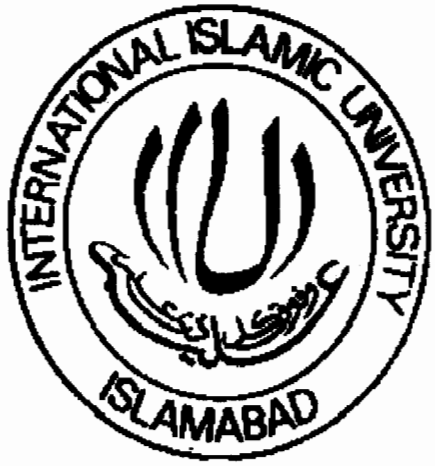
# Ontology-based Semantic Sear
# Holy Quran






*Developed By:*

**Hikmat Ullah Khan**
**(394-FBAS/MSCS/F07)**

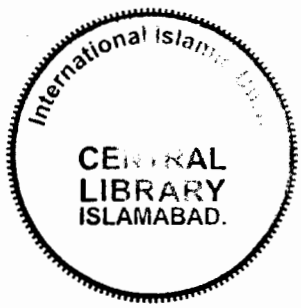
*Supervisor*

**Syed Muhammad Saqlain**
**Assistant Professor, DCS, FBAS, IIUI**

*Co-Supervisor*

**Dr. Khalid Latif**
**Assistant Professor, SEECS, NUST**

**Department of Computer Science**
**Faculty of Basic and Applied Sciences**
**International Islamic University Islamabad**
**2009**



A Thesis Submitted to the

**Department of Computer Science**

International Islamic University Islamabad

as a partial fulfillment of requirements for the award of

the degree of

**MS in Computer Science**

**Dated: .....**

# Final Approval

It is certified that we have examined the thesis titled "Ontology based Semantic Search in Holy Quran" submitted by Hikmat Ullah Khan, Registration No. 394-FBAS/MSCS/F07, and found as per standard. In our judgment, this research project is sufficient to warrant it as acceptance by the International Islamic University, Islamabad for the award of MS Degree in Computer Science.

## Committee

**External Examiner**

**Dr. Muhammad Zubair**

Assistant Professor

Department of Computing,

Ripahah International University, Islamabad

**Internal Examiner**

**Mr. Muhammad Imran Saeed**

Assistant Professor

Department of Computer Science

International Islamic University, Islamabad

**Supervisor**

**Syed Muhammad Saqlain**

Assistant Professor

Department of Computer Science

International Islamic University, Islamabad

Dedicated To:

who can not read

**Holy Quran**

but believe that it has been revealed from

**Allah Almighty**

to His beloved

**Prophet Muhammad**

(Peace and Blessings may ALLAH be upon him)

# Declaration

I hereby declare that this work, neither as a whole nor a part of it has been copied out from any source. It is further declared that I have developed the model, the software on the base of proposed model and the results with my personal efforts; and under the sincere guidance of Syed Muhammad Saqlain and Dr. Khalid Latif. If any part of this project is proved to be copied from any source or found to be reproduction of some other project, I shall stand by the consequences. No portion of the work presented in this dissertation has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

Hikmat Ullah Khan
(394-FBAS/MSCS/F07)

# Acknowledgement

I simply bow my heads before Almighty Allah for giving me faith in my abilities and enabling me to accomplish this work and granting me with His Special Mercy, Blessings and Unlimited Help throughout the phases of the research work.

I conduct my profound regards to project supervisor Mr. Syed Muhammad Saqlain and co-supervisor Dr. Khalid Latif, who provided me their cooperation, guidance and valuable support during the all stages.

I would like to offer my special thanks to Mr. Muhammad Aithsham Aslam Assistant Professor CIIT Lahore, Mr. Muhammad Shoaib Lecturer Allama Iqbal Open University, Muhammad Shahzad Faisal & Mr. Tahir Afzal Malik Lecturers CIIT Attock and Mr Yasir & Mr. Nabeel Ahmed SEECS NUST for their suggestion and comments.

At the last but not the least; I would like to adjudge acknowledgment to my family members. It surely would not have been conceivable without the orisons of my beloved mother.

Hikmat Ullah Khan

(394-FBAS/MSCS/F07)

# Abstract

Time is one of the most important factors in human life. Search is operation that is used for finding desired data or information from data source in less time. Already existing method of keyword has had been extensively used with its feature but it causes the main problem of irrelevant information retrieval. Many works have been done to accomplish keyword search from Holy Quran. Many variations, in this regard, have also been attained ranging from simple keyword search to static topic search. The main problem in all these works is that these are either static or they does not provide us semantic search.

In this thesis, it has been proposed that the concepts of ontology of semantic web can be applied for carrying out semantic search from Holy Quran. Ontology is explicit specification of concepts, thus it is conceptual framework for defining mechanism for inference level work. For this purpose, exploratory search have been done from semantic web field of knowledge. Then recommendations have been made regarding choice of basic building block for ontology, ontology modeling paradigm, ontology language, ontology development tool, ontology query language, reason engines etc.

The sample domain ontology, based on living creatures including animals and birds mentioned in Holy Quran, has been developed in protégé ontology editor tool. The ontology consists of complete class hierarchy, object properties along with domain and range, data properties and sample instances. Various SPARQL Queries have been run to depict the proper role of ontology. Then certain recommendation for the project of attaining semantic search from all domains and resultantly all text of Holy Quran has been proposed. These recommendations include model and framework including creation of Quranic WordNet, integration, merging and mapping of domain ontologies under the umbrella of upper ontology. This work can be extended to other Islamic knowledge sources like Hadith, Fiqh etc.

# Table of Contents

1. Introduction.....1

1.1 Motivation ..... .2

1.2 Introduction.....3

1.2.1 Overview of the Holy Quran .....4

1.2.2 Subject Matter of the Holy Quran ..... .5

1.2.3 Users' preferences while Querying the Quranic Text ..... 5

1.3 Semantic and Semantic Search.....6

1.3.1 Semantic Search in Holy Quran .....7

1.4 The Roadmap .....7

1.5 Objectives.....8

1.6 Scope of the Research.....8

1.7 Thesis Outline .....9

2. Literature Survey .....,,,..10

2.1 Related Research .....12

2.1.1 XML Semantics Rules for Holy Books .....12

2.1.2 XML Semantics Checker Model for Quran .....13

2.1.3 Mophological Analysis of the Holy Quran .....13

2.1.4 Keyphrase Extraction for Islamic Knowledge Ontology .....13

2.1.5 Build Islamic Ontology based on Ontology Learning.....13

2.1.6 OSIS .....14

*2.1.6.1 Why was XML chosen to be the foundation of the OSIS standard?* 14

*2.1.6.2 Is OSIS Model Applicable to the Quran?* .....15

2.2 Present Quranic Software .....15

2.3 Semantic Search.....17

2.4 Summary .....20

3. Problem Analysis .....21
3.1 Holy Quran's Unique Style .....22
3.1.1 The allegorical nature of the Holy Quran .....22
3.1.2 Scattered Discussion of Topics .....23
3.1.3 The Name of a Chapter and their Text .....24
3.2 Information Retrieval or Extraction from the Holy Quran .....24
3.3 Main Problem of Irrelevant Verse Retrieval .....25
3.4 Problem Domain ..... 26
3.4.1 Problem Statement ..... 27
3.5 Obstacles in Developing Algorithm for Semantic Search in Quran.. 27
3.6 The Planned Solution.....28
3.6.1 The Aim.....29
3.7 Summary .....29
4. Exploratory Research.....30
4.1 Concept of Semantic .....31
4.2 Semantic Search .....31
4.3 Semantic Web.....32
4.3.1 Problem of Current Web .....32
4.3.2 Concept of Semantic Web .....32
4.3.3 Semantic Web Present and Future .....33
4.4 Ontology .....34
4.4.1 Ontology vs Conceptual Modeling .....34
4.5 Features of Ontology .....35
4.6 Categorization of Ontology .....35
4.6.1 Categorization on the basis of internal structure.....35
4.6.2 Categorization of ontologies based on Conceptualization .....36
4.7 Benefits of Domain Ontology .....37
4.8 Components of Ontology .....38

4.8.1 Class .....38
4.8.2 Individuals .....38
4.8.3 Properties .....38
4.8.4 Types of Properties .....39
4.8.4.1 Symmetric Properties .....39
4.8.4.2 Transitive Properties .....39
4.8.4.3Functional Properties .....39
4.8.4.4 Inverse Properties .....39
4.8.4.5 Inverse Functional Properties .....39
4.9 Ontology Language .....40
4.9.1 What is ontology language .....40
4.9.2 Features of ontology Language .....40
4.9.3 Requirements of ontology languages .....40
4.9.4 Types of Ontology Language .....41
4.9.4.1 Resource Description Framework .....41
4.9.4.2 RDF Schema .....41
4.9.4.3 DAML+OIL .....41
4.9.4.4 Web Ontology Language (OWL) .....41
4.9.4.5 Types of OWL .....42
4.10 Ontology Editor Tools .....42
4.10.1 Protégé .....43
4.10.2 SWOOP .....43
4.10.3 OntoEdit .....43
4.10.4 Altova Semantic Work .....43
4.10.5 WebODE .....44
4.11 Ontology Modeling Paradigms .....44
4.11.1 Introduction to Ontology Modeling Paradigm .....44
4.11.2 Differences .....44
4.12 Methodology for Ontology Building .....45
4.12.1 METHONTOLOGY .....45

4.12.2 CommonKADS and KACTUS.....46
4.12.3 SENSUS.....46
4.12.4 Uschold and King Method.....46
4.12.5 Cyc Method.....46
4.13 Reasoning Engines.....47
4.13.1 Jena.....47
4.13.2 KAON2.....47.
4.12.3 Java Theorem Prover (JTP).....47
4.13.4 Ontobroker.....47
4.13.5 FaCT++.....47
4.13.6 Pellet.....48
5. Implementation .....49
5.1 Implementation .....50
5.1.1 Selection of Domain for Sample Ontology .....50
5.1.2 Difficulties in Animal Domain .....50
5.1.3 Selection of Translation .....51
5.1.4 Precautions about Ontology development domain of Holy Quran.....52
5.2 Steps in Ontology Construction .....52
5.2.1 Determine the domain and scope of the ontology .....52
5.2.2 Consider reusing existing ontology.....53
5.2.3 Enumerate important terms in the ontology.....53
5.2.4 Define the classes and the class hierarchy.....54
5.2.5 Define the properties of classes- Slots.....56
5.2.6 Define the facets of the properties.....58
5.2.7 Create Instances.....59
5.3 Issues regarding Ontology Creation.....60
5.4 Checking Class hierarchy .....61
5.5 SPARQL Queries and Results .....62
5.5.1 Classes and subclass hierarchy checking .....62
5.5.2 SPARQL Queries for Checking of Semantics .....63

6. Recommendations..... 66

6.1 Selection of Ontology Modeling Paradigms..... 67

6.2 Selection of Basic Ontology Language..... 68

6.2.1 Problems with RDF-S.....68

6.3 Selection of Type of OWL ..... 69

6.4 Selection of Ontology Editors..... 70

6.5 Selection of Reasoning Engines .....71

6.6 Selection of different ontological Concepts..... 72

6.6.1 Upper Ontology ..... 72

6.6.2 List of Domains for ontologies from Holy Quran.....72

6.6.3 Recommendation about Techniques used with ontologies

( Ontology Merging, Ontology Integration and Ontology Mapping ).....75

6.7 Dynamic Contextual Ontology and The Fourteen Primitive Actions.76

6.8 Development of Quranic WordNet as Upper Ontology .....77

6.8.1 Introduction to WordNet.....77

6.8.2 Arabic WordNet .....79

6.8.3 Development of Quranic WordNet and other WordNet. .....79

6.9 Proposed Framework . .....80

6.9.1 Proposed Semantic Search Architecture .....80

6.9.2 Proposed Semantic Search Model .....81

6.9.3 Proposed Architecture for Semantic Search in Islamic Documents. ..... 83

7. Conclusion and Out Look .....85

7.1 Conclusion.....86

7.2 Contribution.....87

7.2 Out Look .....87

Appendices .....88

Appendix A: Sample XML Code for Ontology .....89

Appendix B: Sample Code for Protégé-OWL Java for Ontology .....103

References .....111

# List of Figures & Tables

**Fig: 5.1 Class Hierarchy .....56**

**Fig. 5.2 List of Object Properties.....57**

**Fig. 5.3 List of Data Properties.....58**

**Fig. 5.4 List of Object Properties along with Domain and Range.....59**

**Fig. 5.5 Checking of consistency of ontology by reasoner.....61**

**Fig. 5.6 List of Class and Subclass on the base of SPARQL Query.....63**

**6.1 Ontology languages used by users.....69**

**Fig 6.2 Respondents' use of ontology editors.....70**

**Fig: 6.3 Reasoning engines utilization. .....72**

**Fig 6.4 Taxonomy showing Subject matters of Holy Quran .....74**

**Fig 6.5 Techniques used with Ontology .....76**

**Fig 6.6 WordNet Extract for the concept 'car'.....78**

**Figure 6.7 The relations between domain, ontology, concept, keyword, document and user request in the semantic search system. .....80**

**Figure 6.8 An Overview of the Semantic Search Architecture .....81**

**Figure: 6.9 Ontology-based information retrieval Model .....82**

**Figure 6.10. Comparison between keyword and Semantic Search System .....83**

**Figure 6.11. Query Processing for Semantic Search System .....84**

**Table. 5.1 Enumeration of Important Terms in the domain knowledge .....54**

**Table 5.2. List of Instances vs Classes .....60**

# CHAPTER 1

# INTRODUCTION

# CHAPTER 1

# INTRODUCTION

This is the Book; in it is guidance sure, without doubt, to those who fear Allah.

(Surah Al-Baqrah: 2)

---

## 1.1 Motivation

Allah Almighty has bestowed the human kind with his paramount blessings in the form of sending His beloved Prophet Hazrat Muhammad as His Last Prophet (Peace and Blessings may Allah be upon him). Moreover, it is also great mercy on His part that He sent down his last book, the Holy Quran, to Holy Prophet (Peace and Blessing may Allah be upon him). The Holy Quran, the last and only divine scripture in its true from, is an ocean of knowledge covering subject matters from all spheres of life.

As the last decade is referred as age of information, surely it is an era of knowledge. Every one from scholar to lay man desires authenticated, reliable and relevant results of his/her query. The Holy Quran discusses and provides basics about all areas of knowledge. It is the sacred book of Muslims but the book for the whole humanity so millions of people recite it regularly and consult to get answers of their queries, solve their problems and also get Islamic point of view about a specific subject matter.

Religious scholars, teachers and students struggle hard to collect similar verses that describe a particular topic. The religious personalities relate a verse to a topic on the base of interpretation available to them. How can an acquaintance of the Holy Quran perform subjective search? It is very difficult for him. He has to spend a lot of time to get all the verses that are related to a particular issue. The task is done by the scholar without help of software so takes a lot of effort and time.

The professional of Information technology realize the problem of the users of the Holy Quran and try to facilitate them in searching the Holy Quran for specific topic. Quite a large number of current software about Holy Quran provides searching facility. The majority of the software provide conventional keyword search[1,2,3,4] But this is not good with respect to Holy Quran as keyword search misses majority of the relevant verses. A lot of effort has been carried out to better search facility by providing topic bases indexes. Many such efforts have been carried out manually in the form of books [5,6,7] and in the form static binding indexes in software usually available in internet. [8,9, 10, 11].

But the results do not satisfy the users because all these software lack the idea of semantic search.

There is requirement to develop an application that may give us dynamic conceptual search. With the emergence of semantic web as a modern field where structure can be formulated that may be understandable not only for humans but also for machines. Domain specific ontologies are created and inference from ontologies leads us to Semantic search.[12].

As the Holy Quran is a typical example of documents of unstructured data. The existing ontologies have been used for structured and semi-structured.[13].Thus there is a requirement to do exploratory research in very large field of semantic web and explore that which ontology techniques, language, tools etc are better suited for complex and unstructured document like Holy Quran. This research covers the survey of entire field of semantic search w.r.t Holy Quran. A lot of recommendations have been made and also domain ontology for animals listed in Holy Quran has been created on the basis of recommendation made and it provides the required results and authenticates the recommendations.

## 1.2 Introduction

In this section let us briefly discuss about the Holy Quran, meanings of semantic search and semantic search in Quran.

### 1.2.1 Overview of the Holy Quran

The Quran (also spelt as Quran, Koran, Coran) is the holy book of Muslims. It was revealed from Allah to the Holy Prophet Muhammad (SAW) in Arabic language in almost twenty two years. It consists of thirty sections/Parts (Parahs) and one hundred and fourteen chapters (Surahs). The sections are almost of equal length but the chapters consists of an arbitrary number of verses (Ayahs)---the smallest chapter "Surah Al-kothar" contains 03 verses whereas the longest chapter "Surah Al-Baqrah" 286 verses. Among the revealed books the Holy Quran is the only book present in its revealed language and original text.

Quran has its own style of describing the topics. At some places, some topics are explicitly mentioned while some others are meant implicitly. This implicit nature of the Holy Quran is a great challenge for the NLP and AI experts. Here the context very often helps the scholars to find the implicit meanings but some times it does not provide any clue.

The allegorical nature of the Holy Quran is also a challenge not only for NLP experts but also for the Quranic scholars. Different scholars explain allegorical terms according to their own perception/sect/knowledge etc.

The Holy Quran has unique style of explaining different topics. Normally a topic is discussed at different places. For example, the topic of Hazrat Moosa (AS) is discussed in a large number of chapters; the Oneness of Allah has been discussed through out the Holy Quran. It is not necessary that the consecutive verses must belong to the same topic. A topic may or may not be discussed in consecutive verses. Some consecutive verses or even a single verse may contain many topics. For example, the second verse of Surah Al-Baqrah, consisting of only seven words including prepositions, mentions three topics. This is a little verse and discusses three topics. While lengthy verses may explain dozens of topics. Normally a chapter does not contain only one topic and the same is true for a lot of verses.

The name of a chapter does not mean that all the text of this chapter belongs to this topic. Name of a Surah normally does not help us in perceiving the semantics of the Quranic text of that Surah. But there are certain scenarios in which it helps.

### 1.2.2 Subject Matter of the Holy Quran

The Holy Quran throws light upon almost all fields of life. Some topics are repeated time and again and some are described just once; some explicitly and some implicitly. The frequency

of repetition of a topic is actually a gesture to human to show its importance. Some topics are discussed in detail and some are slightly touched. The list of the topics upon which the Holy Quran lays emphasize is very long. Major Topics of the Holy Quran includes Oneness of Allah, Prophethood of Muhammad (Peace and Blessings may Allah be upon him), other prophets, the purpose of creation of human being, his success and failure, the life after death etc.

### 1.2.3 User' Preferences while Querying the Text of Holy Quran

Now the question arises that what sort of information there lies that should be considered for extraction from the Holy Quran? The complete answer of this question is perhaps difficult to explain. Most of the users want to perform the subject based search in the Holy Quran for a particular topic. A user wants either to know the basic concept or the detail of a topic. The Quranic text is written neither in story-writing or essay-writing style nor in question-answer style. It has its own unique style. A chapter may contain hundreds of; and a verse dozens of topics implicitly or explicitly. A system that is capable of answering the users' queries needs special attention, cooperation and team work of the experts from different areas.

We know that the automatic systems of IR or IE can not provide accuracy to a satisfactory level. These systems show good results for well structured data source. On the contrary, the precision is very low for unstructured data. The automatic annotators show good results for text data to be mapped on the ontology designed for specific domains only if the text is simple and contains limited domains. An ontology designed for a domain will not work for any other domain. The greatest issue while developing any automatic system for the semantic search in the Holy Quran is its unique style.

Much of the work to digitalize the Holy Quran has already been done by different Muslims and non Muslims as well. Some Muslim governments and organizations have expended a lot of money to develop more and more useful software of the Holy Quran. There is a lot of software and databases available on internet and majority are free of cost. A number of commercial organizations have also developed many Quranic software and maintained databases containing Islamic literature. Most of these Quranic software/databases provide the facility to see different translations in different languages. Beside this they also provide the facility to listen the Holy Quran in different reciter's (Qaries) voices. Some of them provide

query word base search facility. In word based search, the key word is compared to the Quranic text and all the verses that contain that particular query word are retrieved.

## 1.3 Semantic and Semantic Search

Linguistically, semantics is usually referred as the study of meaning. The word "semantics" itself denotes a range of ideas, from the popular to the highly technical. It is often used in ordinary language to denote a problem of understanding that comes down to word selection or connotation. [15]

The word semantics is derived from the Greek word σημαντικός (semantikos), "significant" [16] from σημαίνω (semaino), "to signify, to indicate" and that from σήμα (sema), "sign, mark, token" [17].In linguistics, it is the study of interpretation of signs or symbols which are used for some specific contexts [18]. In linguistics, semantic analysis is the process of relating syntactic structures, from the levels of phrases, clauses, sentences and the whole text, to their language-independent meanings. It causes removal of characteristics related to fussy linguistic contexts up till that level that such a project is possible. [19]

Semantic search is a process used to improve online searching by using data from semantic networks to disambiguate queries and web text in order to generate more relevant results. Semantic Search uses semantics or the science of meaning in language to produce highly relevant search results. In most cases, the goal is to deliver the information queried by a user rather than have a user sort through a list of loosely related keyword results [20].

### 1.3.1 Semantic Search in Holy Quran

Semantic search for the Holy Quran is becoming more vital day by day. Especially with the introduction of semantic web the semantic search in the Holy Quran has become necessary.

*"Semantic search in Holy Quran means that the verses that are relevant to a certain topic should be retrieved on querying the Quranic text by putting a query word or any synonym of it either the query word is present in those verses or not."*

As discussed earlier that many software use keyword matching technique to retrieve the query results. In many cases all the retrieved verses are not relevant to the given query word.

Further more, some relevant verses that should be retrieved against the query-word are not retrieved as the particular query-word is not present in those verses. There are hundreds of examples of this kind in the Holy Quran. This requires that some techniques must be used to retrieve the required/related verses from the Holy Quran either the query-word is present in them or not. Similarly irrelevant verses must not be retrieved though the query-word is present in them.

## 1.4 The Roadmap

Initially, the exploration of the Holy Quran has been done and the issues and challenges may come up for doing semantic search. The properties of Holy Quran as document for semantic search have been declared. Then thorough survey of the Semantic search and semantic web has been carried out. It encompasses all about the semantic web building block, semantic languages, semantic techniques, ontology development tools etc. Comparative analysis of all w.r.t our goal of semantic search in Holy Quran has been provided. This exploratory research in field of semantic web helped to provide recommendations not only for Semantic search in sample ontology of animal but also for the all likely domains of the Holy Quran.

For ontology construction, definition of concepts/classes and their instances, relationship among concepts in the form of object properties, the definition of data properties contained by instances have been done as discussed in [13,21] .All these have been done by conceptually modeling by creating directed graph. Eventually, it provided foundation for building sample ontology. Ontology provides the facility of Semantic search in web pages thus the same approach for semantic search has been applied to document of Holy Quran.

Ontologies are created on basis of domains, and then are integrated. [22] Ontologies give definition among them these definition provide foundation for semantic search, inference. Ultimately, it will lay foundation for covering complete work on Semantic search in Holy Quran.

## 1.5 Objectives

The major objectives of our research include:

- To accomplish Semantic search in Holy Quran.
- To explore the Holy Quran for identification of issues and challenges for semantic search.
- To explore the Holy Quran for different categories of subjects, development of classification domains and taxonomy building.
- To explore semantic web domain and recommendations about selection of best suited basic building block, language and tool for Holy Quran.
- Selection of domain as best suitable as base of sample ontology and then implant ontology for that domain to validate recommendations after exploratory survey based research

## 1.6 Scope of the Research

The Significance of our work is extremely important as Holy Quran has Divine knowledge of extreme importance. One of the reasons to implement ontology is that it is not only understandable to humans but also to machines. We can infer many things/concepts from ontologies.

The model will help the Quranic software developers too. Although this model, with some necessary changes, will be applicable to other holy books and law books, yet our focus is only the Holy Quran. After successful implementation for the Holy Quran, this model can be implemented for Hadith books and Fiqh, with some changes of courses.

The following few points is glimpse to cover the scope of the research work, actually intuitive in nature:

- To facilitate the programmers who want to develop Quranic software.
- To provide semantic search facility to users who are extreme subject conscious.
- To facilitate the religious scholars to prepare their topic based Quranic index.
- To motivate IT researchers to develop semantic/intelligent search engine for the Holy Quran and other Islamic literature books.

There is great necessity that a large project on government level should be introduced for Definition of Islamic WordNet based on WordNet [23].This Islamic WordNet should be based on Definition of concepts based on Holy Quran and authentic Hadith books. Then the semantic search in all domains may be done on the base of that WordNet.

## 1.7 Thesis Outline

Let us discuss how the remainder of this thesis documentation has been organized. Chapter number 2, titled literature survey, discusses the prior work done regarding our area is highlighted. In chapter 3 we review the requirement analysis of our proposed model. Chapter 4 discusses design model of the proposed approach. Chapter 5 describes the implementation issues and details. Chapter 6 shows the results taken from the software designed on the base of our proposed model. The last chapter describes the conclusion as well as future work in this field of study. Lastly, appendices are given having List of Entities, classes, instances and properties, The Fourteen Acts and Inferences by Roger Schank, Sample questions to be answered, List of Complete Triplets, RDF/OWL Protégé Code .

# CHAPTER 2

# LITERATURE SURVEY

# CHAPTER 2

# LITERATURE SURVEY

And not an atom's weight in the earth or in the sky escapeth your Lord, nor what is less than that or greater than that, but it is (written) in a clear Book.

(Surah: Yunas: 10)

---

Firstly, it is notable that as the work is of diverse in nature thus the work done in this domain has been divided three main sub-topics. Firstly, the discussion of different digital storage sources and work has done about XML and OSIS which may relate to our proposed work. Secondly, the brief discussion of applications available provides search facility in Holy Quran, thirdly, the discussion of some research papers that have some what relation to our work. Lastly, an extensive discussion has been presented to show the sequence of work done in field if conceptual/semantic search.

Many people have worked on the Holy Quran in different areas. A lot of work has been done in the field of storing the Holy Quran in different format by using different techniques such as.

- Relations of the database.
- Inverted File Structure.
- XML format.
- Unicode

Storing any text and its translations in different attributes of the database relation(s) is very common and popular technique [24]. It is easy to develop and handle the database by providing different constraints and making changes transparent to the users. The Relational

Database Management Systems are well matured today- this is another reason of the popularity of using the relational model. We will also prefer to use relational model for the digital storage of the Holy Quran.

The Inverted File Structure technique for storing the Holy Quran is also used. The Holy Quran text in Malay translations has been stored in an inverted file structure. Malay translated Al- Quran documents are based on user natural query words [25].

XML provides format independent platform to the developers of web applications. It also provides its own semantics for the text. The user can query in any format and get results in the same format in which s/he puts the query [26].

Quranic text is also stored in Unicode [27]. Here we are not concerned with the storage techniques or storage media.

The Arabic text database management system was designed to handle the Arabic databases. English language based database management systems are not so much useful and efficient for Arabic text [28].

## 2.1 Related Research

So for as our topic "semantic search in Quran" is concerned we could not find any relevant material. Much work has been done in the fields of semantic web and text mining. Our work is actually combination of many fields, like, NLP, Machine Learning, Text Mining, Semantic Web, WordNet Ontologies etc. We searched the world-wide web and found some papers which show some work on Quran. The following is the survey of the literature we did in our research.

### 2.1.1 XML Semantics Rules for Holy Books

The SLXS Specification Language, a new specification language for XML, is used for XML semantics and consistency checking. It is employed to specify semantics rules. The writers applied XML semantics approach to specify the consistency of the Holy books that are published in XML format. The goal of the study was to understand the appliance of XML

semantics in a filed of research that is appropriated for the method, and to find out a few of the strong point of XML semantics in this domain. These semantic rules written in XML format, to verify the unusual semantic consistency troubles that exist inside the Holy books [29].

### 2.1.2 XML Semantics Checker Model for Quran

The XML Semantics Checker Model, a novel approach was used to check the semantic consistencies of the Holy Quran, mentioned in Religion 2.00 website. It is managed by group of four religious works having source of many research work. This checker model was successfully applied to check that the number of verses in each chapter was correctly written in the Quran XML format document. It was also notable that Quranic XML document consists of 30 chapters too [30].

### 2.1.3 Morphological Analysis of the Holy Quran

A morphological analysis of Holy Quran and annotation proposed research has been carried out. By developing a tool on the base of finite state automata they apply the rules to the Quranic text and get morphological analysis of Quranic words. This system is currently being used by researchers and teachers for research and teaching purposes [31].

### 2.1.4 Keyphrase Extraction for Islamic Knowledge Ontology

Very initial and basic level taxonomical work for extraction key word and key phrase candidate for developing ontology of Islamic literature has been carried out. Key phrases provide semantic metadata. Collection extraction also known as automatic keyword extraction is to select key words from a particular text. This paper also describes the algorithm for automatic extraction of the key words. Lexico-syntactic and statistical methods have been used for this purpose [32].

### 2.1.5 Build Islamic Ontology based on Ontology Learning

Saidah saad and Naomie Salim have presented a general and skelatel methodology and life cycle for building the ontology for Islamic literature in their paper titled as "Build Islamic Ontology based on Ontology Learning" [33]. Their research was inspired by software

engineering V-process model. They also applied their technique on English text for mining ontologies from natural language.

### 2.1.6 OSIS

The OSIS (Open Scripture Information Standard) model based on XML was specially designed for the Bible. It was developed by the Bible Technologies Group. It defines tags for marking up scripture like that of Bibles', and other related literature. Its current version is 2.1. It is the first project to come out of the Bible Technologies Group [26].

OSIS provides initiative to identify and develop the organizational process, and technical basis that produce markup standards for the Bible and relevant materials. Its goal is to provide open standards for publishers, software manufacturers, Bible Societies, scholars, and a reader of the Bible. OSIS also provides the facility for easy storage and retrieval of text. It also handles indexing and cross-referencing of raw text. The out put can be seen in many formats [26].

#### 2.1.6.1 Why was XML chosen to be the foundation of the OSIS standard?

XML has several important aspects.

**1.** It facilitates to describe the structure of documents. A file can directly feed Web delivery to be produced in multiple formats. Additionally, it can also be used for non-reading formats like indexing and linguistic analysis.

2. XML's predecessor SGML [26] was suitable for very large scale projects like aircraft manuals, policy documents and collections of literary and scholarly texts. It was good for large-scale projects but poor for smaller ones. XML is a tiny subset of SGML. It is easier to learn, use, and implement.

**3.** It is an open standard and it independent of some specific software [26].

In the nutshell, OSIS is perfect for storing the Quran and all translations in. It is originally developed for the Bible but it can be used for the Quran too [34].

## 2.2 Present Quranic Software

There are a large number of Quranic databases in digital form. One can also get variety of Quranic softwares present in the market as well as most of them are available on internet. A large number of these digital Quranic stores are in the form of audio [35] and video files. As for video is concerned there is only the Arabic text with a number of translations in different natural languages [36].

Some of the Quranic Softwares facilitate the reader to query the Arabic text for exact match on the basis of a single word/term or part of the verse or even the whole verse. These softwares find the exact matches and display the results. Normally query is put in Arabic or, in some softwares it is possible to query the database using English alphabets for Arabic terms (e.g. Muttaqeen for "متّقين", it is an Arabic term written in English alphabets meaning, the pious). To query in other natural languages is also possible in some softwares.

We found a software that is a product of the "Harf Company" [37]. It is the best software for Quran, so for as we know. In this software subjective search facility is provided. If someone wants to search a topic or even subtopic in the Quran, he/she can do so by just clicking the name of the topic/subtopic and the relevant verses are retrieved. The main topics are organized alphabetically and the subtopics are also organized under the main topics in the same way. This software also provides the exact match search for words/terms, part of the verse and even some consecutive verses. This is very good software as we have mentioned above but it also has some deficiencies:

- The whole software is in Arabic language, thus only those can use it that are well familiar with Arabic language. For Arabic speakers it is very good.
- Technically speaking, this software is bounded to predefined topics and subtopics.
- Further more, the verses related to any topic/subtopic are predefined. In other words, the data (verse/s) to be retrieved against any topic are pre-clustered.
- The user can not change anything. In other words, the user can not use his opinion for any verse.

- The user can not re-arrange the sequence of retrieved verses.

A brief introduction of some softwares is given below:

We find Quranic software that aims to help professional Muslim users such as religious researchers and students. It is also helps lawyers, judges, book authors, and lecturers, to work with Quran. This software contains hierarchal view of all verses in Quran, either sorted by Surah, or by parts [38].

Multi-Language Quran Software provides Arabic/English Quranic transliterations. It is equipped with Quranic commentary, index, and glossary of more than 500 words. It also provides full search on the key word bases. It supports plug-in Quranic translations [39]. And more diverse content can be found for learning purpose from the designed Holy Quran Recitation reference tool [40].

High quality Quran website provides more than 30 translations of the Quran, and more than 10 recitations. It also provides root word search. It takes the query word as input and after processing the morphological analysis of the query word it tells the root word of the query word as output [41]. The [42] is a professional resource of the Quranic literature. It provides the key word base searchable interface, indexed by Surah number.

Quran browser provides recitations by different well known reciters (Qaries) and a number of translations in different languages [43].

A practical resource for understanding the Quran is accessible. It also includes an effective course for non-Arab to comprehend Arabic abilities while listening to and reading the Arabic text. It also provides introduction and guide to the Quran and dictionaries for children [44]. The [45] facilitates the users to browse Quran and search with translation and Tafseer (interpretation). One can also use same recitation and display engine as qurany.net's IslamKit [46]. The [47] provides audio and video Islamic resources. It provides original search functionality.

"QuraanicLessons" is a Quran Learning Software for English speaking Muslims. It is an easy way to learn the Quranic language for those who are not good in understanding Arabic

language although they can read it. Many simple lessons have been provided to build Arabic vocabulary for learners [48]. English translation of Holy Quran in Malayalam can be used in two different languages, Malayalam and English. Default interface language is English. User can easily switch between the languages [49].

Multilingual Quran Software offered by "topshareware" website provides Arabic and English Quranic Commentary. Different translations in Urdu, Indonesian, Turkish, German, Spanish, French, Malay, Japanese, Tamil and Hausa are also provided by this software. It is equipped with query word base searching facility [50].

In spite of having significant number of sources containing Quranic software and Quranic literature yet we could not find much literature about the searching techniques. By going through different software we perceive that most of them use query word base searching technique. We could not find even a single thesis or paper about semantic search in Quran.

## 2.3 Semantic Search

Ontology is a new phenomenon but has been used for semantic search. The brief literature review of the work is as follows:

A system using new algorithm of information extraction based on ontology has been proposed in [51]. By using semantic reasoning based on ontology, it improves the search engine to understand user's query purpose. A prototype of search engine is developed and the search results are better than those of common search engine.

In proposed framework, search request is first processed by a query parser. After this, it looks for RDF triples in the domain ontologies. Web documents related to the required concepts and individuals specified are retrieved by a document retriever. Resultantly, the retrieved documents are ranked according to their relativity to the query. An extended term-document matrix is developed to depict the relation among concepts, documents, objects, and terms. [52]

SemPub [53] emphasizes on semantic retrieval and user customization. In semantic search, search for terms in a concept space is carried out. Concept space is a graph of terms

occurring in documents associated by the frequency and relationships between them. Thus by optimizing complex object space into a more manageable automatically-generated, more meaningful concept space, customized information retrieval is made easy.

The document-centric query answering, a novel form of query answering for the Semantic Web has been presented [54]. The knowledge base system describes the key techniques used in the software to resolve scalability issues.

The knowledge acquisition system is proposed that dynamically constructs query-based ontology to provide answers for users' queries [55]. For construction of the relationships and hierarchy of concepts in ontology, Formal Concept Analysis (FCA) approach is used. Then system infers the answer without asking users to read all the documents.

[56] Focuses on search advances in image and video search mainly on large-scale semantic concept finding as well as indexing. The purpose of such semantic indexing paradigm has been achieved driven by enhancing availability of the big resources of corpora, original labeling methods, innovative image features, and machine learning techniques for visual content recognition.

The UML sequence diagram is implemented as a tool of general solving problem method description with respect to semantic search. The general structure of semantic search supporting is developed and some principles of followed ontology method formalization have been proposed. [57]

The RDF allows representation and management of all sort of semantic information about multimedia resources. It suggests a likely approach, based on a structure called RDF Descriptor, which allows representing, reconciling and semantically tagging multimedia resources of various media formats, and possibly coming from diverse sources with various unlike representations. [58]

It proposes a new concept learning approach to resolve the issue of overcoming semantic heterogeneity among diverse ontologies and also implements it using multiagent technology together with the IBM's UIMA (Unstructured Information Management Architecture) into a semantic search application [59]. The agent in one system can learn a concept which is

unfamiliar to it by seeking guidance from agents in other systems and integrate the learnt concept into its local ontology. [59]

In this paper, a survey over the primary literature regarding semantic search technologies has been made. By classifying the literature into six main groups, let us briefly discuss their characteristics. Additionally, the issues within the reviewed semantic search methods and engines are analyzed and concluded based on four perspectives. [60]

An ontology-based framework for semantic expansion search is proposed. Based on constructed domain ontology, two algorithms are presented: one for semantic annotation algorithm and another is semantic expansion reasoning algorithm. Also a semantic search prototype system is planned and put into practice. The investigational results show that proposed methodology achieves higher recall ratio and precision ratio.[61]

Word games provide an interesting opportunity for semantic knowledge acquisition that may be used to construct semantic memory. Architecture of the knowledge base inspired by psycholinguistic hypothesis of human cognition process is proposed. The core of the system is an algorithm by means of a vector representation of concepts for semantic search. Based on this algorithm a 20 questions game has been implemented.[62]

Using the RDF, a calculation of semantic similarity among concepts of ontology is represented by taxonomy of ontology concepts. To enhance the calculation of semantic similarity among concepts up to level to make it nearly similar to human's subjective judgment, the similarity be calculated by form of the word used. This paper represents the steps which use semantic similarity in semantic search executing [63].

After analyze the limitations of traditional method of search, bring a semantic search forward, and mainly introduce the thought of the semantic retrieval as well as the way to constitute ontology entity and the language that describes it. Moreover, semantic retrieval system based on ontology is also given.[64]

[65] Present a demo of ESTER, a search engine that combines the easiness of utilize, pace and scalability of full-text search with the powerful semantic capabilities of ontologies. ESTER supports text queries, ontological queries as well as combinations of the two. ESTER

works by reducing queries to two fundamental operations, which are prefix search and join. These two operations can be implemented very efficiently in terms of both processing time and index spaces.

As semantic search methods have proven to work satisfactory in specific domains, they still have to confront two main challenges to scale up to the Web in its entirety. In this work, this issue has been addressed with a new semantic search system that provides the user with the capability to query Semantic Web information using natural language, by means of an ontology-based Question Answering and complements the specific answers retrieved during the QA process with a ranked list of documents from the Web. The results show that ontology-based semantic search capabilities can be used to complement and enhance keyword search technologies [66].

Eliminating irrelevant Web pages, the time-consuming task of revise the obtained results from actual search engines is reduced. The proposed approach is focused on Web pages that are not defined with semantic Web structure (most of the actual Web pages are in this format). The challenge is extract the semantic content from heterogeneous and human oriented Web pages. The approach integrates structures of ontologies, WordNet, and a hierarchical similarity measure to determine the relevance of a Web page.[67]

## 2.4 Summary

What ever the work have been done for the Holy Quran or other Islamic literature is multi-dimensional. How to perform semantic or intelligent search in Holy Quran is not taken as research topic. We are the pioneer in this field to perform the semantic search in the Holy Quran. Work on morphological or syntactic analysis of the Holy Quran has been done but no one, to our best knowledge, tried for semantic search.

# CHAPTER No. 3

# PROBLEM ANALYSIS

# CHAPTER No. 3

# PROBLEM ANALYSIS

Say: "The (Qur'an) was sent down by Him who knows the mystery (that is) in the heavens and the earth: verily He is Oft-Forgiving, Most Merciful."
(Surah – 25:6)

---

In this chapter we give analyze the problems that are faced if we want to apply semantic search techniques on Holy Quran as a document. We face some specific problems in the Holy Quran due to unique style Quranic text.

## 3.1 Holy Quran's Unique Style

Quran has its own style of describing the topics. At some places, some topics are explicitly mentioned while some others are meant implicitly. This implicit nature of the Holy Quran is a great challenge for the NLP and AI experts. Here the context very often helps the scholars to find the implicit meanings but some times it does not provide any clue.

Some times there is dissimilarity between perceptible and actual meanings. For example, in Sura Al-Mo'menoon, Allah says in Holy Quran, "They avoid vain talk". Now, in this verse, the word "لغو" has been used. Its apparent meanings are vain talk according to some interpreters it is meant for music.

### 3.1.1 The allegorical nature of the Holy Quran

The allegorical nature of the Holy Quran is also a challenge not only for NLP experts but also for the Quranic scholars. Different scholars explain allegorical terms according to their own perception/sect/knowledge etc. The Holy Quran comments itself about this issue in the following verses:

> *"He it is Who has sent down to thee the Book: In it are verses basic or fundamental (of established meaning); they are the foundation of the Book: others are allegorical. But those in whose hearts is perversity follow the part thereof that is allegorical, seeking discord, and searching for its hidden meanings, but no one knows its hidden meanings except Allah. And those who are firmly grounded in knowledge say: "We believe in the Book; the whole of it is from our Lord:" and none will grasp the Message except men of understanding[2]". (3:7)*

Allah has used the word Al-jannah for paradise. Jannah means the gardens. Is the paradise exactly same as our worldly gardens? No. The word Al-jannah is used for paradise because human can not realize the things which they have not observed with eyes. In fact, the paradise does not resemble any of worldly things. So if any strange words were used for paradise the human could not have any sketch/idea about the paradise.

### 3.1.2 Scattered Discussion of Topics

The Holy Quran has unique style of explaining different topics. Normally a topic is discussed at different places. For example, the topic of Hazrat Moosa (AS) is discussed in a large number of chapters; the Oneness of Allah has been discussed through out the Holy Quran. It is not necessary that the consecutive verses must belong to the same topic. A topic may or may not be discussed in consecutive verses. Some consecutive verses or even a single verse may contain many topics. For example, the second verse of Surah Al-Baqrah, consisting of only seven words in Arabic text including prepositions, mentions three topics.

*"This is the Book; in it is guidance sure, without doubt, to those who fear God"* (2: 2)

This is a little verse and discusses three topics---The Book (Holy Quran), guidance and the pious. Lengthy verses may contain dozens of topics. Normally a chapter does not contain only one topic and the same is true for a lot of verses.

### 3.1.3 The Name of a Chapter and their Text

The name of a chapter does not mean that all the text of this chapter belongs to this topic. Keeping in view this property we can divide the chapters of the Holy Quran in four categories.

- There are some chapters (for example Surah Ikhlas) whose whole text is related to their names. These chapters are small in number and size.
- There are some other chapters whose major topics are related to their names. They also contain many other topics. They are also few in numbers. Surah Yousaf is an example of this type. Its major topic is narrative of what happened to Hazrat Yousaf (AS). From start to end this story has been discussed in fantastic style. Beside this story many other topics are discussed in both ways: explicitly and implicitly.
- In third category, the name of Surah is relevant to its some verses or words (may be only one word) while other verses are meant for other topics. For example, Surah Al-Baqrah---The Cow (containing 286 verses) discusses the topic of "the Cow---Bani Israel's Cow" just in few verses; and this Surah discusses hundreds of other topics. Majority of the Surahs belong to this category.
- For some cases the starting word of a chapter is the name of the Surah. For example, Surah Yaseen, Surah Haameem. These words, called Hoaroof-e-Muqateat, are not understandable to us and thus all the other verses are not linked to the chapter names at all.

It is apparent that we cannot divide the Book into chapters or Surahs for application of our semantic search. In the fields of Information Retrieval there are many techniques but these are suitable for structured data but in our case of un-structured data, our problem domain leads us to the use of domain ontologies as a mechanism to accomplish the task.

## 3.2 Information Retrieval or Extraction from the Holy Quran

What type of information should be extracted from the Holy Quran? The complete answer of this question is perhaps difficult to explain. Most of the users want to perform the subject based search in the Holy Quran for a particular topic. A user wants either to know the basic

concept or the detail of a topic. He/she may have such type of questions in his/her mind about a query-word:

- What is meant for the key word? Where is it found in Holy Quran?
- How the concept has been defined by Allah Almighty in Holy Quran?
- What are different commands of Allah about it?
- How many times, it has been defined in Holy Quran. It usually depicts its importance in Islam?
- What are its related concepts? If it is an event, who causes it to occur?
- What is the category of the concept, if any? Or in what context it has been defines?
- What is its benefit/reward or loss/punishment in this world or in the world Hereafter?
- Which verses (strongly or weakly) belong to this topic?

It is notable that the last question is relatively easy to answer and it is the subject matter of Information Retrieval. These questions need NLP and AI techniques and WordNet ontologies. The Quranic text is written neither in story-writing or essay-writing style nor in question-answer style. A system that is capable of answering the questions like pointed out above needs special attention, cooperation and team work of the experts from different areas.

## 3.3 Main Problem of Irrelevant Verse Retrieval

There are two major problems with key-word based searching technique:

- First, many relevant verses are not retrieved as the particular query word is not present in them.

- Second, more curious is that some irrelevant verses may be retrieved as the query word is present in them. .

**Examples:**

- Verse numbers 3-6 of surah Al-Baqra are concerned with "Muttaqeen-متّقين", but this word is not present in all these three verses. This word is present at the end verse # 2. So we can say that the context tells us that these verses belong to "Muttaqeen"

- Surah Al-Baqra, verse numbers 8-18 are concerned with hypocrites. Neither the word "hypocrites" nor any synonym of it has been used in these verses. This word is also not present in the context of these verses.

- The word "believers" has been used in verse # 9 of Surah Al-Baqarah but it does not belong to the believers instead it belongs to the hypocrites.

Likewise, there are numerous examples of this kind in the Holy Quran. This requires that some techniques must be used to retrieve the required/related verses from the Holy Quran either the query-word is present in them or not. Similarly irrelevant verses must not be retrieved though the query-word is present in them. Above cases motivate us to use semantic search technique in the Holy Quran.

## 3.4 Problem Domain

Quranic text is written in Arabic, a natural language. As other texts of natural languages create problems for computer to process the text data, the Quranic text also does the same. Arabic language is different from other languages in many ways, so some additional problems of the Arabic text are also faced by the NLP specialists. Quranic text is unique in its style. It needs special attention of the scholars from different fields. The combined efforts of Quranic scholars and the specialists of Computational Linguistics, Artificial Intelligence, Text Mining, Semantic Web, WordNet Ontologies and Machine Leaning are needed. The Quranic text is very complex and sensitive as it is the miracle from Almighty Allah. The Information retrieval and information extraction are very difficult to handle. Query word matching technique does not satisfy the researcher of the Holy Quran. A large number of relevant verses may not be retrieved and some irrelevant verses may be retrieved. Thus, in such scenario, the use of conventional keyword is totally un-efficient method for search of verses related to required concept. So the design of database can not be an option to solve such problem. The database does not provide us flexibility of hierarchy and taxonomy in its true spirit. These are also not good for documents where natural language processing is involved and also tagging can not be option as parser [14]. So there is requirement of application of some methods and techniques to be adopted to overcome this problem.

### 3.4.1 Problem Statement

Semantic search for the Holy Quran is becoming more vital day by day. Especially with the introduction of semantic web, we must plan to achieve semantic search in the Holy Quran. So for as it is concerned we have following questions in our mind.

- What do we mean by semantic search in Holy Quran?
- Does the semantic search for Holy Quran exist?
- Is semantic search for Holy Quran possible?
- Will semantic search for Holy Quran satisfy the readers of the Holy Quran?
- How can we overcome the problems faced by users of the Holy Quran in key word base searching?

*"Semantic search in Holy Quran means that the verses that are relevant to a certain topic should be retrieved on querying the Quranic text by putting a query word or any synonym of it either the query word is present in those verses or not."*

## 3.5 The Obstacles in Developing an Algorithm for Semantic Search in Quran

None of the software provides semantic/intelligent search. Why? There are so many problems to develop intelligent search engine for the Holy Quran. e.g.

- Quran has its own versatile sequence of text - different from human literature.
- Change of topic is very frequent from verse to verse and even within a verse.
- A topic is discussed in many consecutive and non-consecutive verses within a chapter and even in different chapters as well; and a verse may contain many topics.
- Most of the verses in a chapter are not relevant to the chapter name.
- Arabic words have so many forms. Prefix and suffix are also frequently used.
- A word has so many meanings--- polysemy, thus different translators have used it differently.
- A sense may be expressed by so many words--- synonymy, similarly there should be some dynamic mechanism that more words can be added into the basic framework.
- There is significant number of allegorical (Mutshabehaat) verses.

- There are a large number of allegorical words.
- Haroof-e-Muqateaat.
- Many references in Holy Quran have been given in contextual way. Thus the definition of concepts must also be dynamic enough that concepts specific to some context can also be retrieved. Like in the domain of animals, the non-believers have been stated like donkey carrying books.

Different scholars/sects have different opinions for a verse. Even, for some verses, they have opposite opinions.

In the light of above discussion, intuitively, there should way to develop such a structure that first domains can be identified then taxonomy and classification of the subjects of Holy Quran can be carried out. Then the concepts can be defined and relationship among concepts can be defined, if any. Lastly, there should be mechanism to achieve inference from the new structure.

## 3.6 The Planned Solution

The proposed solutions has been briefly discusses by defining the concept of semantic search, the application of semantic search in Holy Quran then objectives of the research have been elaborated.

### 3.6.1 The Aim

The aim of the proposed research to do literature survey in the field of semantic web and provide such as

**How to accomplish the task of semantic search in Holy Quran?**

This question can be answered after we can successfully respond to following questions.

a) What are the problems, challenges and issues in unstructured documents such as Holy Quran?

b) What are different categories, classification and taxonomy of subject matter in Holy Quran and discussion on various with respect to semantic search?

c) What are the different building blocks, languages, ontology development tools etc for Holy Quran and which one is best suited for Holy Quran?

d) Which domain, among many, is best suitable as base of sample ontology and how to evaluate the results and also to validate recommendations after exploratory survey based research?

## 3.7 Summary

Current Quranic software provide searching on key word matching base. For the Holy Quran this technique does not give satisfactory results. A large number of relevant verses are not retrieved against a query word and some irrelevant verses may be retrieved. To resolve such issue, let us present a technique that will provide the satisfactory results regarding semantic search.

# CHAPTER NO. 4

# Exploratory Research

# CHAPTER NO. 4

# Exploratory Research

Surely We Ourselves have revealed the Quran to you revealing (it) in portions.

(Sura Al-Insan:23)

---

## 4.1 Concept of Semantic

Semantics is the study of meaning. The word "semantics" itself depicts an array of ideas, from the general perspective to the highly technical. It is often used in common verbal communication to represent a dilemma of understanding that comes down to word selection or connotation. [15]

The word semantics is derived from the Greek word semantikos meaning significant [16] from semaino, meaning to indicate or to signify and that from sema means sign [17].In linguistics, it is the study of understanding of symbols or signs used by persons or communities in some specific contexts and situations [18]. In linguistics, semantic analysis is the procedure of connecting syntactic arrangement, from the levels of phrases to the clauses and from sentences to complete text [19].

## 4.2 Semantic Search

Semantic search is a process used to enhance level of searching by the use of data from semantic networks to remove ambiguity from queries and web text in order to generate more relevant results. Semantic Search uses semantics, or the science of meaning in language to create vastly related search results. In the majority cases, the goal is to deliver the information queried by user instead of have a user sort through a list of loosely related keyword results [20].

Semantic search is usually referred as an application of the Semantic Web to search. Search is both one of the most significant applications on the Web and an application with considerable room for improvement. It is believed that the addition of explicit semantics may result in improvement of overall search. It attempts to expand the approach and improve traditional search results which were based on conventional information retrieval style and involves semantic into it for subjective search [68].

## 4.3 Semantic Web

The concept of semantic search based on ontology is due to semantic web, not new web but an extension of current web.

### 4.3.1 Problem of Current Web

The increasing popularity of the World Wide Web (WWW) has altered the manner we take our computers. Since the birth of world wide web by CERN laboratories in 1989, the level of communication has changed its shape and internet jumped to global information space that is made up of millions of millions of Web pages. This huge number of We pages ensures that information is available on almost every topic from every sphere of life but problem lies as it makes difficult for user to find required and relevant information only. The prominent search engines like Google and Yahoo aim to help out users when searching the Web for information. Since the content of the Web pages is presented in human language, search engines have no access to the semantics of the content. This lessens the potential of search engines mainly to keyword search. Computers can hardly support the user with this problem since machines do not have access to the semantics of the Web pages.[69].

### 4.3.2 Concept of Semantic Web

Semantic Web is an interlock of information linked up in such a manner so that it may be easily process able by machines. We can imagine of it as an efficient way of representing data on the World Wide Web, or as a globally linked semantically built database.

Semantic Web was idea by Tim Berners-Lee who is also inventor of the WWW, URIs, HTTP, and HTML. The World Wide Web consortium (W3C) is working hard to perk up the

field as many research works in shape of publications, projects and tools etc have already been developed [70]. The main issue regarding use of data on the Web is that it is not available in such a standard way that it can be processed globally. For example, just think of information about national events, prayer timing, weather information, plane times, Major League Baseball statistics, and television guides... all of this information is presented by numerous sites, but all in HTML. The problem with that is that, is some contexts, it is difficult to use this data in the ways that is required by the user. So the Semantic Web can be seen as an enormous result.

Tim Berners-Lee, father of Semantic Web, defines Semantic Web as follows:

"*The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.*" [71].

Following this definition, this extension consists of meta-data describing the semantics of the Web pages in a machine-processable way. Before the Web pages can be described with semantic meta-data, ontology has to be defined for the domain of discourse.[69]

### 4.3.3 Semantic Web Present and Future

It is significant to note that accomplishment of RDF, OWL, and the Semantic Web as a whole will be a steady procedure. The queries like what the Semantic Web is and how it can help business and folks are similar to initial confusion about why we needed HTTP and the web before world world web. But considering how those technologies have flourished, it is probable that the vision of scope of semantic web will be realized.

The true impact of the new field will be known soon due to its staggering potential. Semantic Web will lead to the evolution of human knowledge itself by allowing users to rapidly sort out the gigantic amounts of data that exist in the world in a relevant, productive way. It is also significant to make a note that, analogous to current Web services, the Semantic Web may primarily be limited to intranet and extranet applications until questions about information security can be sufficiently addressed [72].

## 4.4 Ontology

The term ontology has its origin from Greek language. It is combination of two words 'onto' (being), and 'logia' ('written or spoken discourse). It provides a definitive and exhaustive classification of entities in all spheres of being. Computer scientists give a new interpretation of ontology as a specification of a conceptualization [73]. In computer science, explicit ontology is used to facilitate knowledge reuse, as opposite of implicit ontology which is knowledge encoded into software systems [74].

### 4.4.1 Ontology vs Conceptual Modeling

There is frequent confusion between ontology and conceptual models. In few cases, ontology has analogous purpose to that of database schema because it provides meta-information with the purpose of definition of the semantics of the concepts in the data, but there are several important differences between the two concepts:

- Languages for defining and representing ontologies (OWL, etc.) are syntactically and semantically richer than common approaches for databases (SQL, etc.) as far as syntactic and semantic capability is concerned.
- The knowledge that is described by ontology consists of semi structured information as compared to the well-structured data of the database.
- Ontology is referred as shared conceptualization because it is used for information sharing and exchange. Identifiers in a database schema are used for a concrete system and need not attempt to attain the equivalent of ontological agreements.
- Ontology gives us the concepts of a domain theory and does not provide us the structure of a data container [75].

## 4.5 Features of Ontology

The following three features are basic requirement that must be present in ontology:

**Extensibility.** Ontology engineers should be able to develop ontologies in an incremental manner by reusing as many existing popular concepts as possible before creating a new concept from scratch. For example, the concept "man" can be defined as a *sub-class* of an existing concept "person" in WordNet vocabulary. This requirement demands an expressive common reference model as well as distributed symbol resolution mechanisms.

**Visibility.** In order to make knowledge visible on the Web, additional common ontological ground on syntax and semantics is required between information consumers and publishers. This condition is particularly significant to machines since they are not capable of understanding information written in a different language.

**Ability of Inference.** Ontology not only serves the purpose of representation, i.e. enumerating factual domain knowledge, but also serves the purpose of computation, i.e., enabling logical inference on facts. Ontology should provide constructs for effective binding with logical inference primitives and choice to meet a range of expressiveness and computational complexity requirements [74].

## 4.6 Categorization of Ontology

The classification of ontology can be carried out on the basis of internal structure and on the basis of subject of conceptualization.

### 5.6.1 Categorization on the basis of internal structure

An ontology belongs to one of the following categories, based on the level of richness of its internal structure:

- ***Controlled vocabularies*:** Formed by a finite list of terms.
- ***Glossaries:*** Lists of terms with their definitions offered in natural language.
- ***Thesauruses:*** Differentiated from the previous categories in that they offer semantic additions to the terms, including synonyms.
- ***Informal hierarchies:*** Hierarchies of terms which do not correspond to a strict subclass. For example, the terms "rental vehicle" and "hotel" could be modeled informally under the hierarchy "travel" as they are considered key parts of traveling.
- ***Formal hierarchies:*** In this case, a strict "is-a" relationship exists between instances of a class and of its corresponding super class. For example, a teacher "is-a" people. Its objective is to exploit the inheritance concept.

• ***Frames:*** Ontologies which include such classes as properties, which can be inherited by other classes in lower levels of a formal "is-a" taxonomy.

• ***Ontologies with value constraints***: Include value constraints. The most typical case is that of constraints dependent on the data type of a property (for example, a day of the month must be lower than 32).

• ***Ontologies with generic logical constraints:*** These are the most expressive ontologies which permit specific constraints between the terms of the ontology [75] .

It is evident that the proposed domain ontologies for Holy Quran will lie in the last two types among above mentioned.

### 4.6.2 Categorization of ontologies based on Conceptualization

Simultaneously, depending on the subject of the conceptualization, an ontology falls into one of the following types:

• ***Knowledge representation ontologies***: Capture representation primitives used to formalize knowledge under a concrete paradigm of knowledge representation.

• ***Common or generic ontologies:*** Represent common-sense knowledge reusable in distinct domains, for example, vocabulary related to things, events, time, space, etc.

• ***High-level ontologies:*** Describe very common concepts which can be correlated to root terms of all ontologies. An unresolved problem is that many of these high-level ontologies differ in their way of categorizing common concepts. So it results in making it difficult to integrate and exchange ontologies.

• ***Domain ontologies*:** These are ontologies about covering concepts of a specific domain (for example, medicine, engineering, etc.). Its main advantage is reusability.

• ***Task ontologies*:** These illustrate the vocabulary related to some basic activity. They provide a systematic vocabulary of terms used to solve problems. These can be, but not necessarily, related to domain.

• ***Domain task ontologies***: These are of different type as these are reusable in a certain domain, but reusability is not applicable to different domains.

• ***Method ontologies***: Provide definitions of relevant concepts and their relationships. They are applicable to a reasoning process specifically designed to carry out a particular task.

• ***Application ontologies***: These are dependent on the applications. Often, they extend and concentrate on the vocabulary of one domain or task ontology for a particular application [75].

This categorization is interesting, as far as recommendations for semantic search in Holy Quran are concerned. We should Use domain ontologies for about all main domains present in the Holy Quran. All these ontologies will be merged but for merging, upper ontology, also known as high-level ontologies are mandatory to bring the understanding of different concept in diverse domain under common umbrella. Also, it is interesting that to build up a large application; application ontologies are also recommended.

## 4.7 Benefits of Domain Ontology

We can summarize the benefits of ontology as follows:

- Ontology provides a shared and definition and understanding about certain main concepts in the domain.
- It provides a mechanism to use/reuse domain knowledge.
- It formulates the domain assumptions explicit.
- Together with ontology description languages (such as OWL), it offers a way to encode knowledge and semantics such that machines can understand.
- It makes automatic large-scale machine processing possible [103].

## 4.8 Components of Ontology.

Ontology is a formal explicit description of concepts in a domain of interest, (classes), and properties of each concept describing various features and attributes of the concept (slots), and restrictions on slots (facets). Ontology together with a set of individual instances of

classes constitutes a knowledge base. It is confusion due to variety of different terms for same concept and it needs clarification. Classes are also known as concepts, properties are also known as slots and roles, restrictions on slots are called facets, and finally individuals are known as objects and instances too.

### 4.8.1 Class

Classes are the focus of most ontologies. Classes describe concepts in the domain. A class may represent a person, place, or thing - it is an abstraction of a concept. A class gives the description in such a way that it makes assumptions explicit by defining the properties, both object property and data properties, and by defining the relationship between different classes/concepts. A class can have subclasses that represent concepts that are more specific than the superclass. There is usually hierarchy of concepts to define the logical relationship between classes as parent class and its child classes and so on. This taxonomy helps to define the concepts in better way and step by step definition of complex concepts is made easy by creating its taxonomy. If a class C is a *subclass* of a class C', then all instances of C will also be instances of C'.

### 4.8.2 Individuals

The members of a class are known as *instances* of the class. It is also called object or individual. It owns the properties as defined in the class. It is related to other classes and objects on the basis of relationships among the classes w.r.t. its class.

### 4.8.3 Property

Property describes the characteristics of certain concept. It is also known as attribute. It has two basic types. Object property is used to create relation between two concepts. Data Property It is used to specify some data value to object. Like Age can be object property of class person. If we have some instance say named Umar then data property of Mr. Umar can be set by data property. There are a lot of data types available for setting of data property, and then age (in numbers) can be set some property

## 4.8.4 Types of Property

The following is brief introduction of the various sorts of properties:

#### 4.8.4.1 Symmetric Properties

A symmetric property describes the situation in which, if resource R1 is connected to resource R2 by property P, then resource R2 is also connected to resource R1 by the same property. For instance, we can define a property friend_with (for Person class), and if person A is friend_with person B, then person B is certainly friend_with person A.

#### 4.8.4.1 Transitive Properties

A transitive property describes the situation in which, if a resource R1 is connected to resource R2 by property P, and resource R2 is connected to resource R3 by the same property, then resource R1 is also connected to resource R3 by property P.

#### 4.8.4.3 Functional Properties

A functional property describes the situation in which, for any given instance, there is at most one value for that property. In other words, it defines a many-to-one situation: there is at most one unique value for each instance.

#### 4.8.4.4 Inverse Property

An inverse property describes the situation in which, if a resource R1 is connected to resource R2 by property P, then the inverse property of P will connect resource R2 to resource R1. A good example in our camera ontology is the property owned_by. Clearly, if a camera is owned_by a Photographer, then we can define an inverse property of owned_by, say, own, to indicate that the Photographer owns the camera.

#### 4.8.4.5 Inverse Functional Property

Recall the functional property. It states that for a given domain value, there is a unique range value. For instance, for a given camera, there is only one model value. An inverse functional property, as its name suggests, is just the opposite of functional property [104].

## 4.9 Ontology Language

### 4.9.1 What is ontology language

In computer science terminology, formal languages which are used to create ontologies are referred as ontology language. They allow the encoding of knowledge about particular domains and usually comprise of reasoning rules which support the processing of that knowledge [76]. Such languages are generally declarative languages and are commonly based on description logic.

### 4.9.2 Features of ontology Language

Ontology languages are expected to have following features:

**Conceptualization.** Ontology language should be based on some suitable reference model, for instance entity-relationship model, object-oriented model etc and should give related ontology structures to symbolize knowledge, such as defining the entities and relations in a domain.

**Vocabulary.** Not only the language should also cover semantics but also the syntax such as symbol assignment and grammars

**Axiomatization.** To understand the semantics for inference, along with knowledge, rules and constraints are essential. For example, we can use rules to generate new facts from existing knowledge, and to validate the consistency of knowledge [74].

### 4.9.3 Requirements of ontology languages

Ontology languages let us to write formal conceptualizations of domains models [77]. The main requirements are:

1. A well-defined syntax
2. A well-defined semantics
3. Efficient reasoning support
4. Sufficient expressive power
5. Convenience of expression.

### 4.9.4 Types of Ontology Language

#### 4.9.4.1 Resource Description Framework

The Resource Description Framework (RDF) [78] is the data-model for representing metadata in the Semantic Web. It is based on subject, predicate, object triples. These are called RDF statements. Its purpose is to formalize meta-data. RDF is domain independent in that no assumptions about a specific domain are made. It is up to the users to define their own ontologies for the user's domain in ontology language such as RDF-S.

***Subject:*** Subject is the thing about which we want to make a statement.

***Predicate:*** Predicate defines information we want to state about the subject.

***Object:*** The object defines the value of the predicate.

#### 4.9.4.2 RDF Schema

RDFS is ontology definition language to allow users to describe the vocabulary needed to define the resources in the domain with meta-data. To define the ontology RDFS uses the RDF triples format. Therefore, ontology in RDFS is modelled as RDF graph. In RDFS users can define all basic ontology parts such as concepts, relationships and characteristics etc [69].

#### 4.9.4.3 DAML+OIL

DAML is a ontology language, created by DARPA, based on RDF. DAML gets RDF-S further by giving us more comprehensive properties and classes. DAML is more expressive than with RDF-S and provides some simple terms for creating inferences [79]. Additive properties of DAML include properties such as inverses, unique properties, lists, data types, restrictions, cardinalities, disjoint lists etc.

#### 4.9.4.4 Web Ontology Language (OWL)

Web Ontology Language is the most commonly used ontology language full of its characteristics. It is planned for use by applications that need to process information content in spite of just presenting information to humans. It facilitates higher machine interpretability of content as compared to that supported by XML, RDF, and RDF-S, given that it provides additional vocabulary along with a formal semantics. OWL has three increasingly-expressive sublanguages: OWL Lite, OWL DL, and OWL Full.

#### 4.9.4.5 Types of OWL

Requirements for ontology language have prompted W3C's Web Ontology Working Group to define OWL as three different sublanguages, each of which is geared towards fulfilling different aspects of these incompatible full set of requirements:

- **OWL Lite:** It supports those needing a classification hierarchy and simple constraints. For instance, while it supports cardinality constraints, it only permits cardinality of 0 or 1. It should be simpler to provide tool support for OWL Lite than its more expressive relatives, and is good to quickly migrate path for thesauri and other taxonomies. It also has a lower formal complexity than OWL DL. The advantage of this is a language that is both easier to learn and implement. The major disadvantage is a restricted expressivity.
- **OWL DL:** It supports who want the higher expressiveness while ensuring that all conclusions are guaranteed to be computable in finite time. It includes all OWL language constructs which can be used only under certain restrictions. The advantage of this is that it permits efficient reasoning support. The disadvantage is that we loose full compatibility with RDF [77].
- **OWL Full:** It is designed for those who want utmost expressiveness and the syntactic freedom of RDF and no hazard about computational time and decidability. The advantage of OWL Full is that it is completely compatible with RDF, both syntactically and semantically. A legal RDF document is also a legal OWL Full document, and any valid RDF/RDF-S conclusion is also a valid OWL Full conclusion. The disadvantage of OWL Full is the language has become as powerful as thinkable but lack of hope of full reasoning support [69, 77].

## 4.10 Ontology Editor Tools

There are many ontology development tools, few main are discussed briefly as follows:

### 4.10.1 Protégé

Prot´eg´e, an open-source tool developed at Stanford Medical Informatics, has a community of thousands of users. Like most other modelling tools, the architecture of Prot´eg´e is cleanly separated into a "model" part and a "view" part. Prot´eg´e's *model* is the internal

representation mechanism for ontologies and knowledge bases. Prot'eg'e's *view* components give a user interface to show and manipulate the underlying model. Prot'eg'e's *model* is based on a simple but dynamic meta-model [11]. The model is comparable to object-oriented and frame-based systems. It can represent ontologies consisting of classes, properties, property characteristics, and instances. Prot'eg'e provides an open Java API to query and manipulate models. Using the *views* of Prot'eg'e's user interface, ontology designers basically create classes, assign properties to the classes, and then can apply restrictions [81].

### 4.10.2 SWOOP

SWOOP is an application to create, edit, and debug OWL ontologies. It was developed by laboratory named as MIND lab at University of Maryland, but is now an open source project with contributors from all over the world [85]. Swoop is no longer under active development at mindswap, the project for which it was created [86].

### 4.10.3 OntoEdit

OntoEdit is a development environment for create, revision and import of models representing knowledge for application systems. Typically the development of ontologies involves collaborative efforts of many persons. It is an ontology editor that integrates numerous aspects of ontology engineering [87].

### 4.10.4 Altova Semantic Work

Altova SemanticWorks® 2009 is a good RDF and OWL editor for the ontology language. OWL ontologies, graphically designed RDF instance documents are created and then output them in either RDF/XML or N-Triples formats. SemanticWorks makes the work easy with separate interface tabs for individuals, properties, concepts and automatic format checking [88].

### 4.10. 5 WebODE

The current version WebODE 2.0 is the upgraded version of the ontology design environment ODE. It is standalone ontology tool based on tables and graphs allowing users to modify the knowledge model used for conceptualizing their ontologies according to their knowledge representation needs. Its ontology editor integrates most of the ontology,

Ontology based knowledge management system (ODEKM), Semantic Web portal generator (ODESeW), a Web resources annotation tool (ODEAnnotate), and a Semantic Web services editing tool (ODESWS) [82].

## 4.11. Ontology Modeling Paradigms (Frames and OWL)

### 4.11.1 Introduction

Frames and OWL are the two ontology modelling paradigms. There are many similar modelling constructs both of them. The two are built around the notion of classes, representing concepts in the domain of interest. Classes have instances, properties define characteristics of those classes and relationships between them, while restrictions constraints on the values of properties. There is however major differences in the semantics of these constructs and in the means these constructs are used to infer new information in the ontology or to determine whether the ontology is consistent or not. Resultantly, the way that the modelling constructs are used in the two paradigms and the implications of definitions are different.

### 4.11.2 Differences

- **Unique name supposition:** If, in frames, two objects have different names, they are supposed to be different, if not explicitly stated otherwise. In OWL, no such supposition is made.
- **Closed World supposition vs Open World supposition** Using Frames, everything is prohibited until it is permitted while in OWL, everything is permitted until it is prohibited. This is difference like pessimistic vs optimistic. Nothing can be entered into a Frames KB until there is a place for it in the corresponding template while anything can be entered into an OWL KB unless it violates one of the constraints.
- **Single vs Multiple models:** Ontology based on frames has only one minimal model that satisfies each of the assertions of the Frames ontology. This means that models for frames ontology can only contain instances that are explicitly specified. In

general, OWL ontology will have many models consisting of all possible interpretations that satisfy each of the assertions in the OWL ontology.

- **Assertion vs Classification:** Using Frames, defining restriction (facets) on a property (slot) at a class, or defining a constraint on a slot at the top level, makes a statement about all instances of that class. On the other hand, in OWL, there are effectively two kinds of statements about classes: 1) same as frames, restriction will be applied to all instances of class. 2) OWL classifier can use the sufficient conditions to infer which classes are subclasses of the defined class. There is no equivalent feature in Frames. It is most important deficiency for our research as for multiple translations, equivalent feature is required.
- **Consistency checking vs Constraint checking:** The reasoner that checks the classification also checks the consistency of OWL KB. The classifier tries to build a model that satisfies all the axioms in the ontology. The ontology is inconsistent if no such model can be built. When building a model that satisfies all the assertions, a classifier may assign new types to ontology instances, in addition to the types explicitly asserted by a modeler. On the other side, Frames reasoner checks if the constraints are satisfied by the property values on instances, then fine else the instance is non-conformant [84].

## 4.12. Methodology for Ontology Building.

It is noteworthy that there are many methodologies as discussed below but these is not a standard and as ontology development is creative and tough work thus usually it depends upon ontology for selection of its methodology but majority do not use either, as recommended in chapter of recommendations.

### 4.12.1 METHONTOLOGY

METHONTOLOGY, developed by the Ontological Engineering group at echnical University of Madrid, has been used by different groups to build ontologies on Chemistry, Science, knowledge management, e-commerce, etc. METHONTOLOGY allows modelling ontologies through intermediate level representations in graphical as well as tabular intermediate representations that can be understood by experts in one domain even if they are not deeply involved in the ontology field [82].

### 4.12.2 CommonKADS and KACTUS

CommonKADs is a methodology for the development of knowledge based systems having ontologies as important part of it. The KACTUS project was a follow-up project having main aim of ontology development. In it, approach for ontology engineering is adopted, stressing modular design, redesign and reuse. Ontology is constructed from a library of small-scale ontologies requiring mapping among various ontologies to develop new ontology [83].

### 4.12.4 SENSUS

This approach is based on the supposition that if two knowledge bases have been developed on the basis of common ontology, then knowledge can be shared between them more readily since they share a common structure. In the development of a domain-specific ontology, some basic terms known as seeds are selected as representative of relevant domain-specific concepts. Seeds are then linked (by hand) to SENSUS [83].

### 4.12.5 Uschold and King Method

Uschold and King proposed first mthoed of building ontologies. It was based guidelines on their experiences. To build ontology, the following process must be performed: (1) identify the purpose of the ontology, (2) build it (3) evaluate it (4) document it. Then evaluation was also added in these steps [89].

### 4.12.6 Cyc method

It is a huge knowledge base built to gather a large portion of what people usually recon as consensus knowledge about the world. The CycL language was developed for implementation. It is considered as ontology as it may be considered as substrate for building different intelligent systems that can communicate and interact [89].

## 4.13 Reasoning Engines

The use of inference engines or reasoner is used to let applications to inquire as to why a particular conclusion has been reached and also to derive new associations and/or facts from existing information. The brief description of main reasoners is as follows:

### 4.13.2 Jena

RDFS and OWL reasoning are seen as graph-to-graph transforms, producing graphs of virtual triples. In Jena, rich APIs are provided. The Ontology API includes support for RDFS and all forms of OWL. Jena includes reference RDF/XML parser, and offers RDF/XML output using the rich RDF/XML grammar. Jena's query language, RDQL, and the Web API are offered for the next round of standardization [90].

### 4.13.3 KAON2

KAON2 are based on special purpose algorithms. It transforms OWL DL ontologies to disjunctive data-log and is used to apply algorithms for dealing with this formalism [91] .

### 4.13.4 Java Theorem Prover (JTP)

JTP, developed by Knowledge Systems Laboratory of Computer Science Department in Stanford University, is an object-oriented modular reasoning system. It is based on simple general reasoning architecture. It is suitable to expand the system by adding new reasoning modules due to its modular character of the architecture [92].

### 4.13.5 Ontobroker

Ontobroker includes a query interface for writing queries, an inference engine to derive answers to the posed queries, and a crawler for searching and translating the ontological annotations into facts for the inference engine. It is used for developing a knowledge management system for industrial designers in regard to ergonomic questions [93].

### 4.13.6 FaCT++

FaCT++ is a new DL reasoner designed as a platform for experimenting with new tableaux algorithms and techniques. Tableaux algorithms are the one used to reason with OWL ontologies. It consists of standard optimisation techniques included in the FaCT system and also employs many novel ones. This architecture allows for a wider range of heuristic optimisations [94].

### 4.13.5 Pellet

Pellet is OWL-DL reasoner with good performance, extensive middleware, and a number of unique features. Pellet is the first sound OWL-DL reasoner with extensive support for

# CHAPTER NO. 5

# Implementation

# CHAPTER NO. 5

# Implementation and Results

And We have indeed made the Qur'an easy to understand and remember: then is there any that will receive admonition?

(Surah: Al-Qamar [54:40-40] )

---

## 5.1 Implementation Issues

There are many basic implementation issues, which are discussed in detail as under .

### 5.1.1 Selection of Domain for Sample Ontology

As Holy Quran is the having information about many subject matters, thus, there are a lot of domains that may be considered for construction of initial and sample ontology. Many domains like Prophet, Events, Characteristics of Muslims and Non-Muslims, Concept of Ibadah (Prayers), Concept of Ahkaam (Orders) and Nahee (Restrictions) etc were considered for the construction of domain ontology. We do have the knowledge that there are different understanding of the same verse in scholars of different sects and schools of thoughts. Thus, definition of such Islamic beliefs may well require very extensive care and team of different scholars should be given a task to define the concepts in such a way that it should be accepted universally. Thus all such domains have been avoided and domain of animals mentioned in Holy Quran has been selected. The Basic behind it was that it is absolutely free of such concepts that can cause differences. Additionally the concepts are largely physical and well defined. When it was selected that it also helped as the taxonomy of the animals is well defined and unambiguous.

### 5.1.2 Difficulties in Animal Domain

Simple animal ontology is easy to build as taxonomy whether it be general or scientific is easy to formulate and develop, but the definition of animal related concepts present in Holy Quran are difficult to define. First, it is worth-mentioning that we have only tried to declare

all concepts exactly as defined in Holy Quran. There are about 167 direct or indirect references of animals in Holy Quran.[7]. Main difficulties include

1. In Arabic for one animal many Arabic words are present. For instance, a camel has words like Abal, Buaaeer, Gamal.
2. Majority of animals have been mentioned regarding giving similarity to some people. This creates difficulty in creating metaphorical or abstract relationship. For example, the non believers who have no faith in Holy Quranic verses are resembled to donkey carrying books.
3. Many animals have been mentioned regarding their behavior. For instance, ass, mule and horse have been defined as source of usage of ornament and carrying luggage. Behavior is such a difficult domain that two universities SIG are extensively working merely on the domain of animal behavior. [101]
4. More problematic, when animals are mentioned in some specific context. For instance, donkey gets frightened when sees a lion. As a relation of context is NLP is still a problem under consideration thus these were challenging scenarios. A separate ontology structure has been defined for context based animal referencing.
5. There is very thin line between defining basic functionality of animal and its behavior. Thus, concept of property and class has been used to separate. This is defined as separate two classes of enumerated type in separate ontology.

### 5.1.3 Selection of Translation

As already mentioned, we have to be extremely careful about selection of domain regarding issues like acceptance by all, the selection of translation is not different. As the many modern translations are linked to some particular school of thought thus translation of Pickthall has been selected. [102]. It is regarded as one of classical and authentic English translations as done by the person who has English as first language.

There come obvious question that this will be very limited to only one translation. Thus, to overcome of this problem, after research, we have come to solution of equivalent class. Thus, with addition of new term and making it equivalent to some already concept fulfills the requirement.

### 5.1.4 Precautions about development of Ontology from domain of Holy Quran.

Being a Muslims, we have basic belief that words of Holy Quran are divine and eternal. No alteration is it whatsoever is possible as Allah Almighty Himself has taken into His Hand. Even being static document, we still have to be very cautions about the following regarding definition of concepts in Holy Quran.

1. Define the concepts exactly the same manner defined in Holy Quran. No perception, addition of concept, addition of characteristics should be allowed. Even this addition is obvious ad right one.
2. If extensibility of ontology is required, for instance, we want to define the terms as mentioned in Holy Quran as well as in Hadith books, then either new ontology should be developed and Holy Quran ontology should be used Upper ontology. [103].
3. If we want to add definition of other knowledge domain then separate ontology should be developed and mapping/merging of that ontology should be carried out. For instance, if we want to improve the search of animals mentioned in Holy Quran then we can define the taxonomy of animals on scientific basis and map with our ontology.

## 5.2 Steps in Ontology Construction

Ontology construction is a creative work. Ontology are iterative in nature and never complete. There are many steps and issues that are considered in sequence to create an ontology. [104] is the regarded as best source for first ontology construction written by Stanford University, Stanford, CA, main group working on semantic web. It elaborates the steps in details. These steps are from Stanford developers whose group are leading one and developed protégé too. These steps are taken from this document and discussion of construction of our ontology is being mentioned in line with those of the document.

### 5.2.1 Determine the domain and scope of the ontology

The domain of out ontology is animal s references in Holy Quran. It not only includes animals but also birds mentioned. The domain is very clear and no ambiguity like overlapping of basic concepts is overlapping or confusing one. The scope of the ontology is intuitive as main purpose is to define the concepts as defined in Holy Quran. The main purpose obviously is that this ontology will help in semantic search in Holy Quran. The users

of this ontology can be all those researchers who want to take work to bigger level to implement semantic search for about all main domains within Holy Quran.

### 5.2.2 Consider reusing existing ontology

As far as out knowledge is concerned, there is no ontology present which may have defined as concepts defined in Holy Quran. This there is no question of reusing. Yes, there are many ontologies for definition of concepts regarding animals **[105,106]**, but all these do not fulfill our requirement of definition of concepts as defined in Holy Quran as these are basically based on general or scientific taxonomy.

### 5.2.3 Enumerate important terms in the ontology

This was quite extensive work as it was meant to be done extensive manual exploration work. All the verses of Holy Quran has been searched by using certain index books **[5,6,7]** and helping material. The terms, at start, were identified as follows:

| Agents | Animal Classes | Animal Instances | Evens | Objects | Property | Body Part | Other Classes |
|---|---|---|---|---|---|---|---|
| ALLAH | Animal | | Create, slaughter | | | | Prophets |
| Musa (A.S) | Cow | Cow_Musa | Sacrifice, Plough, Irrigate, | Water | Fat, thin | | People-Group |
| Nation_Musa , | Calf | Calf_Musa, Calf_Ibrahim | Adopted, Brought | | Fat | | Uses |
| Yousaf (A.S) | Camel | | Punish | Fields | Color | | Function |
| Yunas (A.S) | She-Camel | She-Camel_Saleh | | Legs | For Load | | Behavior |
| Ibrahim (A.S) | Horse | Horse ? | Eat | Tree | Ride, Ornament , Mark, For Award | | Things |
| Sulayman (A.S) | Donkey | Ass/Donkey | Panicked, See | | Worst Sound | | |
| | | Mule | | | Ride, Ornament | | |
| SAMOOD | Dog | Dog_Kauf | Lolls out + Pants, spread | | Similar | Tongue, Hands | |
| Nation_Firon | | Gnat/Mosquito | Similitude | | Fat | | |
| | | | | | | | |
| Musa (A.S), Yunas (A.S) | Fish | Fish_Yunas, Fish_Musa | Make way into, Rise above water, Forget, Devour | | Sound | | |

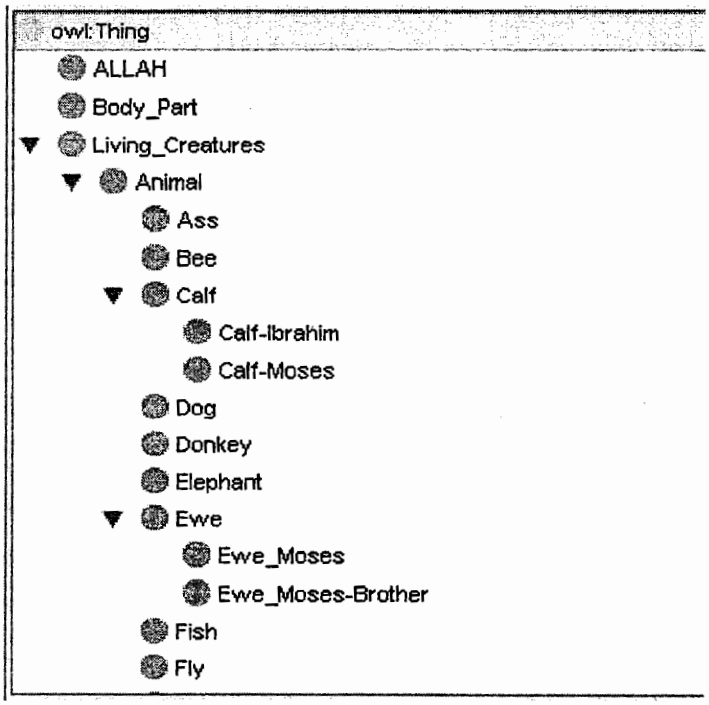
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Musa (A.S) | | Snake ? | Become, | Stick | | | |
| ALLAH | | Elephant | Say ??? | | | | |
| | | Wolf | Fly | | | | |
| Non-ALLAH | | Spider | Owns | House | Flimsier | | |
| Non-ALLAH | | Fly | Create, Steals | | | | |
| | | | | | | | |
| ALLAH | | Bee | Make, Drink, Eat | House, Tree, Mountain, Fruit, | | | |
| | | Ant | Says (to Ant) | | | | |
| | | Mule | Water | | | | |
| | | Lion | Sent Down/ | | | | |
| Nation_Musa | | Pig ??? | Converted | | Haram/Forbidden | Swine /pork (meat) | |
| Nation_Musa | | Monkey | Converted | | | | |
| | | | | | | | |
| Musa (A.S) | | Goat | | | | | |
| Saleh (A.S) | | | | | | | |
| | | | Slaughter | | | | |
| | | | Spread | | | | |
| | | | | | Thin | | |
| ALLAH | | Lice | Sent down | | | | |
| ALLAH | | Locusts | Sent down | | | | |
| ALLAH | | Frog | Sent down | | | | |
| | | | | | | | |
| | | | | | | | |
| | | Crow | Scratch | | Earth | | |
| ALLAH, Nationa_Musa | | Quail | Sent down | | | | |

**Table. 5.1 Enumeration of Important Terms in the domain knowledge**

### 5.2.4 Define the classes and the class hierarchy.

Following the work of extraction of important terms, classes have been defined and class hierarchy has been defined as well. Here is picture of classes from part of ontology. It is important to mention that there comes ambiguity of whether one concept should be defined being class or its instance. If it can not be decomposed into further characteristics, it is defined as instance, but the extensibility purposed, we have focused of creating more classes so that in future, more characteristics regarding other knowledge subjects like science, Hadith

etc can easily be made part of it. The Class hierarchy showing classes is shown . The pictures have been taken from Protégé tool.




Frog
Gnat
Goat
Horse
Branded_Horse
Lion
Locust
Mosquito
Mule
Snake
Spider
Swine
Vermin
Bird
Crow
Quail

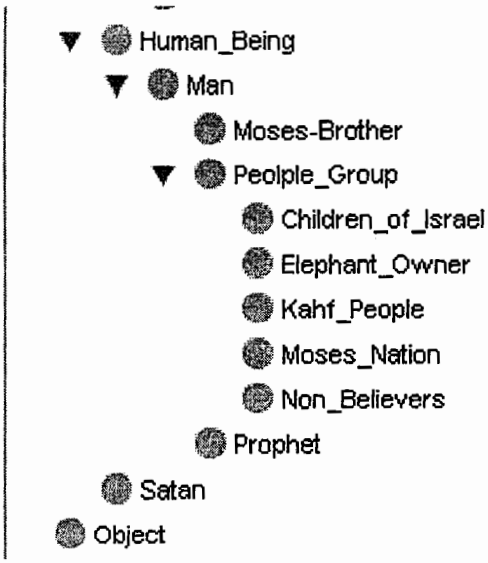



**Figure: 5.1 Class Hierarchy**

### 5.2.5 Define the properties of classes- Slots

Properties are one of the important part as these define the characteristics of classes and also can be used to make the relationship among classes. As defined earlier, there are two types of properties, object properties and data properties. It is iterated that only those characteristics of concepts have been created which are and as defined in Holy Quran.

It is notable that as Holy Quran defines the concepts according to its characteristics thus, about more than 90% properties are object properties. Lack of data properties depicts that very few numeral and data type values are mentioned in Holy Quran. The list of properties is as follows:

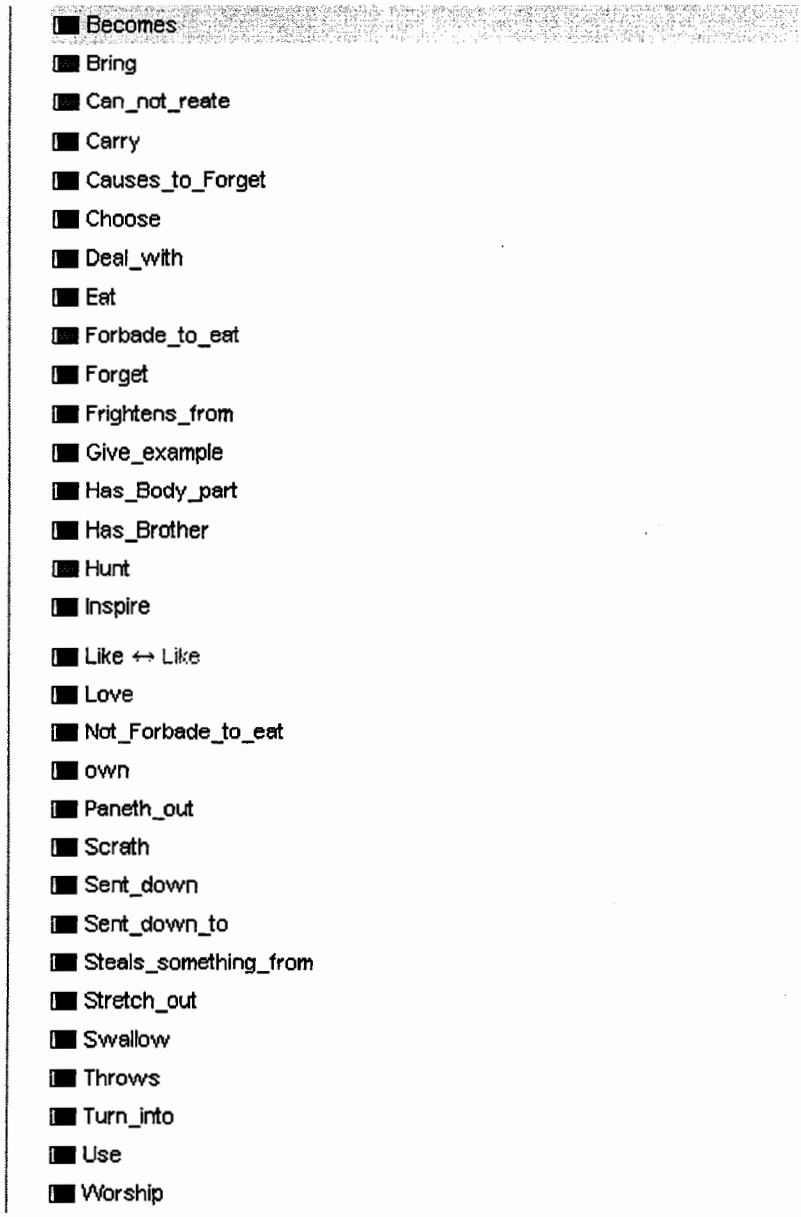



**Fig. 5.2 List of Object Properties**

The list of data properties is as follows:

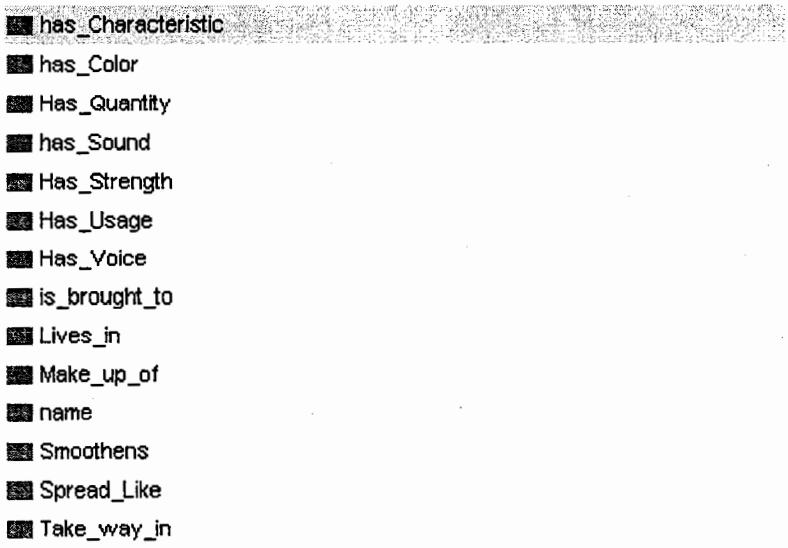



**Fig. 5.3 List of Object Properties**

### 5.2.6 Define the facets of the properties.

Defining the facets includes defining the data type of data object and defining the domain and range of the object properties. Thus, data type of very few data types have been defined and domain and range have also been given.

The complete list of the properties and facets is as follows:

| Name | Prefix | Range | Domain | Inverse | Other Characteri... |
|---|---|---|---|---|---|
| Becomes | | Snake or Swine | Moses_Nation ⊔ Object | | |
| Bring | | Calf-Ibrahim | owl:Thing | | |
| Can_not_reate | | Fly | Man | | |
| Carry | | Object | Ass | | |
| Causes_to_Forget | | | Satan | | |
| Choose | | Calf-Moses or Object | Bee ⊔ Moses_Nation | | |
| Deal_with | | Elephant_Owner | ALLAH | | |
| Eat | | Goat or Object or Qu... | Bee ⊔ Children_of_Israel ⊔ Man | | |
| Forbade_to_eat | | Swine | ALLAH | | |
| Forget | | Fish | owl:Thing | | |
| Frightens_from | | Lion | Ass | | |
| Give_example | | Gnat | ALLAH | | |
| Has_Body_part | | Body_Part | Bee ⊔ Goat ⊔ Horse | | |
| Has_Brother | | Moses-Brother | owl:Thing | | |
| Has_Quantity | | owl:oneOf{} | Ewe_Moses ⊔ Ewe_Moses-Brother | | |
| Has_Strength | | owl:oneOf{"Weakes... | Object | | |
| Has_Usage | | owl:oneOf{"Orname... | Ass ⊔ Horse ⊔ Mule | | |
| Has_Voice | | owl:oneOf{"Harshes... | Ass | | |
| Hunt | | Fish | Man | | |
| Inspire | | Bee | ALLAH | | |
| Like | | Dog or Non_Believers | Ass ⊔ Non_Believers | Like | Symmetric |

| Name | Prefix | Range | Domain | Inverse | Other Character |
|---|---|---|---|---|---|
| Lives_in | | owl:oneOf("Sea") | Fish | | |
| Love | | Branded_Horse | Man | | |
| Make_up_of | | owl:oneOf("Orname... | Calf-Moses | | |
| Not_Forbade_to_eat | | Goat | ALLAH | | |
| Paneth_out | | Body_Part | Dog | | |
| Scrath | | Object | Crow | | |
| Sent_down | | Crow or Frog or Loc... | ALLAH | | |
| Sent_down_to | | Children_of_Israel | Quail | | |
| Smoothens | | owl:oneOf("Ways fc... | ALLAH | | |
| Spread_Like | | owl:oneOf("Men cor... | Locust | | |
| Steals_something_from | | Man | Fly | | |
| Stretch_out | | Body_Part | Dog | | |
| Swallow | | Prophet | Fish | | |
| Take_way_in | | owl:oneOf("Water") | Fish | | |
| Throws | | Object | owl:Thing | | |
| Turn_into | | Moses_Nation | ALLAH | | |
| Use | | Horse | Man | | |
| Worship | | Calf-Moses | Moses_Nation | | |
| has_Characteristic | | owl:oneOf("Fat" "Ro... | owl:Thing | | |
| has_Color | | owl:oneOf("Yellow") | Calf-Moses | | |
| has_Sound | | owl:oneOf("Lowing") | Calf-Moses | | |
| is_brought_to | | owl:oneOf("Angels") | Calf-Ibrahim | | |
| name | | rdfs:Literal | Human_Being | | |
| own | | Dog or Elephant or E... | Elephant_Owner ⊔ Kahf_People ⊔ Moses-Brothe... | | |

**Fig. 5.4 List of Object Properties along with Domain and Range**

### 5.2.7 Create Instances

The last step is to instantiate the classes by making objects/instances/individuals of the classes. It is important to mention that concepts defined in Holy Quran once are defined as classes and same name with postfix of _1 depicts its instance except Allah where ALLAH depicts class while Allah depicts instance. But as the main idea behind ontology creation was to define concepts and thus only following instances have been created to depict the running of SPARQL queries.

| Class | Instance |
|---|---|
| ALLAH | Allah |
| Ass | Ass_1 |
| Fish | Fish1 |
| Horse | Horse_1 |
| Branded_Horse | Branded_Horse_2 |
| Mule | Mule_1 |
| Swine | Swine_1 |
| Man | Man_1 |

| | |
|---|---|
| Prophet | Ibrahim, Moses, Ibrahim |

**Table 5.2. List of Instances vs Classes**

## 5.3 Issues regarding Ontology Creation

There is a list of certain issues that must be addressed while creating ontology.

- **Ensure that class hierarchy is correct.** It can be done by checking that in Protégé tool, when the user defined ontology can be checked and it resultantly it gives Asserted ontology. If ontology is correct already, then it will not give Asserted ontology, the same happened for our ontology too.
- **Analyze siblings in class hierarchy.** This is quite interesting that we should be clear that we are defining siblings for the correct. Does all siblings are logically similar type.
- **Multiple inheritance.** This is another interesting phenomenon and very similar to that of OOP concepts. Multiple inheritance is used if we want to inherit class from more than one class.
- **When to introduce new class.** This is important for limiting or extending the details of the concept. An ontology is never complete, so introducing the concepts related to already defined concepts depends on till what limit we want to define the relationship of concepts.
- **A new class or property value.** It is very important because some times when we want to create property then it is also possible that that property itself be defined as new concept. Also data property can be defined as enumerated class.
- **An instance or a class.** The question whether one concept should be defined as class or instance depends on whether we want to make our ontology extendable or not. As if we define class then it can be further inherited. An instance is chosen if we have completely defined in such a way that it further can not defined.
- **Limiting the scope.** Limiting the scope actually determines the breadth of ontology. Ontology should be in such a way that it can be used for some specific type of application. General ontology usually gives more definition but less inference.

- **Disjoint classes.** Disjoint classes is one which must be very distinguished fromeach other. Also disjoint classes has the property that an instance must not be created that may lie in hierarchy of disjoint classes.
- **Naming conventions and naming issues.** To me, naming convention and code reusability are the two salient features of object oriented programming and same can easily be applied to ontology. Naming consistency makes the work of development, modification and updation easy.

## 5.4 Checking Class hierarchy

Protégé provides the facility to use reasoner to check whether the ontology that was created is consistent and valid or not. Pellet reasoner is the built-in by default reasoner that is used to check the correctness of ontology. The user defined hierarchy is known as Asserted hierarchy while the correct hierarchy created by the pellet reasoner is known as Inferred ontology. If the ontology is not correct and can not be corrected then it gives error, else it re-generates the same ontology. Problem is non-consistent ontologies include when we are using multiple inheritance, un-natural class relationship etc. The sample ontology has been checked by pellet reasoner and both Asserted and Inferred ontologies are same as shown in the following figure.

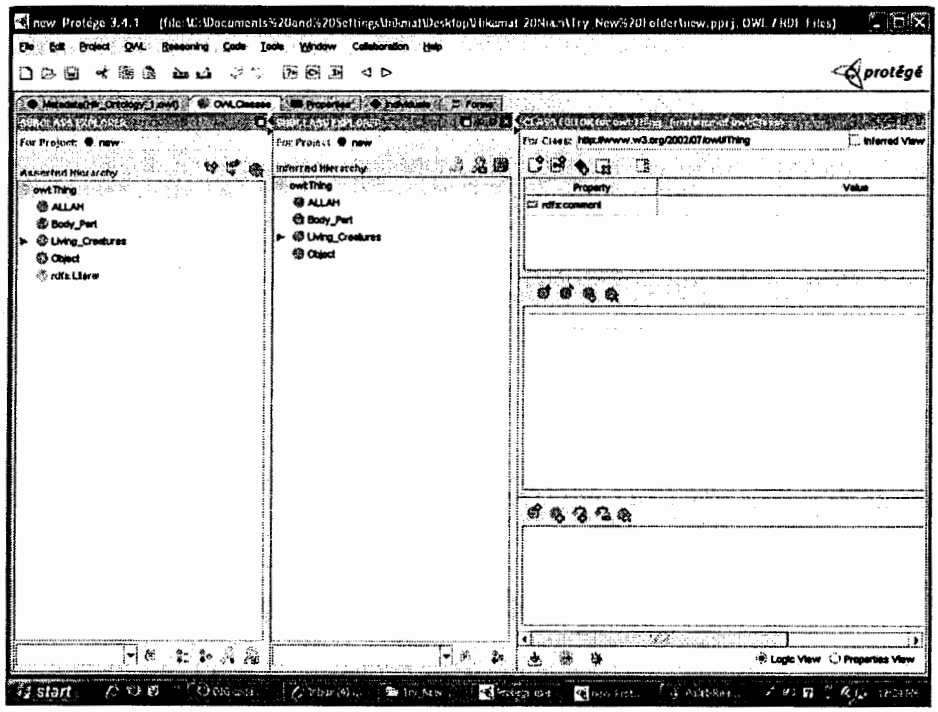



**Fig. 5.5 Checking of consistency of ontology by reasoner**

## 5.5 SPQARL Queries and Results

To depict the accuracy and functionality of the ontology that concepts are defined correctly and also that we can get answer of the concepts from relationship of other concepts and as well as from its characteristics not necessarily from its name. Thus ensures that using ontology we can do semantics manipulation and inference.

### 5.5.1 Classes and subclass hierarchy checking

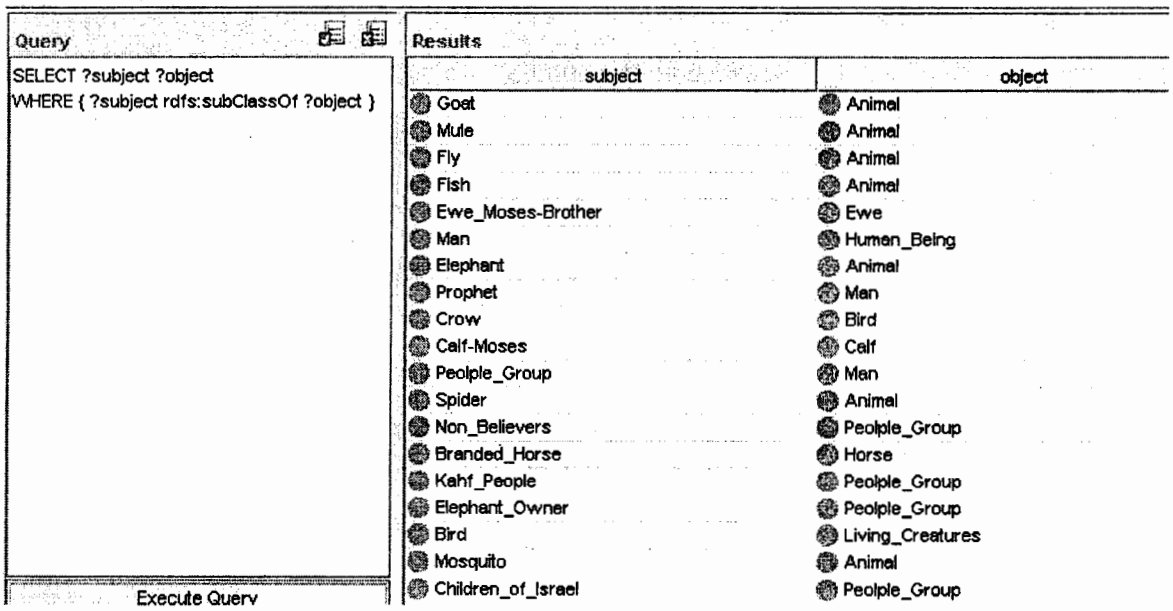
Q1: Provide list of subject and objects in the ontology.

SPARQL Query:

*SELECT ?subject ?object*

*WHERE { ?subject rdfs:subClassOf ?object.}*

Answer: List of subject and object is the result shown here.



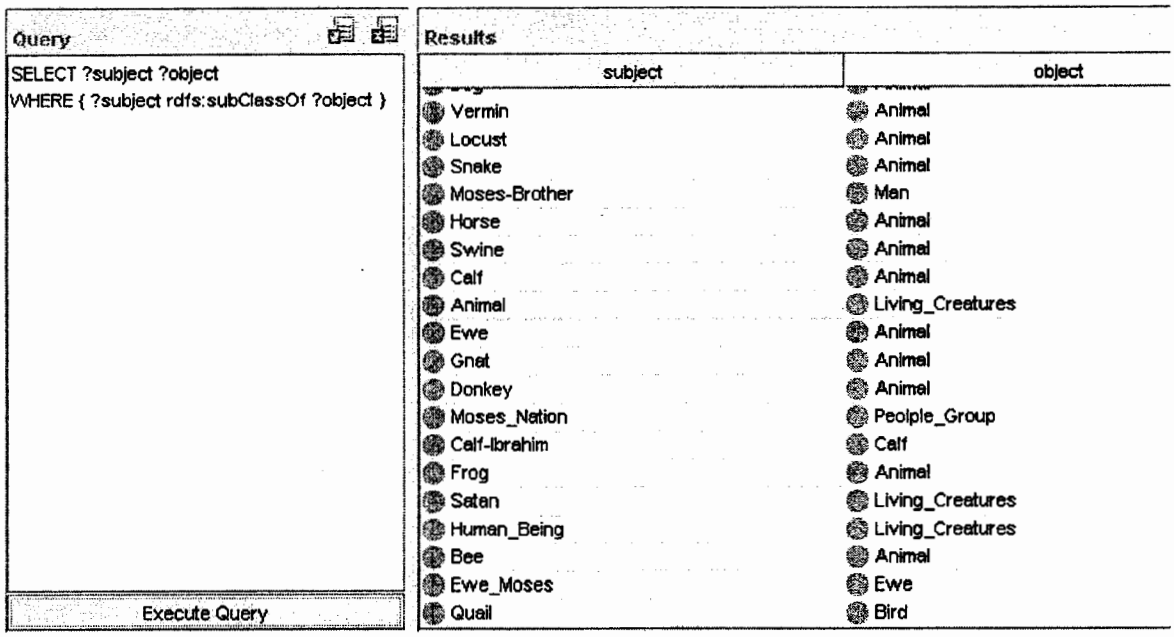



**Fig. 5.6 List of Class and Subclass on the base of SPARQL Query**

### 5.5.2 SPARQL Queries for Checking of Semantics

**Scenario 1:** There are two questions in which concept is search wrt to its object property to other concept. Then the same concept has been checked but also the animal has been seached wrt its property to line in sea.

Q 2;Which animal swallowed Yunas (A.S).

Answer: Fish

SPARQL Query:

*SELECT ?animalClass*

*WHERE { ?animal :Swallow :Yunas.*

*:Yunas rdf:type :Prophet.*

*?animal rdf:type ?animalClass.*

*?animalClass rdfs:subClassOf :Animal }*

Q3: Which animal swallow Prophet and also lives in Sea? Answer: Fish

SPARQL Query:

*SELECT ?animalClass*

*WHERE { ?animal :Swallow :Yunas.*

*:Yunas rdf:type :Prophet.*

*?animal rdf:type ?animalClass.*

*?animalClass rdfs:subClassOf :Animal.*

*?animal :Lives_in "Sea" }*

**Scenario 2**: Here it has been shown that concept can be found from one concept to other class in hierarchy. (class man, object property: love, class: branded_horse subclass of horse)

Q4: Which animal is loved by Man. A: Branded Horse subclass of Horse

SPARQL Query:

*SELECT ?animalClass*

*WHERE {*

*?man :Love ?animal.*

*?animal rdf:type ?animalClass*

*}*

Q5: Which Animal is forbidden (to eat) by ALLAH A: Swine

SPARQL Query:

*SELECT ?animalClass*

*WHERE {*

*:Allah :Forbade_to_eat ?animal.*

*?animal rdf:type ?animalClass*

*}*

**Scenaio 3: (class: ALLAH, object property: sent down, class)**

ALLAH sent down Quail to Children of Israel (Bani-Israqel), so here first query checks for subject while second checks for object, third checks related concept of object.

Q6: Who send Quail to Children of Israel? A: ALLAH

SPARQL Query:

*SELECT ?subject*

*WHERE { ?subject :Sent_down ?object.*

*?object rdf:type :Quail.*

*?object :Sent_down_to ?coi.*

*?coi rdf:type :Children_of_Israel }*

Q7: Which Bird was sent down to Children of Israel: A: Quial

SPARQL Query:

```
SELECT ?birdClass
WHERE { ?bird :Sent_down_to ?coi.
?object rdf:type :Children_of_Israel.
?bird rdf:type ?birdClass.
?birdClass rdfs:subClassOf :Bird
}
```

Q8: Children of Israel eat which Bird A: Quail

SPARQL Query:

```
SELECT ?birdClass
WHERE { ?coi :Eat ?bird.
?bird rdf:type ?birdClass.
?coi rdf:type :Children_of_Israel
}
```

**Scenario 4:** Nowly, lastly, the ontology checking via data property. One data property has_usage and string values Ride and ornament are linked to not only only class but to three different concepts of Horse, Mule and Ass ).

Q9: which animal has usage of ride: A: Horse, Mule and Ass

SPARQL Query:

```
SELECT ?animalClass
WHERE { ?animal :Has_Usage "Ride".
?animal rdf:type ?animalClass
}
```

**Scenario 6: (again data property has_vioce, string value harshest)**

Q10: which Animal has harshes voice A: Ass.

SPARQL Query:

```
SELECT ?animalClass
WHERE { ?animal :Has_Voice "Harshest".
?animal rdf:type ?animalClass }
```

# CHAPTER NO. 6

# Recommendations

# CHAPTER NO. 6

# RECOMMENDATION

Qaf. I swear by the glorious Quran (that Muhammad is the Apostle of Allah.)
(Surah: Qaf [50:1-1] )

---

## 6. Recommendations

It is worth mentioning that main objective of the research is the exploratory study to be carried out in the new filed of Semantic web and recommendations be made for the semantic search in the Holy Quran. These recommendations may set foundation for carrying out such activity while forming into the project done at government level having sufficient team and finances. It can also be start of some sort of SIG – research group that can take this recommendation work into a reality. It is notable that all recommendations are being made on the base on exploratory work mentioned in chapter 4 and also extensive reading work was carried out and such surveys have been selected for recommendation which provide recommendation on the basis of logic, research work and experimentation.

## 6.1 Selection of Ontology Modeling Paradigms

There is just two basic ontology modeling paradigm - Frames or OWL. The various differences between frames and OWL could prove intimidating to users who need to choose methodology for development of their systems. In some cases, the frame representation with its closed-world semantics is a good fit for the application, while in other cases; the potential and clarity of OWL are needed to deliver the functional requirements.

Generally, Frames is particularly useful when the application has the following requirements:

• Creation of ontologies for domains having closed-world assumption is appropriate.

• The main purpose of application is data acquisition.

• Application domain requires constraints on slot values.

• The model links classes to other classes.

Likewise, the following requirements of applications may make OWL a preferred choice:

• Creating robust terminologies in which classes are defined.

• Need for DL reasoning to ensure logical consistency of ontologies.

• Controlled terminologies are published on the Semantic Web and accessed by other applications.

• Applications in which classification is a paradigm for reasoning [84].

Resultantly, OWL is recommended as Frames are for closed world assumption, where data acquisition is necessity and reasoning is not issue. OWL , on other hand, are simple choice due to extensibility, reasoning capability, controlled terminologies, excellent classification capability etc.

## 6.2 Selection of Basic Ontology Language.

Selection of ontology language is also interesting one. There are two options, RDF, then RDF-S is extended version. So let us compare the problems of RDF-S and

### 6.2.1 Problems with RDF-S

In the following we discuss some of the shortcomings of RDFS that can be expressed in OWL:

1. **Locale scope of properties.** For instance, we need to define a property `eats` that is used in the domain of the `Goat` and `Lion` classes. The `rdfs:range` primitive cannot be defined that a `Goat` only eats plants, while the `Lion` only eats meat.

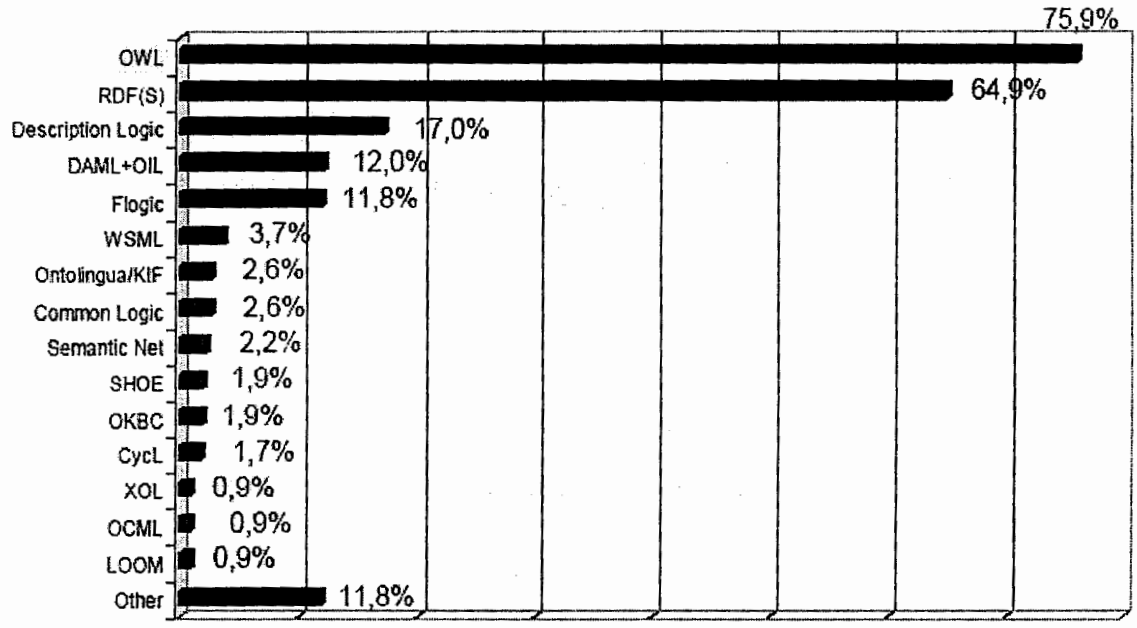
2. **Disjoint classes.** RDFS does not present primitives to define two classes to be disjoint. For example, the classes `Male` and `Female` are disjoint.

3. **Boolean combination of classes.** In some cases we have to define new classes by introducing the *union*, *intersection*, or *complement* of other classes. For example, we want to define the class `Person` to be union of the classes `Male` and `Female`.

4. **Cardinality restrictions.** RDFS does not provide any means to restrict the number of distinct values a property may or must take. For example, we want to specify that a person has exactly two parents, or that a course is studied by at least one student.

5. **Special characteristics of properties.** In RDFS we cannot declare that a property is *transient* (e.g. .greater than.), *unique* (e.g. .is mother of.), or the *inverse* of another property (e.g. likes. and is liked by.) [69].

In the light of above discussion, it is evident that RDF-S is lacking in some features which are important for contextual based domain ontologies from Holy Quran. DAML is automatically excluded as based on RDF [79]. Thus OWL is recommended. Our recommendation is validated by the survey research conducted [102] and the result in graphical way is as follows:




**6.1 Ontology languages used by users.**

## 6.3 Selection of Type of OWL

Three types of OWL have been exclusively discussed with advantages and dis-advantages in chapter of exploratory research.

Holy Quran domain ontology requires such a language that can describe difficult concept and can have control over description so that ontologies be consistent and reasoning can be applied. OWL –Lite is simple and merely for classification while OWL-Full on the contrary has complete power of expressiveness but no computational guarantees [77]. OWL-DL is recommended for its basic advantage because it is well controlled language

as it ensures computation completeness. Also it is best for reasoning and also ensures ontology checking and ontology consistency. OWL-DL, is in the middle of the two and sufficient for expressiveness as well as for inference.

## 6.4 Selection of Ontology Editors

The extensive research survey work conducted on the basis of 627 surveys that were filled in and conducted [102] and interesting results were found. Surveys were filled by diverse population covering researcher, professor, knowledge engineers, programmer, managers, system designers etc and reveals that more than 68.2% of them are using Protégé due to its exceptional features. The salient features of protégé [103] are as follows:

- Protégé is an open source ontology editor
- Protégé user community is very much involved in Protégé code development, contributing enhancements to the software on regular basis.
- Protégé user participating in online discussion groups involved in modeling questions, technical-support issues, and requests for new features.
- Can be linked with many reasoners like Jena, FaCT++, Pellet etc.
- Support of Query language like SPARQL.
- Consistent addition of Addins expanding diverse potentials of the editor.

The results of [102] have been presented in the graph as follows:

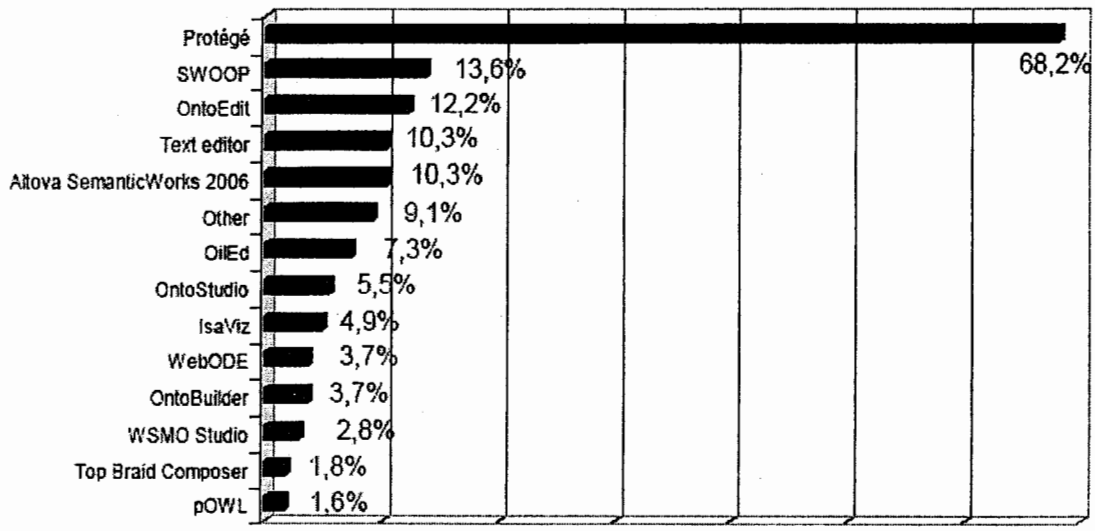



**Fig 6.2 Respondents' use of ontology editors**

## 6.5 Selection of Reasoning Engines

Jena is actually collection of following predefined reasoners:

- **Transitive reasoner**
  It presents support for storage and traversing of class and property lattices. This implements just the *transitive* and *reflexive* properties of `rdfs:subPropertyOf` and `rdfs:subClassOf`.
- **RDFS rule reasoner**
  It is used for implementation of a configurable subset of the RDFS entailments.
- **OWL, OWL Mini, OWL Micro Reasoners**
  It has a set of functional but incomplete implementation of the OWL/Lite.
- **DAML micro reasoner**
  It is used to enable the legacy DAML API to provide minimal (RDFS scale) inferencing.
- **Generic rule reasoner**
  It is rule based reasoner that supports user defined rules. Various strategies like forward chaining, tabled backward chaining and hybrid execution are supported.

The results of the survey are helping in our decisions. The largest segment (53.6%) indicated that they were using Jena (McBride, 2002). Smaller groups indicated they were using Racer (28%) and Pellet (23.7%). FaCT++ (13.3%) and OWLJessKB (8.1%) have also gained preference by few contributors. Also, engines such as OntoBroker, JTP, KAON2, TRIPLE, etc have small market share. The survey results [102] in comparative graphs are as follows:

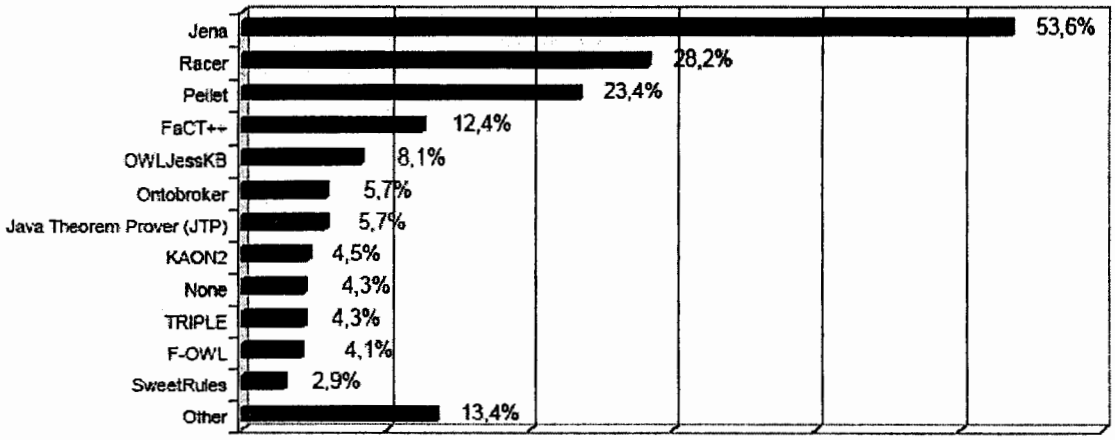



**Fig: 6.3 Reasoning engines utilization.**

## 6.6 Selection of different ontological Concepts

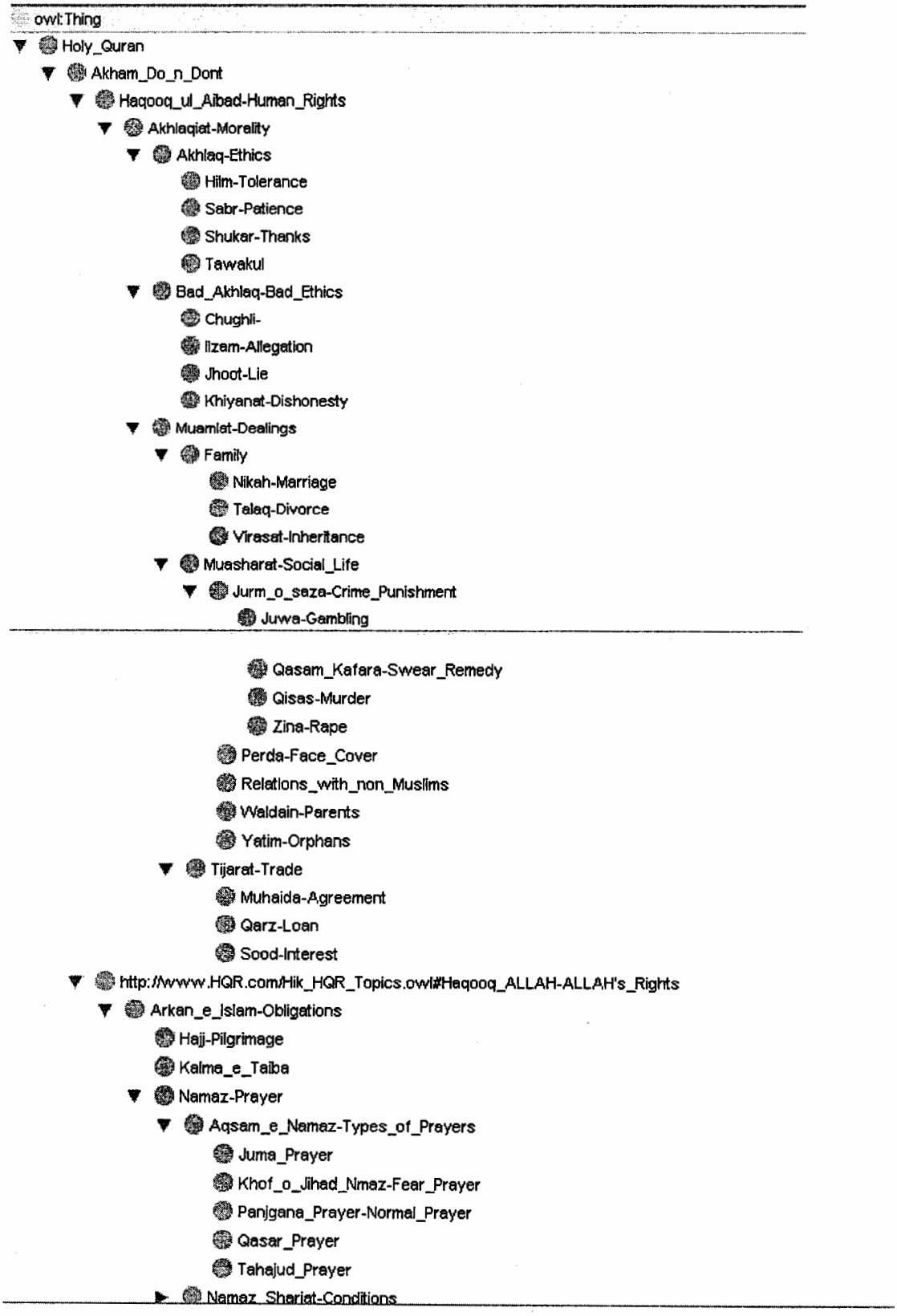
There is just general discussion of ontological concepts that are recommended for semantic search in Holy Quran.

### 6.6.1 Upper Ontology

There is necessity of main Quranic concepts to be declared explicitly above the definition of domain ontological concepts. Such sort of definition of concepts is referred as Upper ontology or Foundation ontology. Such ontology helps to give hierarchy of concepts at abstraction level and helps in modularity of whole project and removal of confusion of ambiguous terms. The classes and properties defined in this ontology should not be related to any specific domain or any special assumptions. This is quite different from a normal ontology, which is always tied to a particular knowledge domain.

### 6.6.2 List of Domains for ontologies from Holy Quran

There is a lot of books and research work done to the subjects mentioned in Holy Quran. The subjects have been declared on the basis of variety of reasons like religious concepts, some are based on Quranic indexes, some are based on scientific concepts etc. While the following subjects have been defined which have been defined so that domain ontologies for these subjects may be taken as initial ones.


owl:Thing
Holy_Quran
Akham_Do_n_Dont
Haqooq_ul_Aibad-Human_Rights
Akhlaqiat-Morality
Akhlaq-Ethics
Hilm-Tolerance
Sabr-Patience
Shukar-Thanks
Tawakul
Bad_Akhlaq-Bad_Ethics
Chughli-
Ilzam-Allegation
Jhoot-Lie
Khiyanat-Dishonesty
Muamlat-Dealings
Family
Nikah-Marriage
Talaq-Divorce
Virasat-Inheritance
Muasharat-Social_Life
Jurm_o_saza-Crime_Punishment
Juwa-Gambling
Qasam_Kafara-Swear_Remedy
Qisas-Murder
Zina-Rape
Perda-Face_Cover
Relations_with_non_Muslims
Waldain-Parents
Yatim-Orphans
Tijarat-Trade
Muhaida-Agreement
Qarz-Loan
Sood-Interest
http://www.HQR.com/Hik_HQR_Topics.owl#Haqooq_ALLAH-ALLAH's_Rights
Arkan_e_Islam-Obligations
Hajj-Pilgrimage
Kalma_e_Taiba
Namaz-Prayer
Aqsam_e_Namaz-Types_of_Prayers
Juma_Prayer
Khof_o_Jihad_Nmaz-Fear_Prayer
Panjgana_Prayer-Normal_Prayer
Qasar_Prayer
Tahajud_Prayer
Namaz_Shariat-Conditions

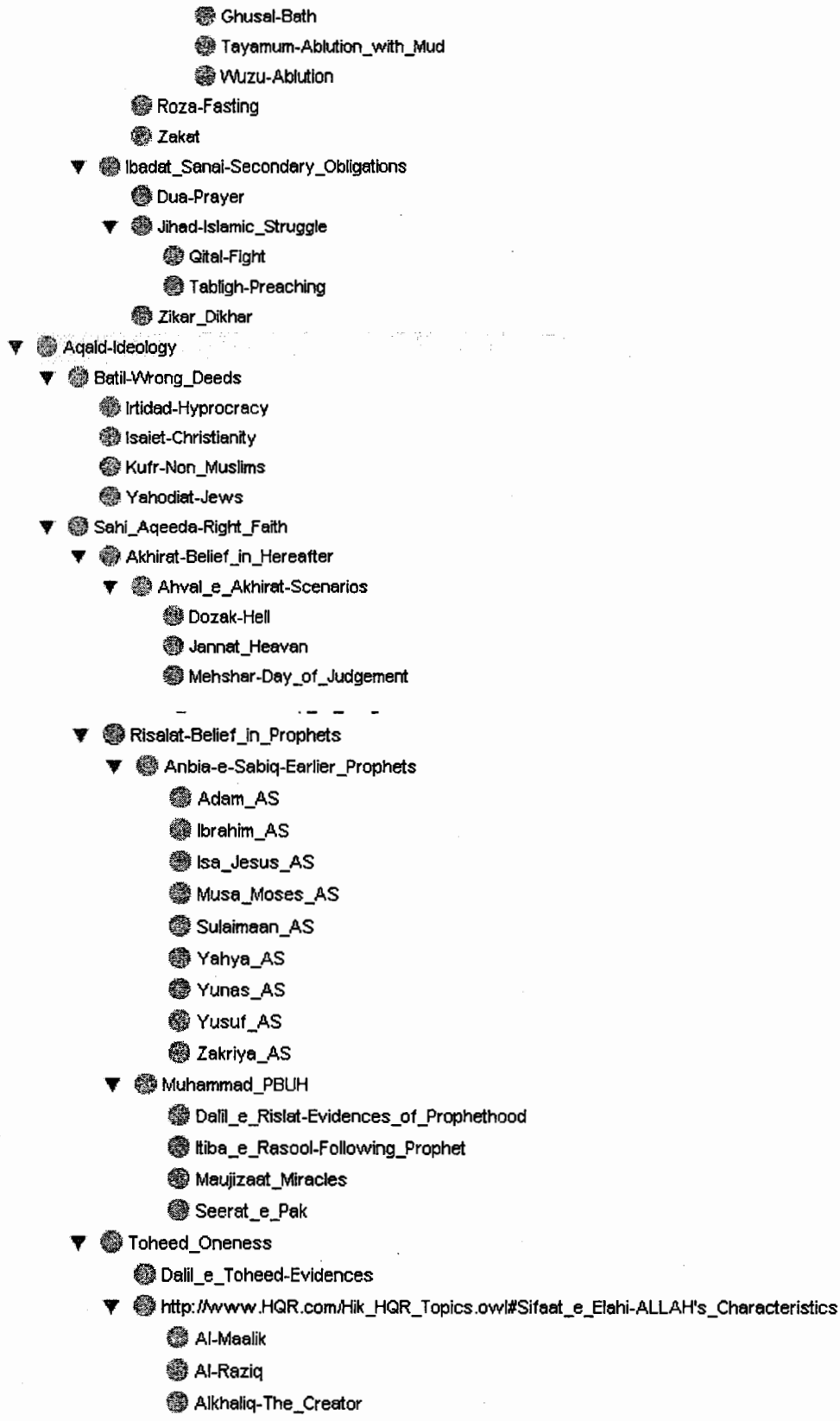



**Fig: 6.4 Taxonomy showing Subject matters of Holy Quran**

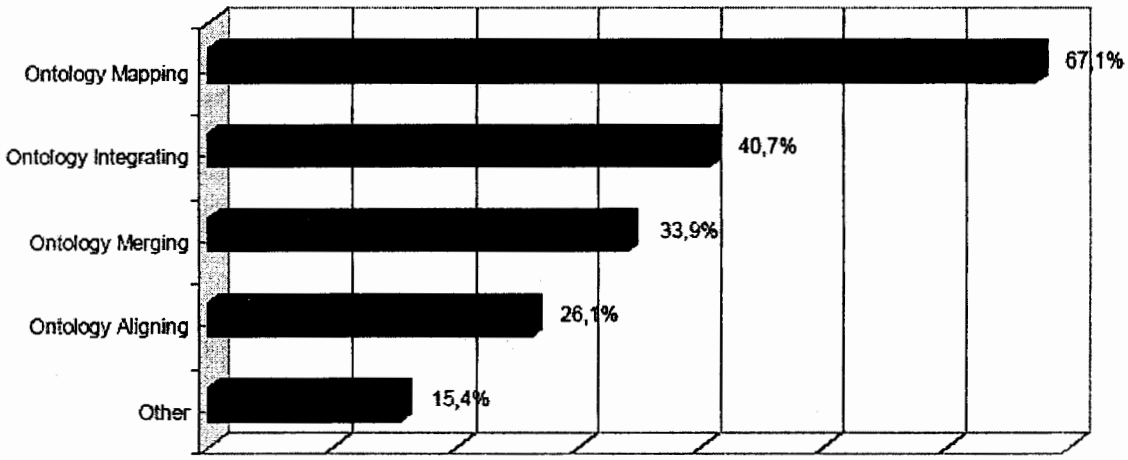
### 6.6.3 Recommendation about Techniques used with ontologies ( Ontology Merging, Ontology Integration and Ontology Mapping )

Let us first define three and then recommend. Ontology mapping is relating similar concepts or relations from different sources to each other by an equivalence relation. Ontology integration is building a new ontology reusing other available ontologies (assemble, extend, specialize), and ontology merging may be defined as merging different ontologies about the same subject into a single one that unifies all of them.

Ontology mapping is recommended for our project because different persons define the same concept mentioned in Holy Quran in different ways on the basis of their beliefs thus ontology mapping would be the right choice as different ontologies by different developers about the same domain needs to be mapped. This is very much applicable and recommended solution for difference of terms found in different translations of Holy Quran.

Ontology integration is also recommended as this way we will be able to extend our work into better integrated form. It also gives us option to extend our work to the concepts of Holy Quran to be applicable in similar concepts to those found in Hadith book etc.
Ontology merging is another recommended but precautions should be considered in this regard. Ontology merging is recommended for instance, if we create one domain ontology of animals , then we define another scientific taxonomy base ontology and merge the both then it will increase the level of semantic search work to its new heights. For instances, if Q/A base software uses question like: List the mammals mentioned in Holy Quran? Then our ontology can not answer but our ontology merged with scientific ontology will give us result.

The survey [102] gives support to our recommendation as follows:




**6.5 Techniques used with Ontology**

## 6.8 Dynamic Contextual Ontology and The Fourteen Primitive Actions

There has been unique endeavour in this regard that as it has revealed after exploration of Holy Quran for getting domain knowledge that most of the references are given with respect to some context and all these are very unique and very rarely relating to each other. So ontology based on this would be static one as for each context, we will have to make ontology specifically for that context.

Roger C. Schank established a conceptual structure to represent the conceptual information underlying a natural language sentence. It uses the basic actor-action-object framework. It was the intent that these structures have only one representation for one meaning, regardless of the semantic form of the sentence being represented. Actions were reduced to their basic parts so as to effect this. It was found that only fourteen basic actions were needed as building blocks by which all verbs can be represented. Each of these actions has a set of actions or states which can be inferred when they are present. The brief list and description of fourteen primitive acts, divided into five categories, for logical inference is as follows:

**Physical Acts**

1. Propel (Application of Force)
2. Move (To move body part)
3. Ingest (Take something inside)

4. Expel (Inside from you to out)
5. Grasp (to hold or capture something)

**Global Acts**

6. PTrans (Physical change in location)
7. ATrans (Abstract transfer like ownership etc)

**Instrumental Acts**

8. Smell
9. Speak
10. Look at
11. Listen to

**Mental Acts**

12. Conceptualization (Imagination)
13. MTrans Memory Transfer)
14. MBuild (Results as knowledge or as a belief)

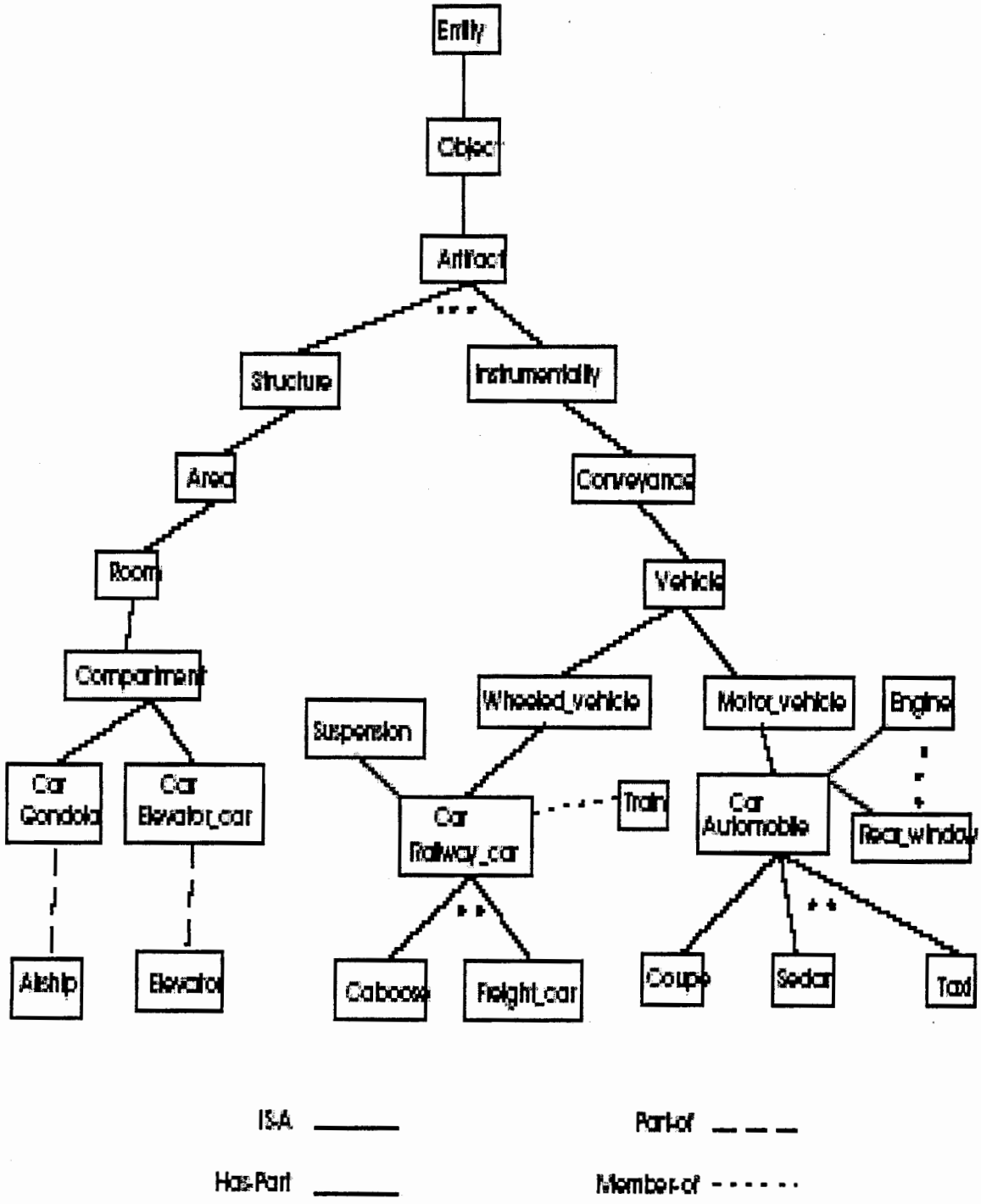
It has been found out that these fourteen actions classification does not fulfil the requirement for actions mentioned in Holy Quran. For instance, Slaughter is an action that can not fall in any of the fourteen categories mentioned above. So it is suggested that research should be done and new classification of primitive actions should be introduced.

## 6.8 Development of Quranic WordNet as Upper Ontology

### 6.8.1 Introduction to WordNet

WordNet is the product of a research project at Princeton University which has attempted to model the lexical knowledge of a native speaker of English[23]. The system has the power of both an on-line thesaurus and an on-line dictionary [23], and much more, as shown in fig 7.6 [106]. Information in WordNet is organised around logical groupings called sunsets. Each synset consists of a list of synonymous word forms and semantic pointers that describe relationships between the current synset and other synsets. A word form can be a single word or two or more words connected by underscores, referred to as collocations. The semantic pointers can be of a number of different types including:

- Hyponym/Hypernym (IS-A/ HAS A)
- Meronym/Holonym (Part-of / Has-Part)
- Meronym/Holonym (Member-of / Has-Member),
- Meronym/Holonym (Substance-of / Has-Substance).




**6.6 WordNet Extract for the concept 'car'**

### 6.8.2 Arabic WordNet

Arabic WordNet (AWN) is a multi-lingual concept dictionary with mappings between word senses in Arabic and those in the Princeton WordNet (English) v2.0. AWN supports searching and browsing of Arabic and English terms. It can be downloaded from [107]. It is an online thesaurus and can be used for querying from AWN [108]. The construction of an Arabic WordNet, following the development process of Princeton WordNet and Euro WordNet. It utilizes the Suggested Upper Merged Ontology as an interlingua to link Arabic WordNet to previously developed wordnets [109].

Arabic WordNet can be used if our proposed work can be extended from English language to Arabic language. It is also suggested that Urdu WordNet should also be developed inline with the development of AWN.

### 6.8.3 Development of Quranic WordNet and other WordNet.

It is proposed that as WordNet and AWN has been constructed as an upper merged ontology for the querying from the concerned language thus Quranic WordNet should be developed. Quranic WordNet will be collection of all sort of terms use din Holy Quran. The benefits of proposed work are obvious and immense. The Quranic WordNet will be collection of Quranic concepts and will be used to define the Islamic terms which are either not defined in WordNet and Arabic WordNet and if found then these concepts have defined by non-Muslims then surely we Muslims would like to have our own definition of concepts according to our faith and belief.

This development may be extended further to develop Islamic WordNet, a huge WordNet containing definition of concepts from all spheres of Islamic knowledge. This Islamic WordNet may comprise of Quranic WordNet, Hadith WordNet and Fiqh WordNet and many more. All these will be used as upper merged ontology and domain ontologies from all these sources will be created and would be used using techniques like ontology merging, ontology integration and ontology mapping at abstract level.

## 6.9 Framework

Let us finally discuss the proposed framework. There are various consideration level of proposed framework thus these are being proposed in context with certain already existing framework not only proposed but also implemented and are working nicely.

### 6.9.1 Proposed Semantic Search Architecture

Before actually coming to the point about proposed architecture, let us see what is the relations among domain of knowledge, document and user request in keyword based semantic search using ontology [52].

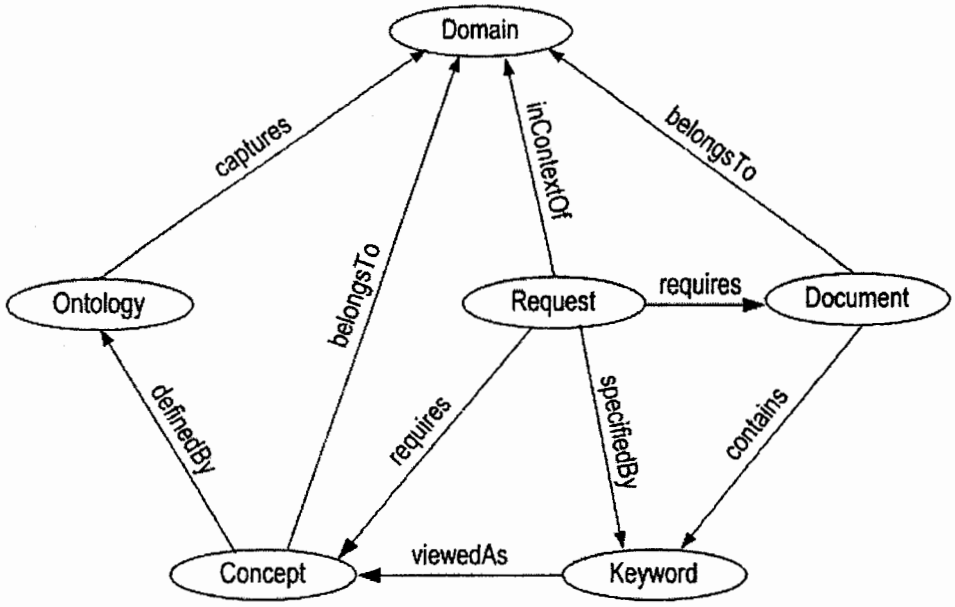
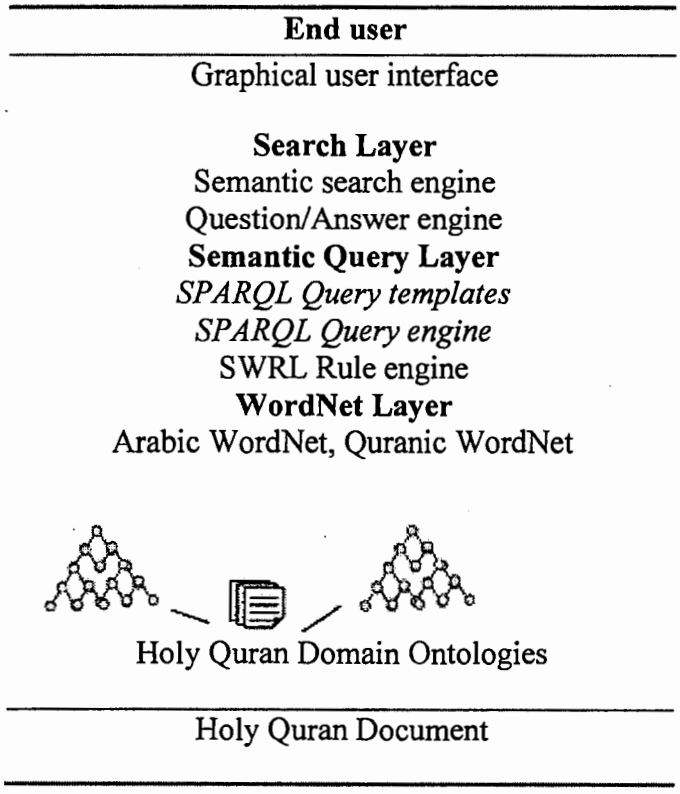



**Figure 6.7 The relations between domain, ontology, concept, keyword, document and user request in the semantic search system.**

On the basis of recommendations, the following simple figure depicts the different layers for proposed framework [110] The End user will be given highly user-friendly graphical user interface for semantic search as well as directly asking queries from Holy Quran. Then semantic query layer contains the usage of SPARQL query language. Certain query templates can also be designed. Also SWRL rules can also be made to formulise better reasoning as well. Next layer uses the concept of WordNet. Arabic and Quranic wordnets should be built to give dynamic behaviour and irrespective of translation of Holy Quran document being used for annotation. At the lower layer there is large knowledge base of domain ontologies based on Holy Quran text.




## 6.8 An Overview of the Semantic Search Architecture

### 6.9.2 Proposed Semantic Search Model

On the basis of framework proposed in [64], the framework has been proposed that explains the layers into better modules and in a sequence. It clearly depicts that Holy Quran document will be used for annotations and that will be directly linked with Semantic Web Rule Language that will be used to give us knowledge base consisting of classification and rules.

At the same level, ontology developers along with mutual consultation with domain experts will develop domain ontology. Consultation is mandatory because proper definition of concepts is ultimate goal and no second thought can be taken into consideration. The same care has been taken into account that concepts have only been defined as defined in Holy Quran only.

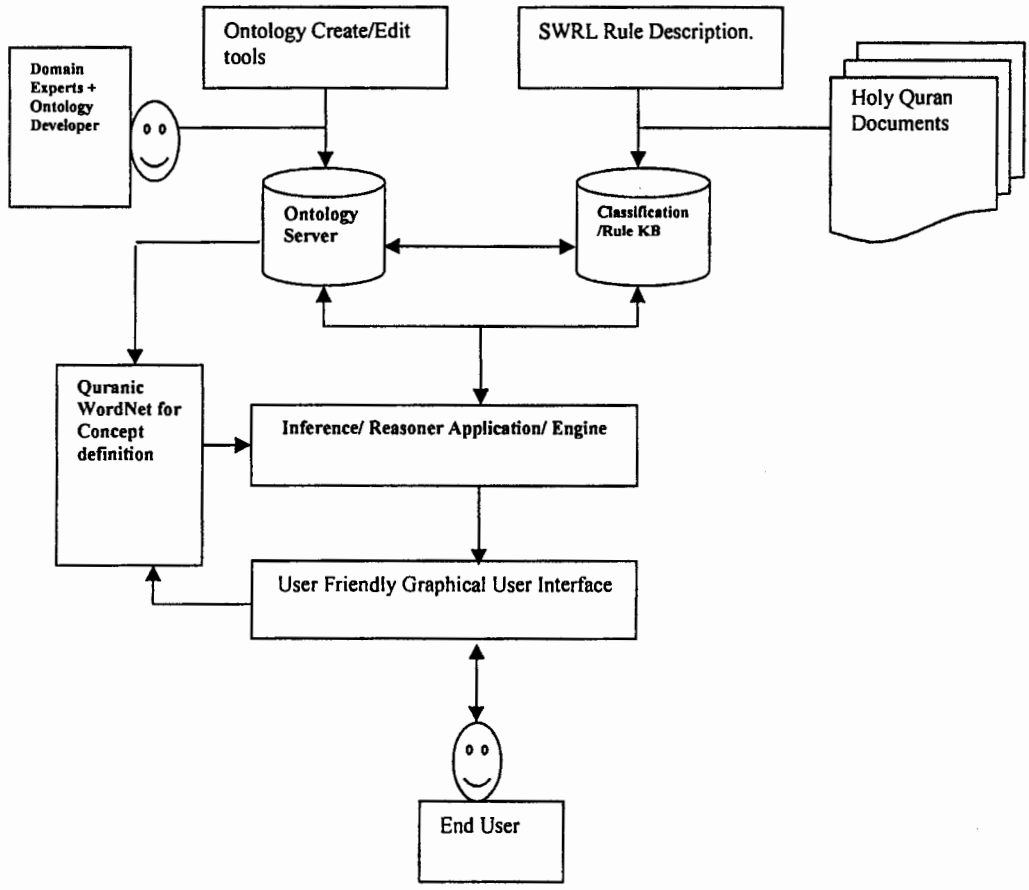



**Figure: 6.9 Ontology-based information retrieval Model**

Just to update the level of difference in efficiency achieved in the actual proposed can be analysed from following figure:

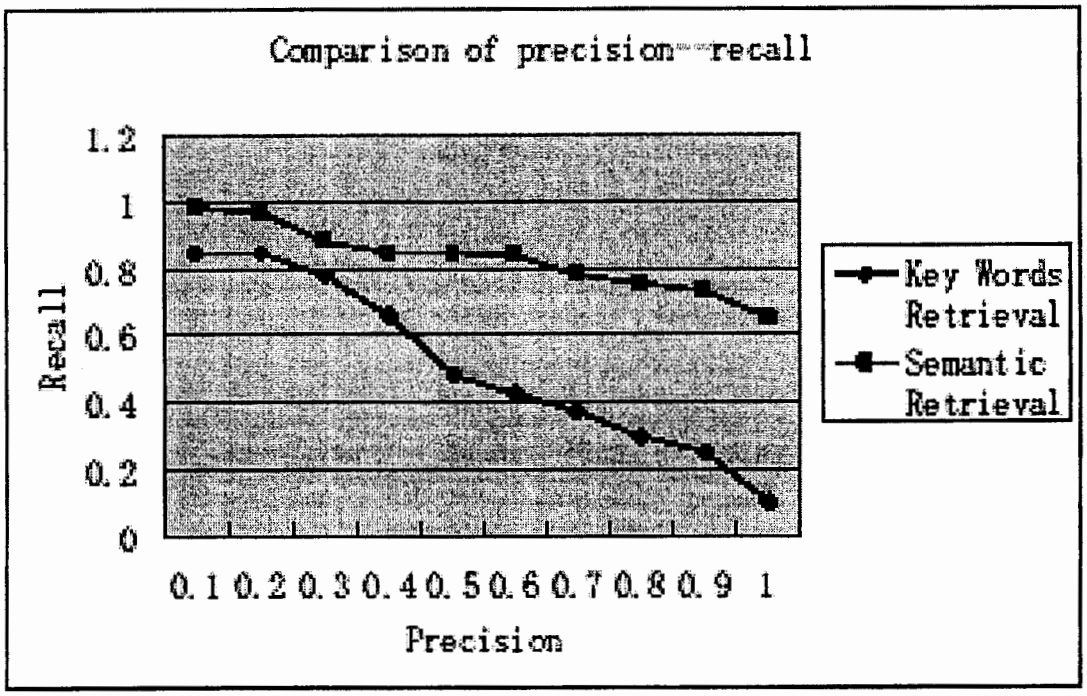



**Figure 6.10. Comparison between keyword and Semantic Search System**

### 6.9.3 Proposed Architecture for Semantic Search in Islamic Documents.

As in our recommendation that this model can be extended to Islamic Documents/Books of Hadith and Fiqh thus for such complex architecture, the following architecture is being proposed based on [52]. It is evident that query will be processed in the following order that when the user request a query. Then if it is simple keyword based query that it can be bypassed from wordnet, the sole purposeis not to complicate the processs into many definifiotn, as not required also less time will be consumed. Then parser will be used as the query can be broken down into token and parsing can be done. Next step is reasoner, as inference freasoner is based on ontologies then thus reasoner is linked with ontologies data/knowledge base which were created on the basis of Holy Quran text. As we want to do search the concepts from other Islamic sources on the basis of ontologies of Holy Quran thus other documents can be annotated for semantic search from those documents. As the number of documents are many then ranker has been proposed to rank the result on simple logic of count o f number of occurrences in all those documents except Holy Quran.

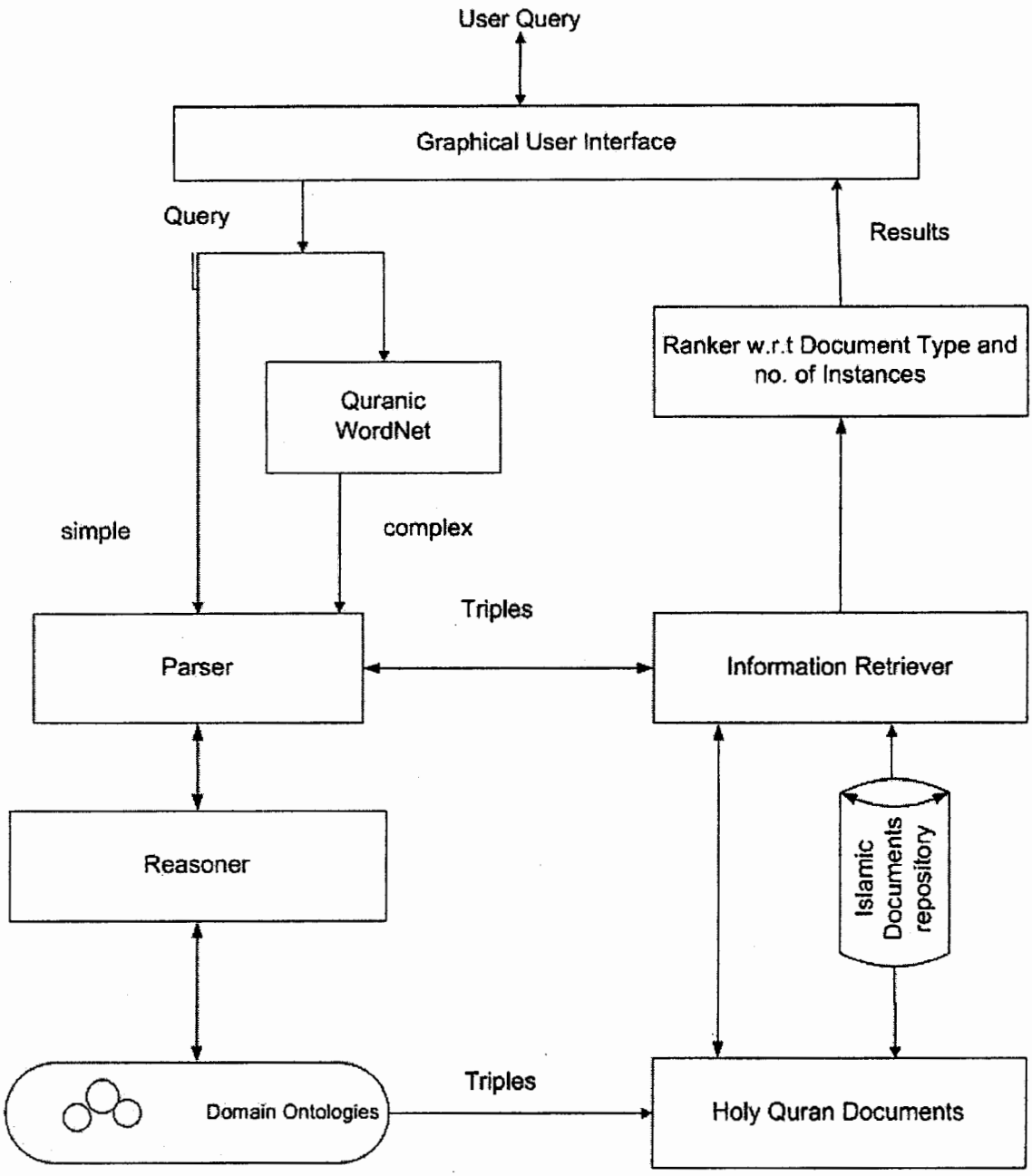



**Figure 6.11. Query Processing for Semantic Search System**

# CHAPTER NO. 7

# CONCLUSION AND OUTLOOK

# CHAPTER NO. 7

# CONCLUSION AND OUTLOOK

(Allah) Most Gracious! It is He Who taught the Quran.

(Surah Al-Rehman: 1-2)

---

## 7.1 Conclusion

In this work, the problems of keyword search have been addressed. A lot of work has been done in information retrieval domain of knowledge but no work has been done for efficient topic search from Holy Quran. Thus, this work proposes that the concepts of semantic web used can be used for implementation of semantic search in Holy Quran. It is basically exploratory work, thus initially, the causes of problems of search from Holy Quran has been explored. Then the extensive exploratory work has been carried out to get the knowledge in this domain. Then the recommendations regarding proposed model including proposed basic building block, ontology language, type of OWL, ontology editor, reasoners etc. Then the important concepts like upper ontology, ontology merging, ontology integration and ontology mapping has been discussed. WordNet is important structure giving dynamic touch to information retrieval from documents as well as web pages. But already existing WordNet can not be used thus it is proposed that Arabic WordNet and Quranic WordNet should be built. Inline with the basic recommendation, sample ontology for domain of animals mentioned in Holy Quran has been developed and SPARQL queries of various types has successfully been run that show the Inferenceability and ability of ontology that it can lead to overall big project for semantic search project for Holy Quran.

## 7.2 Contribution

The main contribution of this work is that it has laid foundation on the basis of which a huge work can be built. It can lead us to define knowledge structure on the basis of domain ontological recommendation not only for Holy Quran but also for the other Islamic sources of knowledge like Hadith etc. No work about semantic search from Islamic sources has been done so there is requirement that it should be carried out at very large group level and properly funded as well as properly managed because let the Islamic concepts should not defined by non-Muslims.

## 7.3 Out Look

It is not the end; it is actually the end of new beginning. I affirm to carry out these recommendation to new heights and develop a complete project of definition of Islamic concepts and attaining semantic search from Islamic sources of knowledge. It, as recommended, will be like development of. domain ontologies for Holy Quran. Then ontology techniques will be applied so that all these ontologies become manageable under large upper ontology. Then Quranic WordNet will provide it dynamic touch. Then this work will be enhanced to Hadith and Fiqh books and other significance sources of Islamic knowledge. May ALLAH help us to achieve this noble cause. Ameen.

# Appendices

# Appendix A

## Sample XML Code of the ontology

```
<?xml version="1.0"?>


<!DOCTYPE rdf:RDF [
   <!ENTITY p1 "http://www.HQR.com/" >
   <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
   <!ENTITY swrl "http://www.w3.org/2003/11/swrl#" >
   <!ENTITY swrlb "http://www.w3.org/2003/11/swrlb#" >
   <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
   <!ENTITY owl2xml "http://www.w3.org/2006/12/owl2-xml#" >
   <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
   <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
   <!ENTITY protege "http://protege.stanford.edu/plugins/owl/protege#" >
   <!ENTITY xsp "http://www.owl-ontologies.com/2005/08/07/xsp.owl#" >
]>


<rdf:RDF xmlns="&p1;Hik_Ontology_1.owl#"
    xml:base="&p1;Hik_Ontology_1.owl"
    xmlns:owl2xml="http://www.w3.org/2006/12/owl2-xml#"
    xmlns:p1="http://www.HQR.com/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"
    xmlns:swrl="http://www.w3.org/2003/11/swrl#"
    xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"
    xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:owl="http://www.w3.org/2002/07/owl#">
   <owl:Ontology rdf:about=""/>
   <owl:Class rdf:ID="ALLAH"/>
   <ALLAH rdf:ID="Allah">
      <Forbade_to_eat rdf:resource="#Swine_1"/>
      <Sent_down rdf:resource="#Quail_1"/>
   </ALLAH>
   <owl:Class rdf:ID="Animal">
      <rdfs:subClassOf rdf:resource="#Living_Creatures"/>
   </owl:Class>
   <owl:Class rdf:ID="Ass">
```

```
    <rdfs:subClassOf rdf:resource="#Animal"/>
  </owl:Class>
  <Ass rdf:ID="Ass_1">
    <Has_Usage rdf:datatype="&xsd;string">Ornament</Has_Usage>
    <Has_Usage rdf:datatype="&xsd;string">Ride</Has_Usage>
    <Has_Voice rdf:datatype="&xsd;string">Harshest</Has_Voice>
  </Ass>
  <owl:ObjectProperty rdf:ID="Becomes">
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Moses_Nation"/>
          <owl:Class rdf:about="#Object"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <owl:versionInfo rdf:datatype="&xsd;string">TODO:</owl:versionInfo>
    <rdfs:comment rdf:datatype="&xsd;string"
      >stick becomes snake ?</rdfs:comment>
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Snake"/>
          <owl:Class rdf:about="#Swine"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
  <owl:Class rdf:ID="Bee">
    <rdfs:subClassOf rdf:resource="#Animal"/>
  </owl:Class>
  <owl:Class rdf:ID="Bird">
    <rdfs:subClassOf rdf:resource="#Living_Creatures"/>
  </owl:Class>
  <owl:Class rdf:ID="Body_Part"/>
  <owl:Class rdf:ID="Branded_Horse">
    <rdfs:subClassOf rdf:resource="#Horse"/>
  </owl:Class>
  <Branded_Horse rdf:ID="Branded_Horse_2"/>
  <owl:ObjectProperty rdf:ID="Bring">
    <rdfs:range rdf:resource="#Calf-Ibrahim"/>
  </owl:ObjectProperty>
  <owl:Class rdf:ID="Calf">
    <rdfs:subClassOf rdf:resource="#Animal"/>
  </owl:Class>
  <owl:Class rdf:ID="Calf-Ibrahim">
```

```
    <rdfs:subClassOf rdf:resource="#Calf"/>
  </owl:Class>
  <owl:Class rdf:ID="Calf-Moses">
    <rdfs:subClassOf rdf:resource="#Calf"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="Can_not_reate">
    <rdfs:domain rdf:resource="#Man"/>
    <rdfs:range rdf:resource="#Fly"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="Carry">
    <rdfs:domain rdf:resource="#Ass"/>
    <rdfs:comment rdf:datatype="&xsd;string"
      >Again here Book from Object is to be modelled else make
DProperty.</rdfs:comment>
    <rdfs:range rdf:resource="#Object"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="Causes_to_Forget">
    <rdfs:domain rdf:resource="#Satan"/>
  </owl:ObjectProperty>
  <owl:Class rdf:ID="Children_of_Israel">
    <rdfs:subClassOf rdf:resource="#Peolple_Group"/>
  </owl:Class>
  <Children_of_Israel rdf:ID="Children_of_Israel_1">
    <Eat rdf:resource="#Quail_1"/>
  </Children_of_Israel>
  <owl:ObjectProperty rdf:ID="Choose">
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Bee"/>
          <owl:Class rdf:about="#Moses_Nation"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <rdfs:comment rdf:datatype="&xsd;string">Habitant ?</rdfs:comment>
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Calf-Moses"/>
          <owl:Class rdf:about="#Object"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
  <owl:Class rdf:ID="Crow">
    <rdfs:subClassOf rdf:resource="#Bird"/>
```

```
</owl:Class>
<owl:ObjectProperty rdf:ID="Deal_with">
  <rdfs:domain rdf:resource="#ALLAH"/>
  <rdfs:range rdf:resource="#Elephant_Owner"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="Dog">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:Class rdf:ID="Donkey">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="Eat">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Bee"/>
        <owl:Class rdf:about="#Children_of_Israel"/>
        <owl:Class rdf:about="#Man"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:comment rdf:datatype="&xsd;string">Fruit ?</rdfs:comment>
  <rdfs:range>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Goat"/>
        <owl:Class rdf:about="#Object"/>
        <owl:Class rdf:about="#Quail"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:range>
</owl:ObjectProperty>
<owl:Class rdf:ID="Elephant">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:Class rdf:ID="Elephant_Owner">
  <rdfs:subClassOf rdf:resource="#Peolple_Group"/>
</owl:Class>
<owl:Class rdf:ID="Ewe">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:Class rdf:ID="Ewe_Moses">
  <rdfs:subClassOf rdf:resource="#Ewe"/>
</owl:Class>
<owl:Class rdf:ID="Ewe_Moses-Brother">
  <rdfs:subClassOf rdf:resource="#Ewe"/>
```

```
</owl:Class>
<owl:Class rdf:ID="Fish">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<Fish rdf:ID="Fish1">
  <Lives_in rdf:datatype="&xsd;string">Sea</Lives_in>
  <Swallow rdf:resource="#Yunas"/>
</Fish>
<owl:Class rdf:ID="Fly">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="Forbade_to_eat">
  <rdfs:domain rdf:resource="#ALLAH"/>
  <rdfs:range rdf:resource="#Swine"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="Forget">
  <rdfs:range rdf:resource="#Fish"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="Frightens_from">
  <rdfs:domain rdf:resource="#Ass"/>
  <rdfs:range rdf:resource="#Lion"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="Frog">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="Give_example">
  <rdfs:domain rdf:resource="#ALLAH"/>
  <rdfs:range rdf:resource="#Gnat"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="Gnat">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:Class rdf:ID="Goat">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="Has_Body_part">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Bee"/>
        <owl:Class rdf:about="#Goat"/>
        <owl:Class rdf:about="#Horse"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:comment rdf:datatype="&xsd;string"
```

```
        >HorseBack 2b set
Belly to be set
Womb</rdfs:comment>
      <rdfs:range rdf:resource="#Body_Part"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:ID="Has_Brother">
      <rdfs:range rdf:resource="#Moses-Brother"/>
    </owl:ObjectProperty>
    <owl:DatatypeProperty rdf:ID="has_Characteristic">
      <rdfs:range>
        <owl:DataRange>
          <owl:oneOf>
            <rdf:List>
              <rdf:first rdf:datatype="&xsd;string">Fat</rdf:first>
              <rdf:rest>
                <rdf:List>
                  <rdf:first rdf:datatype="&xsd;string">Roasted</rdf:first>
                  <rdf:rest rdf:resource="&rdf;nil"/>
                </rdf:List>
              </rdf:rest>
            </rdf:List>
          </owl:oneOf>
        </owl:DataRange>
      </rdfs:range>
    </owl:DatatypeProperty>
    <owl:DatatypeProperty rdf:ID="has_Color">
      <rdfs:domain rdf:resource="#Calf-Moses"/>
      <rdfs:range>
        <owl:DataRange>
          <owl:oneOf>
            <rdf:List>
              <rdf:first rdf:datatype="&xsd;string">Yellow</rdf:first>
              <rdf:rest rdf:resource="&rdf;nil"/>
            </rdf:List>
          </owl:oneOf>
        </owl:DataRange>
      </rdfs:range>
    </owl:DatatypeProperty>
    <owl:DatatypeProperty rdf:ID="Has_Quantity">
      <rdfs:domain>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Ewe_Moses"/>
            <owl:Class rdf:about="#Ewe_Moses-Brother"/>
          </owl:unionOf>
        </owl:Class>
```

```
    </rdfs:domain>
    <rdfs:range>
      <owl:DataRange>
        <owl:oneOf/>
      </owl:DataRange>
    </rdfs:range>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="has_Sound">
    <rdfs:domain rdf:resource="#Calf-Moses"/>
    <rdfs:range>
      <owl:DataRange>
        <owl:oneOf>
          <rdf:List>
            <rdf:first rdf:datatype="&xsd;string">Lowing</rdf:first>
            <rdf:rest rdf:resource="&rdf;nil"/>
          </rdf:List>
        </owl:oneOf>
      </owl:DataRange>
    </rdfs:range>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Has_Strength">
    <rdfs:domain rdf:resource="#Object"/>
    <rdfs:comment rdf:datatype="&xsd;string">House 2b Set</rdfs:comment>
    <rdfs:range>
      <owl:DataRange>
        <owl:oneOf>
          <rdf:List>
            <rdf:first rdf:datatype="&xsd;string">Weakest</rdf:first>
            <rdf:rest rdf:resource="&rdf;nil"/>
          </rdf:List>
        </owl:oneOf>
      </owl:DataRange>
    </rdfs:range>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Has_Usage">
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Ass"/>
          <owl:Class rdf:about="#Horse"/>
          <owl:Class rdf:about="#Mule"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <rdfs:range>
      <owl:DataRange>
```

```
        <owl:oneOf>
          <rdf:List>
            <rdf:first rdf:datatype="&xsd;string">Ornament</rdf:first>
            <rdf:rest>
              <rdf:List>
                <rdf:first rdf:datatype="&xsd;string">Ride</rdf:first>
                <rdf:rest rdf:resource="&rdf;nil"/>
              </rdf:List>
            </rdf:rest>
          </rdf:List>
        </owl:oneOf>
      </owl:DataRange>
    </rdfs:range>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Has_Voice">
    <rdfs:domain rdf:resource="#Ass"/>
    <rdfs:range>
      <owl:DataRange>
        <owl:oneOf>
          <rdf:List>
            <rdf:first rdf:datatype="&xsd;string">Harshest</rdf:first>
            <rdf:rest rdf:resource="&rdf;nil"/>
          </rdf:List>
        </owl:oneOf>
      </owl:DataRange>
    </rdfs:range>
  </owl:DatatypeProperty>
  <owl:Class rdf:ID="Horse">
    <rdfs:subClassOf rdf:resource="#Animal"/>
  </owl:Class>
  <Horse rdf:ID="Horse_1">
    <Has_Usage rdf:datatype="&xsd;string">Ornament</Has_Usage>
    <Has_Usage rdf:datatype="&xsd;string">Ride</Has_Usage>
  </Horse>
  <owl:Class rdf:ID="Human_Being">
    <rdfs:subClassOf rdf:resource="#Living_Creatures"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="Hunt">
    <rdfs:domain rdf:resource="#Man"/>
    <rdfs:range rdf:resource="#Fish"/>
  </owl:ObjectProperty>
  <Prophet rdf:ID="Ibrahim"/>
  <rdfs:Literal rdf:ID="ibrahim"/>
  <owl:ObjectProperty rdf:ID="Inspire">
    <rdfs:domain rdf:resource="#ALLAH"/>
    <rdfs:range rdf:resource="#Bee"/>
```

```
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:ID="is_brought_to">
  <rdfs:domain rdf:resource="#Calf-Ibrahim"/>
  <rdfs:range>
    <owl:DataRange>
      <owl:oneOf>
        <rdf:List>
          <rdf:first rdf:datatype="&xsd;string">Angels</rdf:first>
          <rdf:rest rdf:resource="&rdf;nil"/>
        </rdf:List>
      </owl:oneOf>
    </owl:DataRange>
  </rdfs:range>
</owl:DatatypeProperty>
<owl:Class rdf:ID="Kahf_People">
  <rdfs:subClassOf rdf:resource="#Peolple_Group"/>
</owl:Class>
<owl:SymmetricProperty rdf:ID="Like">
  <rdf:type rdf:resource="&owl;ObjectProperty"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Ass"/>
        <owl:Class rdf:about="#Non_Believers"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <owl:inverseOf rdf:resource="#Like"/>
  <rdfs:range>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Dog"/>
        <owl:Class rdf:about="#Non_Believers"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:range>
</owl:SymmetricProperty>
<owl:Class rdf:ID="Lion">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:DatatypeProperty rdf:ID="Lives_in">
  <rdfs:domain rdf:resource="#Fish"/>
  <rdfs:range>
    <owl:DataRange>
      <owl:oneOf>
        <rdf:List>
```

```
                <rdf:first rdf:datatype="&xsd;string">Sea</rdf:first>
                <rdf:rest rdf:resource="&rdf;nil"/>
              </rdf:List>
            </owl:oneOf>
          </owl:DataRange>
        </rdfs:range>
      </owl:DatatypeProperty>
      <owl:Class rdf:ID="Living_Creatures"/>
      <owl:Class rdf:ID="Locust">
        <rdfs:subClassOf rdf:resource="#Animal"/>
      </owl:Class>
      <owl:ObjectProperty rdf:ID="Love">
        <rdfs:domain rdf:resource="#Man"/>
        <rdfs:range rdf:resource="#Branded_Horse"/>
      </owl:ObjectProperty>
      <owl:DatatypeProperty rdf:ID="Make_up_of">
        <rdfs:domain rdf:resource="#Calf-Moses"/>
        <rdfs:range>
          <owl:DataRange>
            <owl:oneOf>
              <rdf:List>
                <rdf:first rdf:datatype="&xsd;string">Ornaments</rdf:first>
                <rdf:rest rdf:resource="&rdf;nil"/>
              </rdf:List>
            </owl:oneOf>
          </owl:DataRange>
        </rdfs:range>
      </owl:DatatypeProperty>
      <owl:Class rdf:ID="Man">
        <rdfs:subClassOf rdf:resource="#Human_Being"/>
      </owl:Class>
      <Man rdf:ID="Man_1">
        <Love rdf:resource="#Branded_Horse_2"/>
      </Man>
      <Prophet rdf:ID="Moses"/>
      <rdfs:Literal rdf:ID="moses"/>
      <owl:Class rdf:ID="Moses-Brother">
        <rdfs:subClassOf rdf:resource="#Man"/>
      </owl:Class>
      <owl:Class rdf:ID="Moses_Nation">
        <rdfs:subClassOf rdf:resource="#Peolple_Group"/>
      </owl:Class>
      <owl:Class rdf:ID="Mosquito">
        <rdfs:subClassOf rdf:resource="#Animal"/>
      </owl:Class>
      <owl:Class rdf:ID="Mule">
```

```
    <rdfs:subClassOf rdf:resource="#Animal"/>
  </owl:Class>
  <Mule rdf:ID="Mule_1">
    <Has_Usage rdf:datatype="&xsd;string">Ornament</Has_Usage>
    <Has_Usage rdf:datatype="&xsd;string">Ride</Has_Usage>
  </Mule>
  <owl:DatatypeProperty rdf:ID="name">
    <rdfs:domain rdf:resource="#Human_Being"/>
    <rdfs:range rdf:resource="&rdfs;Literal"/>
  </owl:DatatypeProperty>
  <owl:Class rdf:ID="Non_Believers">
    <rdfs:subClassOf rdf:resource="#Peolple_Group"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="Not_Forbade_to_eat">
    <rdfs:domain rdf:resource="#ALLAH"/>
    <rdfs:range rdf:resource="#Goat"/>
  </owl:ObjectProperty>
  <owl:Class rdf:ID="Object"/>
  <owl:ObjectProperty rdf:ID="own">
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Elephant_Owner"/>
          <owl:Class rdf:about="#Kahf_People"/>
          <owl:Class rdf:about="#Moses-Brother"/>
          <owl:Class rdf:about="#Spider"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <rdfs:comment rdf:datatype="&xsd;string">House</rdfs:comment>
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Dog"/>
          <owl:Class rdf:about="#Elephant"/>
          <owl:Class rdf:about="#Ewe_Moses"/>
          <owl:Class rdf:about="#Ewe_Moses-Brother"/>
          <owl:Class rdf:about="#Object"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="Paneth_out">
    <rdfs:domain rdf:resource="#Dog"/>
    <rdfs:comment rdf:datatype="&xsd;string"
      >Here Tongue , do else make DP.</rdfs:comment>
```

```
      <rdfs:range rdf:resource="#Body_Part"/>
    </owl:ObjectProperty>
    <owl:Class rdf:ID="Peolple_Group">
      <rdfs:subClassOf rdf:resource="#Man"/>
    </owl:Class>
    <owl:Class rdf:ID="Prophet">
      <rdfs:subClassOf rdf:resource="#Man"/>
    </owl:Class>
    <Prophet rdf:ID="Prophet_4"/>
    <owl:Class rdf:ID="Quail">
      <rdfs:subClassOf rdf:resource="#Bird"/>
    </owl:Class>
    <Quail rdf:ID="Quail_1">
      <Sent_down_to rdf:resource="#Children_of_Israel_1"/>
    </Quail>
    <owl:Class rdf:ID="Satan">
      <rdfs:subClassOf rdf:resource="#Living_Creatures"/>
    </owl:Class>
    <owl:ObjectProperty rdf:ID="Scrath">
      <rdfs:domain rdf:resource="#Crow"/>
      <rdfs:comment rdf:datatype="&xsd;string"
        >Here value of Earth to be set actually if not than make it Data-
Property</rdfs:comment>
      <rdfs:range rdf:resource="#Object"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:ID="Sent_down">
      <rdfs:domain rdf:resource="#ALLAH"/>
      <rdfs:range>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Crow"/>
            <owl:Class rdf:about="#Frog"/>
            <owl:Class rdf:about="#Locust"/>
            <owl:Class rdf:about="#Quail"/>
            <owl:Class rdf:about="#Vermin"/>
          </owl:unionOf>
        </owl:Class>
      </rdfs:range>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:ID="Sent_down_to">
      <rdfs:domain rdf:resource="#Quail"/>
      <rdfs:range rdf:resource="#Children_of_Israel"/>
    </owl:ObjectProperty>
    <owl:DatatypeProperty rdf:ID="Smoothens">
      <rdfs:domain rdf:resource="#ALLAH"/>
      <rdfs:range>
```

```
        <owl:DataRange>
          <owl:oneOf>
            <rdf:List>
              <rdf:first rdf:datatype="&xsd;string">Ways for Bee</rdf:first>
              <rdf:rest rdf:resource="&rdf;nil"/>
            </rdf:List>
          </owl:oneOf>
        </owl:DataRange>
      </rdfs:range>
    </owl:DatatypeProperty>
    <owl:Class rdf:ID="Snake">
      <rdfs:subClassOf rdf:resource="#Animal"/>
    </owl:Class>
    <owl:Class rdf:ID="Spider">
      <rdfs:subClassOf rdf:resource="#Animal"/>
    </owl:Class>
    <owl:DatatypeProperty rdf:ID="Spread_Like">
      <rdfs:domain rdf:resource="#Locust"/>
      <rdfs:range>
        <owl:DataRange>
          <owl:oneOf>
            <rdf:List>
              <rdf:first rdf:datatype="&xsd;string"
                >Men come forth from the graves</rdf:first>
              <rdf:rest rdf:resource="&rdf;nil"/>
            </rdf:List>
          </owl:oneOf>
        </owl:DataRange>
      </rdfs:range>
    </owl:DatatypeProperty>
    <owl:ObjectProperty rdf:ID="Steals_something_from">
      <rdfs:domain rdf:resource="#Fly"/>
      <rdfs:range rdf:resource="#Man"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:ID="Stretch_out">
      <rdfs:domain rdf:resource="#Dog"/>
      <rdfs:comment rdf:datatype="&xsd;string">Paw</rdfs:comment>
      <rdfs:range rdf:resource="#Body_Part"/>
    </owl:ObjectProperty>
    <owl:ObjectProperty rdf:ID="Swallow">
      <rdfs:domain rdf:resource="#Fish"/>
      <rdfs:range rdf:resource="#Prophet"/>
    </owl:ObjectProperty>
    <owl:Class rdf:ID="Swine">
      <rdfs:subClassOf rdf:resource="#Animal"/>
    </owl:Class>
```

```
<Swine rdf:ID="Swine_1"/>
<owl:DatatypeProperty rdf:ID="Take_way_in">
  <rdfs:domain rdf:resource="#Fish"/>
  <rdfs:range>
    <owl:DataRange>
      <owl:oneOf>
        <rdf:List>
          <rdf:first rdf:datatype="&xsd;string">Water</rdf:first>
          <rdf:rest rdf:resource="&rdf;nil"/>
        </rdf:List>
      </owl:oneOf>
    </owl:DataRange>
  </rdfs:range>
</owl:DatatypeProperty>
<owl:ObjectProperty rdf:ID="Throws">
  <rdfs:comment rdf:datatype="&xsd;string">Stick, ?</rdfs:comment>
  <rdfs:range rdf:resource="#Object"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="Turn_into">
  <rdfs:domain rdf:resource="#ALLAH"/>
  <rdfs:range rdf:resource="#Moses_Nation"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="Use">
  <rdfs:domain rdf:resource="#Man"/>
  <rdfs:range rdf:resource="#Horse"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="Vermin">
  <rdfs:subClassOf rdf:resource="#Animal"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="Worship">
  <rdfs:domain rdf:resource="#Moses_Nation"/>
  <rdfs:range rdf:resource="#Calf-Moses"/>
</owl:ObjectProperty>
<rdfs:Literal rdf:ID="younas"/>
<Prophet rdf:ID="Yunas"/>
</rdf:RDF>
```

# APPENDIX B

## Sample code of Protégé-OWL Java Code

```
import impl;

import edu.stanford.smi.protege.model.FrameID;
import edu.stanford.smi.protegex.owl.model.*;
import edu.stanford.smi.protegex.owl.model.impl.OWLUtil;
import edu.stanford.smi.protegex.owl.javacode.ProtegeJavaMapping;

import java.util.*;

/**
 * Generated by Protege-OWL  (http://protege.stanford.edu/plugins/owl).
 *
 * @version generated on Wed Oct 07 02:25:41 ACT 2009
 */
public class Hik_HQR_Jave_Code {

   private OWLModel owlModel;

   static {
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Fish",
Fish.class, DefaultFish.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Animal",
Animal.class, DefaultAnimal.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Fly", Fly.class,
DefaultFly.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#ALLAH",
ALLAH.class, DefaultALLAH.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Prophet",
Prophet.class, DefaultProphet.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Man",
Man.class, DefaultMan.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Mosquito",
Mosquito.class, DefaultMosquito.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Calf-Moses",
Calf_Moses.class, DefaultCalf_Moses.class);
      ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Calf",
Calf.class, DefaultCalf.class);

ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Children_of_Israel",
Children_of_Israel.class, DefaultChildren_of_Israel.class);
```

```
ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Peolple_Group",
Peolple_Group.class, DefaultPeolple_Group.class);

ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Human_Being",
Human_Being.class, DefaultHuman_Being.class);

ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Living_Creatures",
Living_Creatures.class, DefaultLiving_Creatures.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Spider",
Spider.class, DefaultSpider.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Crow",
Crow.class, DefaultCrow.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Bird",
Bird.class, DefaultBird.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Moses-
Brother", Moses_Brother.class, DefaultMoses_Brother.class);

ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Branded_Horse",
Branded_Horse.class, DefaultBranded_Horse.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Horse",
Horse.class, DefaultHorse.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Satan",
Satan.class, DefaultSatan.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Calf-Ibrahim",
Calf_Ibrahim.class, DefaultCalf_Ibrahim.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Body_Part",
Body_Part.class, DefaultBody_Part.class);

ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Elephant_Owner",
Elephant_Owner.class, DefaultElephant_Owner.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Ewe",
Ewe.class, DefaultEwe.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Snake",
Snake.class, DefaultSnake.class);

ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Moses_Nation",
Moses_Nation.class, DefaultMoses_Nation.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Gnat",
Gnat.class, DefaultGnat.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Vermin",
Vermin.class, DefaultVermin.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Goat",
Goat.class, DefaultGoat.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Mule",
Mule.class, DefaultMule.class);
```

```
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Locust",
Locust.class, DefaultLocust.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Ass", Ass.class,
DefaultAss.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Object",
Object.class, DefaultObject.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Donkey",
Donkey.class, DefaultDonkey.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Bee",
Bee.class, DefaultBee.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Quail",
Quail.class, DefaultQuail.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Lion",
Lion.class, DefaultLion.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Kahf_People",
Kahf_People.class, DefaultKahf_People.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Swine",
Swine.class, DefaultSwine.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Elephant",
Elephant.class, DefaultElephant.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Ewe_Moses",
Ewe_Moses.class, DefaultEwe_Moses.class);

ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Non_Believers",
Non_Believers.class, DefaultNon_Believers.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Frog",
Frog.class, DefaultFrog.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Dog",
Dog.class, DefaultDog.class);
    ProtegeJavaMapping.add("http://www.HQR.com/Hik_Ontology_1.owl#Ewe_Moses-
Brother", Ewe_Moses_Brother.class, DefaultEwe_Moses_Brother.class);
  }

  public Hik_HQR_Jave_Code(OWLModel owlModel) {
    this.owlModel = owlModel;
  }

  public <X> X create(Class<? extends X> javaInterface, String name) {
    return ProtegeJavaMapping.create(owlModel, javaInterface, name);
  }


  public RDFSNamedClass getFishClass() {
    final String uri = "http://www.HQR.com/Hik_Ontology_1.owl#Fish";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
```

```
    }

    public Fish createFish(String name) {
        final RDFSNamedClass cls = getFishClass();
        if (name == null) {
            name = owlModel.getNextAnonymousResourceName();
        }
        return  new DefaultFish(owlModel, cls.createInstance(name).getFrameID());
    }

    public Fish getFish(String name) {
        RDFResource res =
owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
        if (res == null) {return null;}
        if (res instanceof Fish) {
            return (Fish) res;
        } else if (res.hasProtegeType(getFishClass())) {
            return new DefaultFish(owlModel, res.getFrameID());
        }
        return null;
    }

    public Collection<Fish> getAllFishInstances() {
        return getAllFishInstances(false);
    }

    public Collection<Fish> getAllFishInstances(boolean transitive) {
        Collection<Fish> result = new ArrayList<Fish>();
        final RDFSNamedClass cls = getFishClass();
        RDFResource owlIndividual;
        for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext();) {
            owlIndividual = (RDFResource) it.next();
            result.add(new DefaultFish(owlModel, owlIndividual.getFrameID()));
        }
        return result;
    }


    public RDFSNamedClass getAnimalClass() {
        final String uri = "http://www.HQR.com/Hik_Ontology_1.owl#Animal";
        final String name = owlModel.getResourceNameForURI(uri);
        return owlModel.getRDFSNamedClass(name);
    }

    public Animal createAnimal(String name) {
        final RDFSNamedClass cls = getAnimalClass();
```

```
    if (name == null) {
      name = owlModel.getNextAnonymousResourceName();
    }
    return  new DefaultAnimal(owlModel, cls.createInstance(name).getFrameID());
  }

  public Animal getAnimal(String name) {
    RDFResource res =
owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof Animal) {
      return (Animal) res;
    } else if (res.hasProtegeType(getAnimalClass())) {
      return new DefaultAnimal(owlModel, res.getFrameID());
    }
    return null;
  }

  public Collection<Animal> getAllAnimalInstances() {
    return getAllAnimalInstances(false);
  }

  public Collection<Animal> getAllAnimalInstances(boolean transitive) {
    Collection<Animal> result = new ArrayList<Animal>();
    final RDFSNamedClass cls = getAnimalClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext();) {
      owlIndividual = (RDFResource) it.next();
      result.add(new DefaultAnimal(owlModel, owlIndividual.getFrameID()));
    }
    return result;
  }


  public RDFSNamedClass getFlyClass() {
    final String uri = "http://www.HQR.com/Hik_Ontology_1.owl#Fly";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
  }

  public Fly createFly(String name) {
    final RDFSNamedClass cls = getFlyClass();
    if (name == null) {
      name = owlModel.getNextAnonymousResourceName();
    }
    return  new DefaultFly(owlModel, cls.createInstance(name).getFrameID());
```

```
    }

    public Fly getFly(String name) {
        RDFResource res =
owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
        if (res == null) {return null;}
        if (res instanceof Fly) {
            return (Fly) res;
        } else if (res.hasProtegeType(getFlyClass())) {
            return new DefaultFly(owlModel, res.getFrameID());
        }
        return null;
    }

    public Collection<Fly> getAllFlyInstances() {
        return getAllFlyInstances(false);
    }

    public Collection<Fly> getAllFlyInstances(boolean transitive) {
        Collection<Fly> result = new ArrayList<Fly>();
        final RDFSNamedClass cls = getFlyClass();
        RDFResource owlIndividual;
        for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext();) {
            owlIndividual = (RDFResource) it.next();
            result.add(new DefaultFly(owlModel, owlIndividual.getFrameID()));
        }
        return result;
    }


    public RDFSNamedClass getALLAHClass() {
        final String uri = "http://www.HQR.com/Hik_Ontology_1.owl#ALLAH";
        final String name = owlModel.getResourceNameForURI(uri);
        return owlModel.getRDFSNamedClass(name);
    }

    public ALLAH createALLAH(String name) {
        final RDFSNamedClass cls = getALLAHClass();
        if (name == null) {
            name = owlModel.getNextAnonymousResourceName();
        }
        return  new DefaultALLAH(owlModel, cls.createInstance(name).getFrameID());
    }

    public ALLAH getALLAH(String name) {
```

```
        RDFResource res =
owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
        if (res == null) {return null;}
        if (res instanceof ALLAH) {
            return (ALLAH) res;
        } else if (res.hasProtegeType(getALLAHClass())) {
            return new DefaultALLAH(owlModel, res.getFrameID());
        }
        return null;
    }

    public Collection<ALLAH> getAllALLAHInstances() {
        return getAllALLAHInstances(false);
    }

    public Collection<ALLAH> getAllALLAHInstances(boolean transitive) {
        Collection<ALLAH> result = new ArrayList<ALLAH>();
        final RDFSNamedClass cls = getALLAHClass();
        RDFResource owlIndividual;
        for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext();) {
            owlIndividual = (RDFResource) it.next();
            result.add(new DefaultALLAH(owlModel, owlIndividual.getFrameID()));
        }
        return result;
    }


    public RDFSNamedClass getProphetClass() {
        final String uri = "http://www.HQR.com/Hik_Ontology_1.owl#Prophet";
        final String name = owlModel.getResourceNameForURI(uri);
        return owlModel.getRDFSNamedClass(name);
    }

    public Prophet createProphet(String name) {
        final RDFSNamedClass cls = getProphetClass();
        if (name == null) {
            name = owlModel.getNextAnonymousResourceName();
        }
        return  new DefaultProphet(owlModel, cls.createInstance(name).getFrameID());
    }

    public Prophet getProphet(String name) {
        RDFResource res =
owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
        if (res == null) {return null;}
        if (res instanceof Prophet) {
```

```
        return (Prophet) res;
    } else if (res.hasProtegeType(getProphetClass())) {
        return new DefaultProphet(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<Prophet> getAllProphetInstances() {
    return getAllProphetInstances(false);
}

public Collection<Prophet> getAllProphetInstances(boolean transitive) {
    Collection<Prophet> result = new ArrayList<Prophet>();
    final RDFSNamedClass cls = getProphetClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new DefaultProphet(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}
```

# References

# References

[1] http://www.freedownloadmanager.org/downloads/Al_Quran_Explorer_27366_p/

[2] http://www.freedownloadmanager.org/downloads/Al_Quran_Explorer_27366_p/

[3] http://tanzil.info/

[4] http://www.topshareware.com/Qur'an-Viewer-(Koran)-download-14247.htm

[5] Afzalur Rahman, Subject Index of Quran, by, Published by: Islamic Publications (pvt) Ltd. Lahore Pakistan.

[6] Sarwar Hussain Khan Qadri Jhangiri, Jhangiri Qurani Ashariya, Published by: Maktaba Ashat Talemaat-ul-Quran, Karachi, Pakistan

[7] Maulana Abdul Majid Darya-abadi, Al- Hawanat fil Al-Quran, Hewanat-e-Qurani, Book Published by: Hijra International Publishers, Lahore

[8] http://www.islamicity.com/mosque/TOPICI.HTM

[9] http://www.islamicity.com/Quransearch/

[10] http://www.quran.org.uk/out.php?LinkID=12

[11] http://www.submission.org/quran/koran-index.html

[12] Peter Mika, Ontologies are us: A unified model of social networks and semantics. Vrije Univrsities, Amsterdam 1081HV Amsterdam, The Natherlands.

[13] Fernando Bobillo et al., Representation of context-dependent knowledge in ontologies: A model and an application. *Expert Systems with Applications 35 (2008) 1899-1908.* Retrieved on http://www.ScienceDirect.com/

[14] http://www. shaplus.com/exes/divinequotes/dqdivinequotes.html

[15] http://en.wikipedia.org/wiki/Semantics.

[16] Henry George Liddell, Robert Scott. "Semantikos". http://www.perseus.tufts.edu/cgi-bin/ptext?doc=Perseus%3Atext%3A1999.04.0057%3Aentry%3D%2393797

[17] "Semaino, Henry George Liddell, Robert Scott, An Intermediate Greek-English Lexicon, at Perseus.

[18] Otto Neurath (Editor), Rudolf Carnap (Editor), Charles F. W. Morris (Editor) (1955). International Encyclopedia of Unified Science. Chicago, IL: University of Chicago Press.

[19] http://en.wikipedia.org/wiki/Semantic_analysis_%28linguistics%29.

[20] http://en.wikipedia.org/wiki/Semantic_search.

[21] Robert M. Colombo, Ontology and the Semantic web, Volume 156, Frontiers in Artificial Intelligence and Applications. Published by: IOS Press, Washington, USA.

[22] Paul Buitelaar et al., Ontology-based information extraction and integration from heterogeneous data sources (2008). *International Journal of Human-Computer Studies 66(2008) 759-788.* Retrieved September 10, 2008 from ScienceDirect on htt://www.elsevier.com/locate/ijhcs/.

[23] http://wordnet.princeton.edu/

[24] http://www.guidedways.com/mobile/quranwordforword/download_quranw4w. php

[25] Evaluating the Effectiveness of Thesaurus and Stemming Methods in Retrieving Malay Translated Al-Quran Documents

[26] http://www.bibletechnologies.net

[27] http://www.islamware.com/

[28] Bayan: Arabic text database management syytem http://portal.acm.org/citation .cfm?id =93597 .93614.

[29] Y. Kotb, K. Gondow, and T. Katayama. "The SLXS Specification Language for Describing Consistency of XML Documents", Proc. of the Fourth Workshop on Information and Computer Science (WICS'2002), IEEE Comp. Soc., El-Damam, Saudi Arabia, pp. 289-304, March 2002.

[30] Y. Koth, K. Gondow, and T. Katayama. "The XML Semantics Checker Model", Proc. of the Third International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'02), Kanazawa, Japan, pp. 430-438, September 2002.

[31] Judith Dror, Dudu Shaharabani, Rafi Talmon, Shuly Winter. "Morphological Analysis of the Holy Quran ", Literary and Linguistic Computing, Vol. 19, No.4@ALLC 2004.

[32] Saidah saad, Naomie Salim, Nazlia Omar. "Kkeyphrase Extraction for Islamic Knowledge Ontology", ....

[33] Saidah saad, Naomie Salim. "Build Islamic Ontology based on Ontology Learning", Postgraduate annual Research Seminar 2007, (3-4 July 2007 )

[34] http://lists.arabeyes.org/archives/developer/2005/May/msg00055.html

[35] http://www.shaplus.com/free-quran-software/quran-mp3-software/QuranReciter/quranreciter-features.htm

[36] http://www.shaplus.com/free-quran-software/quran-multiple-translation-software/QuranTrans/qurantrans-free-download.htm

[37] http://www.eislamicsoftware.com/qurandir.htm

[38] http://www.freedownloadmanager.org/downloads/Al_Quran_Explorer_27366_p/

[39] http://www.allworldsoft.com/software/4-159-qur-an-viewer-koran.htm

[40] http://www.qurantoolkit.com/

[41] http://tanzil.info/

[42] http://www.globalquran.com/

[43] http://www.quranexplorer.com/

[44] http://www.understandquran.com/u/default.asp

[45] http://www.quran.com/

[46] http://www.yaquran.com/

[47] http://www.quransource.com/quran/

[48] www.quraaniclessons.com/

[49] http://www.hudainfo.com/QuranCD.htm

[50] www.topshareware.com/Qur'an-Viewer-(Koran)-download-14247.htm.

[51] DU Zhi-Qiang et al (2005), "The Research of the Semantic Search Engine based on the Ontology", Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[52] Wei-Dong Fand et al (2005), "Towards a Semantic Search Engine based on Ontologies", Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2009. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[53] Raja Loganantharaj and Vardarajan Badri Narayan (2006), *"SemPub: An Ontology Based Semantic Literature Retrieval"*, Proceedings of the 19th IEEE Symposium on Computer –Based Medical Systems (CBMS'06). Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[54] Yuanho Guo and Jeff Heflin (2007), Document-Centric Query Answering for the Semantic Web. 2007 IEEE/WIC/ACM International Conference on Web Intelligence.

Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[55] Tung-Cheng Hsien et al (2007), Query-based Ontology approach for Semantic Search, , 2007 International Conference on Machine Learning and Cybernetics. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[56] Shih-Fu Chang (2007), Recent Advances and Challenges of Semantic Image/Video Search, ICASSP 2007. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[57] Roman Pasichnyk, Anatoly Sachenko (2007), Semantic WEB-Search Developing by Problem-Oriented Ontology Means. IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications 6-8 September 2007, Dortmund, Germany. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[58] Dario Bonino et al (2007), Versatile RDF Representation for Multimedia Semantic Search, 19th IEEE International Conference on Tools with Artificial Intelligence. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[59] Zilan (Nancy) Yang (2007), A Practical Ontology-based concept Learning in MAS, Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[60] Hai Dong et al (2008), A Survey in Semantic Search Technologies, Second IEEE International Conference on Digital Ecosystems and Technologies (IEEE DEST 2008). Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[61] Guobing Zou (2008), An Ontology-based Methodology for Semantic Expansion Search. Fifth International Conference on Fuzzy Systems and Knowledge Discovery.

Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[62] Julian Szyma´nski and Włodzisław Duch (2008), Knowledge Representation and Acquisition for Large-Scale Semantic Memory. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[63] Sheng Qiuyan et al (2008), Measuring Semantic Similarity in Ontology and Its Application in Information Retrieval. 2008 Congress on Image and Signal Processing. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[64] Xiaohuan Zhang and Ying Guisheng (2008), Ontology-based Semantic Retrieval System. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[65] Holger Bast et al (2008). Semantic Full-Text Search with ESTER:Scalable, Easy, Fast. 2008 IEEE International Conference on Data Mining Workshops. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[66] Miriam Fernandez (2008), Semantic Search meets the Web. The IEEE International Conference on Semantic Computing. Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[67] Dulce Aguilar-Lopez (2008), Toward the semantic search by using ontologies. 2008 5th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE 2008). Retrieved January 29, 2009 from IEEE on World Wide Web:http://ieeexplore.ieee.com/

[68] R Guha (2003), Semantic Search, *WWW2003*, May 20-24, 2003, Budapest, Hungary. ACM 1-58113-680-3/03/0005. available on URL: http://www2003.org/cdrom/papers/refereed/p779/ess.html. Downloaded on August 3, 2009.

[69] Gerald Reif (2005) WEESA - Web Engineering for Semantic Web Applications. Available on http://www2005.org/cdrom/docs/p722.pdf

[70] http://infomesh.net/2001/swintro/

[71] Berners-Lee et al (2001), "The Semantic Web" Scientific American article. Avaliable on http://www.scientificamerican.com/article.cfm?id=the-semantic-web.

[72] http://www.altova.com/semantic_web.html

[73] Gruber ThomasR. A Translation Approach to Portable Ontology Specifications. Knowl Acquis 1993;5(2):199–220.

[74] DU Li Ding et al (2006), Using Ontologies in the Semantic Web: A Survey, .

[75] Coral Calero et al (), Ontologies in Software Engineering and Softeare Technology (Book), Calero · Ruiz Piattini (Eds.). Springer.

[76] http://en.wikipedia.org/wiki/Ontology_language.

[77] Grigoris Antoniou and Frank van Harmelen Web Ontology Language: OWL. Available on http://www.cs.vu.nl/~frankh/postscript/OntoHandbook03OWL.pdf

[78] Graham Klyne and Jeremy J. Carroll eds. Resource Description Framework (RDF):Concepts and Abstract Syntax. W3C Recommendation, 10 February 2004. http://www.w3.org/TR/2004/REC-rdf-schema-20040210/.

[79] http://infomesh.net/2001/swintro/#ontInference.

[80] http://www.w3.org/TR/owl-features/

[81] Holger Knublauch et al (), The Prot´eg´e OWL Plugin: An Open Development Environment for SemanticWeb Applications.

[82] Oscar Corcho et al (), Building legal ontologies with METHONTOLOGY and WebODE. Available on http://www.cs.man.ac.uk/ocorcho/documents/LawSemWeb2004_CorchoEtAl.pdf

[83] Dean Jones et al (1998), Methodology for Ontology Development. Available on http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.52.2437

[84] Hai H. Wang et al (),Frames and OWL Side by Side. Available on http://protege.stanford.edu/conference/2006/submissions/abstracts/7.2_Wang_Hai_Protege_conf.pdf.

[85] http://code.google.com/p/swoop/w/list

[86] http://www.mindswap.org/2004/SWOOP/

[87] York Sure et al (),OntoEdit: Collaborative Ontology Development for the Semantic Web, I. Horrocks and J. Hendler (Eds.): ISWC 2002, LNCS 2342, pp. 221–235, 2002. Springer-Verlag Berlin Heidelberg 2002.

[88] http://www.altova.com/semanticworks.html.

[89] Asuncion Gomez-Perez et al (), Ontological engineering , p115 available on http://www.springer.com/computer/information+systems/book/978-1-85233-551-9?cm_mmc=Google-_-Book%20Search-_-Springer.

[90] Jeremy J. Carroll et al (2004), Jena: implementing the semantic web recommendations, Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, available on : http://portal.acm.org/citation.cfm?id=1013367.1013381&coll=GUIDE&dl=GUIDE&CFID=46702712&CFTOKEN=71574806.

[91] Massimo Marchiori et al (), Web reasoning and rule systems , Book available on http://www.springer.com/computer/database+management+%26+information+retrieval/book/978-3-540-72981-5?cm_mmc=Google-_-Book%20Search-_-Springer.

[92] http://ksl.stanford.edu/software/JTP/.

[93] Dieter Fensel et al (1998), Ontobroker: The Very High Idea, In Proceedings of the 11th International Flairs Conference (FLAIRS-98) , Sanibal Island, Florida, May 1998.

[94] Dmitry Tsarkov and Ian Horrocks (0) FaCT++ Description Logic Reasoner: System Description, http://owl.man.ac.uk/factplusplus.

[95] Evren Sirin et al (),Pellet: A practical OWL-DL reasoner. Copyright © 2007 Published by Elsevier B.V. Avaliable on : http://www.sciencedirect.com.

[96] http://ontogenesis.ontonet.org/moin/AnimalBehaviourOntologyDevelopment.

[97] http://www.islam101.com/quran/quranPickthal/trans-pickthal.text.

[98] Standard Upper Ontology Working Group (SUO WG), IEEE P1600.1, available on http://suo.ieee.org/

[99] Natalya F. Noy and Deborah L. McGuinness (), Ontology Development 101: A Guide to Creating Your First Ontology.

[100] http://www.ai.rug.nl/ki/2006/practicum/eerstejaars/onto/animal.owl.

[101] http://metadata.net/WildNET/Animal.owl

[102] *Jorge Cardoso*, "The Semantic Web Vision: Where are We?" IEEE Intelligent Systems, September/October 2007, pp.22-26, 2007.

[103] http://protegewiki.stanford.edu/index.php/Protege4Features.

[104] Liyang Yu (), Introduction to Semantic Web and Web Services, Published by: Chapman and Hall/CRC Taylo & Francis Group, New York.

[105] Roger C. Schank (1973), The Fourteen Primitive Actions and their Inferences. Stanford Artificial Laboratory. Computer Science Department. Stanford University, USA.

[106] R.Richardson et al, Using WordNet as a Knowledge Base for Measuring Semantic Similarity between Words. School of Computer Applications. Dublin City University. Glasnevin, Dublin 9, IRELAND.

[107] http://sourceforge.net/projects/awnbrowser/

[108] http://www.lsi.upc.edu/~mbertran/arabic/awn/index.html.

[109] http://www.globalwordnet.org/AWN/.

[110] Yuangui Lei et al, SemSearch: A Search engines for the Semantic Web.