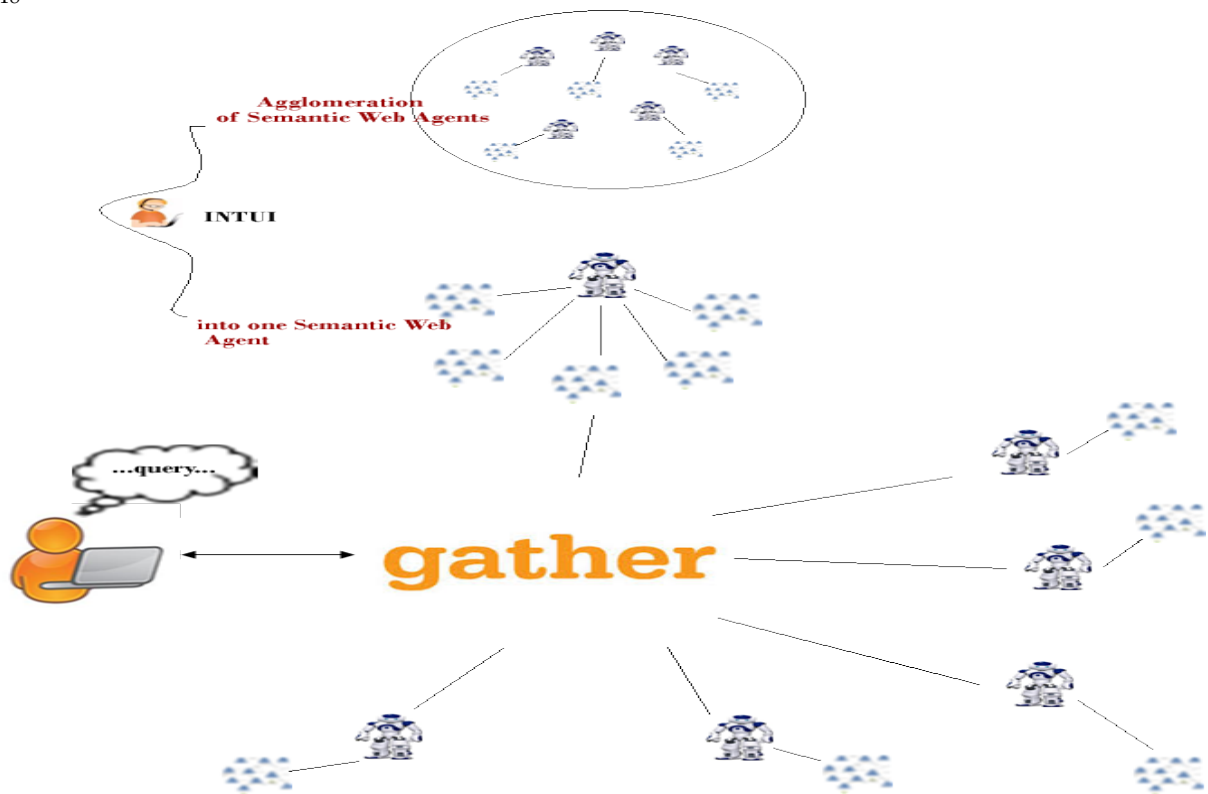


Figure 4: Using Proximity to Agglomerate Agents

We further illustrate this concept with a model in Figure 4.

Equation 2 shows that as n increases, the approximate value of $Wb(m)$ will almost be halved.

The mathematical exegesis done above does not imply that sole agent driven web applications are not efficient. But we sense that the era of one point data repository seems to be coming out of fashion in software development. Linked Data initiative has even further provided agents to crawl over web data across largely spatial location in a bid to answer a query. Hence, we only seek to canvas for the use of multiple agents representing nodes in speedily tackling complex query or answering query whose response is expected to be highly precise and also access web data whose sources are sparsely located. And should we want to over task an agent in answering such difficult query or tackling a complex task, then we might be working against the proven principle of efficiency in

computing upon which high performance computing rests.

Foundation for Intelligent Physical Agents (FIPA) is an example of implementation of multi agent framework.

5 Future of OWL in AI

We sense that research continues in the field of semantic web, web applications that portends to be fictitious even as at this level of development in semantic web may soon become realized. And since the concept of agglomeration discussed in the paper suggest that considerably large distributed agents, that solve complex problems, can be reduced intelligently, then we envisage the implementation of super web applications with

higher response rate. Recall that the semantic web promotes the concept of reasoning over data source that are stored across different locations and then retrieving those that satisfies a given query. We present our idea of what multi agent systems can be harnessed to do especially when they are further reducing in size through agglomeration, though maintaining efficiency and output.

1. Some semantic web based applications demonstrating some aspect of human organs and systems level. The human body is a complex system and research in the semantic web when focused in this direction may result into web applications thinking and acting like men. We perceive that such breath taking applications are feasible.
2. Large, though sparsely stored, information are on the web and each of this data stores is rich in its own are of peculiarity. We imagine information retrieval across the entire web that does not just thematically mined, but each theme is considered alongside the spatial and temporal data associated with such information. Domains of News broadcasting and others may benefit from this. Note that the data repository of such information is virtually the entire web data.
3. The social network is a system that is well embraced by people of all walks of life and the level of acceptance it has enjoyed will continue to grow. Interacting with such applications mostly rely on the intellectual and emotional disposition of the user. Do you imagine a personal intelligent agent tweeting for you, thinking as you will think in responding handling your social networks? This is not fiction but we are at close to using such apps.
4. E-Commerce systems now incorporate different web services in satisfying consumers need. Purchase entails knowing and finding what to buy, brokering, negotiating, choosing a particular one, paying for it and shipping it for delivery.

Though a shades of this systems are been harmonized as web services, however, we see unexplored areas that will make this experience wholesome.

5. Several research institutes are scattered across the globe and each making great research findings in different fields. Technically, there exist some levels of intercreativity among fields of study, not to mention those fields of research that are closely related. And ethically, most researchers tend to keep mum or low until their research are successful. However, we look forward to a harmonizing web application where each of this research locale are nodes in the distributed systems which helps researchers in enriching their thoughts and ideas as it intelligently retrieve information for them.

6 Conclusion

We have reviewed the syntax of OWL, how ontology may be model from it and highlighted other semantic web languages that it interoperates with. Furthermore, we underpinned the propensity of OWL in developing intelligent agents, and also reviewed some novel OWL based intelligent agents. Finally, we argued that agglomerate of semantic web based intelligent agents may further future research efforts in building complex web based autonomous systems.

References

1. Medić A. and Golubović A., (2010). Making Secure Semantic Web. *Universal Journal of Computer Science and Engineering Technology*, Vol. 1, No. 2, pp. 99-104.
2. Antoniou, G., and Frank, V. H., (2004). *Semantic Web Primer* MIT Press, Cambridge, Massachusetts, London, England.
3. DAML+OIL. Retrieved 28 September 2013 from <http://www.daml.org/2001/03/daml+oil-index.html>.

4. DAML-ONT. Retrieved 28 September 2013 from <http://www.daml.org/2000/10/daml-ont.html>.
5. Ghobadi, A., and Maseud, R., (2011). *An Ontology-based Semantic Extraction Approach for B2C eCommerce*.
6. Girdhar, P., Suresh, K., and Manjeet, S., (2012). *Agent Oriented Schemes for Trustworthy Semantic Web Services*.
7. Horrocks I, Patel-Schneider PF, Boley H, Tabet S, Grosz B, Dean M., (2004). *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. W3C Member Submission.
8. Gruber, T., (1993). *A Translation Approach to Portable Ontologies*, Knowledge Acquisition, 5(2) pp 199-220.
9. Jeff Z. Pan and Ian Horrocks- RDFS (FA): Connecting RDF(S) and OWL DL, 2006.
10. Jena ARQ. Retrieved September 28, 2013 from www.jena.sourceforge.net/ARQ/Tutorial/filters.html,
11. Knublauch, H., (2005). *Ontology-Driven Software Development in the Context of the Semantic Web: An Example Scenario with Protégé/OWL*, Stanford Medical Informatics, Stanford University, CA.
12. Lawson, T., (2004). *A Concept of Ontology*.
13. Martin O'Connor and Amar Das- SQWRL: a Query Language for OWL. CEUR conference proceedings 2009.
14. O'Connor M, J., Ravi D. S, Csongor N, Samson W. T, Amar K. D. (2008). *Developing Web-Based Application Using OWL and SWRL. AAAI Spring Symposium: AI Meets Business Rules and Process Management* Pg. 93-98
15. Mohan, R. A., Arumugam, G., (2011). *Constructing Railway Ontology using Web Ontology Language and Semantic Web Rule Language*.
16. Necula, S. C., (2012). *Implementing the Main Functionalities Required by Semantic Web Search in Decision Support Systems*.
17. Nováčak, V., (2010). *Eureka! Towards a Practical Emergent Processing Knowledge*, National University of Ireland, Galway.
18. OIL. Retrieved 28 September 2013 from <http://www.ontoknowledge.org/oil/>.
19. Pellet. Retrieved September 28, 2013 from <http://pellet.owldl.com>.
20. Rubbani, H., (2007). *Semantic Web Solutions*, M.Sc. Thesis, IT University of Copenhagen.
21. Sören, A., Christian, B., Georgi, K., Jens, L., Richard, C., and Zachary, I., (2008). *DBpedia : A Nucleus for a web of Open Data*.
22. SWRL Built-in. Retrieved September 28, 2013 from <http://www.daml.org/2004/04/Swrl/builtins.html>.