Volume 6, Issue 1, January 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Representation of Knowledge in Multi-Agent System for Club Membership and Ontology Development using RuleML

[1]Mayur M. Patel , [2]Dr. Priya Swaminarayan
Research Scholar, Rai University, Ahmedabad
[1] Assistant Professor, Computer Science Department, Natubhai V. Patel College of Pure and Applied Sciences, Vallabh Vidyanagar, Gujarat, India
[2] Professor and Head, MCA Department, ISTAR, Vallabh Vidyanagar, Gujarat, India

Abstracts: Current Web content has been designed for direct human processing and thus lack of computer processing elements. Semantic Web aims at computer process-able information. The semantic Web is considered to be the next generation web, so a lot of research and development is going on. In the era of Semantic Web, the ontology has established as a powerful tool to enable knowledge sharing and it is an important means in Semantic Web to achieve the semantic interoperability among heterogeneous distributed systems. Both ontology and RuleML are central to the semantic web, and their combined use will enable the restriction of the ontology in adaptable and extensible manner. We have partially implemented MAS for ClubMembership Ontology in which we used RuleML as a mark-up language for sharing and publishing rules. This paper focuses on the utilization of combining both Ontology and RuleML structure towards system integration for Club Member. We have used Web Ontology Language (OWL) for the development of domain ontology for the Club and RuleMl as an agent Implementation of ClubMembership Ontology.

Index Terms-- Ontology, RuleML, Semantic Web, OWL

I. INTRODUCTION TO SEMANTIC WEB

The Semantic Web is a Web of data. We can use so many data every day and all are not part of the web. For example, we can see our Credit Card statements on the web, and our photographs, and we can see our appointments in a calendar. But can we see my photos in a calendar to see what we were doing when we took them? Can we see Credit Card statement lines in a calendar? Why not?, because we have a web of documents but we do not have a web of data. Because data is controlled by applications, and each application keeps it to itself [1].

Semantic Web technologies can be used in a variety of application areas. For example, in data integration, whereby data in various locations and various formats can be integrated into one, seamless application, in resource discovery and classification to provide better, domain specific search engine capabilities.

Tim Berners-Lee has created Semantic Web Layer Cake model which illustrates the hierarchy of languages which can be used for the implementation of Semantic Web. 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[2]. Following figure shows Semantic Web Layer Cake Model.

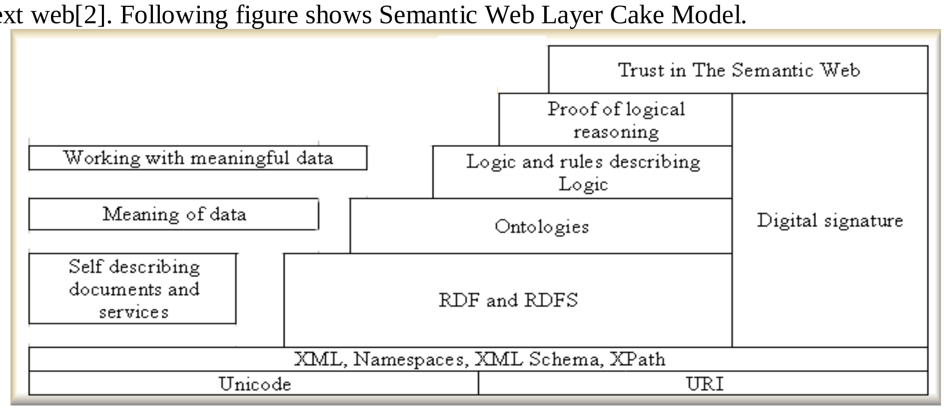

Figure – 1: Semantic Web Layer Cake Model proposed by Tim Berners-Lee

The semantic Web is a

- Set of technologies: semantic graphs, description logic, …
- Set of standards: RDF, OWL, HTTP, RDFa,
- Set of Tools: Protege, Jena, Sesame, Semantic Media Wiki, ...
- Set of artifacts: DBpedia, Cyc, Linkeddata.org[3]

Semantic web good for [4]

- Improved search (now)
- Improved classification (now)
- Facilitate controlled vocabulary development (now)
- Improved selective information dissemination (now)
- Information (data + schema) integration (research)
- Data mashups, visualization (prototypes)
- Automated web services synthesis (research)

Current Web content has been designed for direct human processing and thus lack of computer processing elements. Semantic Web aims at computer process able information. The semantic Web is considered to be the next generation web, so a lot of research and development is going on.

II. INTRODUCTION TO ONTOLOGY

Ontologies define the concepts and relationships used to describe and represent an area of knowledge. Ontologies are used to classify the terms used in a particular application, characterize possible relationships, and define possible constraints on those relationships. Building upon RDF and RDFS, OWL defines the types of relationships that can be expressed in RDF using an XML vocabulary to indicate the hierarchies and relationships between different resources. In fact, this is the definition of "ontology" in the context of the Semantic Web: a schema that formally defines the hierarchies and relationships between different resources. Semantic Web ontologies consist of taxonomy and a set of inference rules from which machines can make logical conclusions. [6]

Rules on Semantic web

The term "rules" in the context of the Semantic Web refers to elements of logic programming and rule based systems bound to Semantic Web data. Rules offer a way to express, for example, constraints on the relationships defined by RDF, or may be used to discover new, implicit relationships.

To define various rules in single rule language called "Horn" rule, which has the form "if conditions then consequence", but it places certain restrictions on the kinds of conditions and consequences that can be used.

A general example may help. While integrating data coming from different sources, the data may include references to persons, their name, homepage, email addresses, etc. However, the data does not say when two persons should be considered as identical, although this is clearly important for a full integration. An extra condition can be expressed stating that "if two persons have similar names, home pages, and email addresses, then they are identical". Such condition can be naturally expressed with Horn rules.

Introduction to RuleML

RuleML is used to transcribe and refine our ontology as a knowledge base, consisting of facts and rules. The Datalog (constructor-function-free) sublanguage of Horn logic is the foundation for the kernel of RuleML. Datalog is the language in the intersection of SQL and Prolog. It can thus be considered as the subset of logic programming needed for representing the information of null-value-free relational databases, including (recursive) views. That is, in Datalog we can define facts corresponding to explicit rows of relational tables and rules corresponding to tables defined implicitly by views.RuleML Datalog, being a markup language, can conveniently represent relational information where all of the columns are natural-language phrases. To explain the Datalog features we will develop a small example formalizing natural-language business rules in RuleML.[7]

Partial Development of ClubMembership Ontology

In this paper, we have represented knowledge of Club Membership Domain. The Domain Ontology for this paper is the Club Memebrship Ontology. Figure-3 shows concepts and the relationships existing within its domain. In Club Ontology, we have defined main five concepts namely, Company, Customer, Hotel, Reservation; and association between concepts also represented in the following diagram.

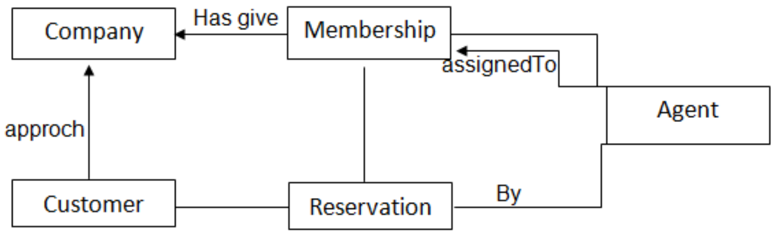

Figure – 3: Domain Ontology for Club Membership

The Domain Ontology needs to be translated into a machine readable language and for this we have utilized OWL to achieve this goal. As OWL is written in XML, therefore it has all the benefits that XML can provide. It allows information to be exchanged easily across different platforms on top of that; it also allows the exchange to be taken place between different applications. OWL also captures information relating to classes, properties, attributes as well as the constraints displayed by UML and OCL together. The Web ontology language (OWL) can formally describe the semantics of classes and properties used in Web documents. For machines to perform useful reasoning tasks on these documents, the language must go beyond the basic semantics of RDF Schema.

In Club Membership domain ontology, we have defined the following OWL classes for Club Membership Environment.

```
<owl: Class rdf:ID=”Company”/>
<owl: Class rdf:ID=”Agent”/>
<owl: Class rdf:ID=”Hotel”/>
<owl: Class rdf:ID=”Rate”/>
<owl: Class rdf:ID=”Reservation”/>
```

In OWL, We can also define the relationship between the classes. Following are the statements in OWL, which expressed the relationship between classes of Club.

```
<owl: ObjectProperty rdf:ID=”hasMember”>
        <rdfs: domain rdf:resource=”#Agent”/>
        <rdfs:domain rdf:resource= “#Hotel”/>
</owl: ObjectProperty>
```

Data type can also be defined in OWL through, owl:DatatypeProperty. For example, AgentID is a string; therefore it can be expressed as:

```
<owl: DatatypeProperty rdf: ID=”AgentID”>
        <rdfs:domain rdf:resource=”#”/>
        <rdfs:range rdf:resource=”&xsd; string”/>
</owl: DatatypeProperty>
```

Ontology is usually expressed in a logic-based language, so that detailed, accurate, consistent, sound, and meaningful distinctions can be made among the classes, properties, and relations. Some ontology tools can perform automated reasoning using the ontologies, and thus provide advanced services to intelligent applications such as conceptual/semantic search and retrieval, software agents, decision support, speech and Ontology provides the fundamental in describing concepts as well as their relationships.

III. IMPLEMENTATION OF CLUB MEMBERSHIP ONTOLOGY USING RULEML

There are many languages one could choose for this purpose, but with the Benefit and the fact that OWL is XML-based, an XML-based language such as RuleML is really the only approach we would seriously consider. RuleML currently appears as the most logical choice for representing rules in the context of OWL ontologies.Our approach to the rule construction process is that of building RuleML rules on top of OWL ontologies. We assume that the ontologies are pre-defined and that all their classes are available to be used as elements in the rules.

For example:

<_head><atom><_opr><rel>**Club**</rel>…

A domain theory can be viewed as a network that defines the inter-relationships among the theory’s axioms or rules. If we select a single node representing the head of a rule and consider only its descendants, we can create a view that forms a tree (with possibly replicated nodes). Such a tree representing a subset of a hypothetical domain theory for battlefield scenarios is depicted in the nodes.

```
<imp><head><atom><opr><rel>isMember of Club</rel></_opr><var>Customer</var></atom>
</head><body><and><atom><_opr><rel>isgetPackage</rel></_opr><var>Customer</var></atom>
</and></_body></imp></rulebase>
```

Implementation File of OWL

```
<rdfs:range rdf:resource=”&xsd; string”/></owl: DatatypeProperty>
</owl:Class></rdf:RDF><?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
      xmlns:owl="http://www.w3.org/2002/07/owl#">
<owl: Class rdf:ID=”Club”/>
<owl: ObjectProperty rdf:ID=”hasAssign”><rdfs: domain rdf:resource=”#Agent”/>
<rdfs:domain rdf:resource= “#Agent”/></owl: ObjectProperty><owl: DatatypeProperty rdf: ID=”AgentID”>
<rdfs:domain rdf:resource=”#Agent>
```

Implementation File of RuleML

```
<?xml version="1.0" encoding="UTF-8"?><imp><_head><swrl:classAtom>
<owllx:Class owllx:name="d:\college\others\high\practice\temp.owl#Club"/>
<_opr><rel>isAssigned Package</rel></_opr><ruleml:var>Agent</ruleml:var>
</swrl:classAtom></_head><_body><and><swrl:classAtom>
<owllx:Class owllx:name="d:\college\others\high\practice\temp.owl#Club"/>
<opr><rel>isMember of Club</rel></_opr>
<ruleml:var>Customer</ruleml:var></swrl:classAtom></_head><_body>
<and><swrl:classAtom>
```


<owllx:Class owllx:name="d:\college\others\high\practice\temp.owl#Club"/>
<_opr><rel>isAttendCourses</rel></_opr>
<ruleml:var>Customer</ruleml:var></swrl:classAtom></and></_body></imp>…</rulebase>

IV. CONCLUSION

In this research, we can develop a Partial Club based Ontology System using RuleML is approach to provide different services. Also, it discovers a seamless information processing system across an organization that the user can access from any location. The implementation of a computer understandable representation of the semantics of academic programs is complex. In this research I can develop a club membership multi agent system is capable of to select the best package for Accommodation .That's why is more user friendly. The amount of manual intervention decrease and machine understandable logic linking of data is achieved. The System can be further redefined by Comparing.

We presented a case for using RuleML to construct domain theory rules on top of OWL ontologies. The context for this work is that of reasoning about situation awareness, for which we are currently developing a Situation Awareness Assistant.We presented a SAW Core ontology and showed how it can be extended to handle domain-specific situations.

we argued that RuleML not only provides general implication in the form of Horn clauses but also that its XML representation makes it the ideal chose for use with OWL. We then showed how we use RuleML to define rules that reference relation subclasses in our core and domain specific ontologies.

REFERENCES

[1] W3C Semantic Web Activity "http://www.w3.org/2001/sw/"

[2] "Semantic Web Stack", wikipedia, the free encyclopedia http://en.wikipedia.org/wiki/Semantic_Web_Stack

[3] Semweb Central http://semwebcentral.org/

[4] Semantic Web Wiki http://semanticweb.org/wiki/Main_Page.

[5] FIPA "Ontology Service Specification" Document Number : XC00086D] Document Source: FIPA Architecture Board www.fipa.org

[6] Proceedings of the 1st Workshop on Context, Information and Ontologies, **(2009)** May 31-June 9; Heraklion, Griechenland.

[7] Design Rationale of RuleML:A Markup Language for Semantic Web rules by Harold Boley1, Said Tabet2 and Gerd Wagner3

[8] Constructing RuleML-Based Domain Theories on top of OWL Ontologies Christopher J. Matheus1, Mitch M. Kokar2,

