Abstract- Semantic Web technology enables us to specify metadata about things in the world and help us to relate this metadata using ontology databases and language. The core of the Semantic Web is Ontology, which is used to explicitly represent our conceptualizations. This paper is an effort to improve the relevancy of results in a search system for a particular domain by exploiting the domain knowledge captured in an OWL ontology using the protégé tool. We make ontological database from static relational database using protégé. The key ingredients that make up an ontology are a vocabulary of basic terms, semantic interconnections, simple rules of inference and some logic for a particular topic. We extract semantic meaning from query entered by the user using tool and then we map this meaning into ontological database. To achieve semantic search, a search engine is needed which can interpret the meaning of a user's query and the relations among the concepts that a document contains with respect to a particular domain. Protégé is an extensible and customizable tool for constructing ontologies and for developing applications that use these ontologies. After that we display this data on user screen. The output of this project is to provide appropriate result to user for enter his queries.

Keywords- Logical Consequences, Semantic Web, Ontology, Ontology Web Language, Protégé.

I. INTRODUCTION

In this project, we are developing and providing a structured semantic database, which will be able to give proper response towards the search query from user. We are constructing, we propose a forward chaining knowledge base and reasoning engine, supporting OWL, with the ability to perform non-monotonic, retractable inference. Within our project, the knowledge over the collection can be collected in several ways. Our aim is to create and implement Ontological database, for managing such knowledge based collection to response numbers of users search queries. We want to reduce the contention on current system & allow maximum numbers of user to access search tools with quick and proper required response on search query. We are focused to coming up with solutions that serve customer requirements today and anticipate future needs.

II. EXISTING SYSTEM

Currently, most web search engines are however based on purely statistical techniques. Users express their information need via simple queries based on keywords, but these techniques are not able to find out the exact meaning of a query which is entered by user. These engines usually provide a huge number of answers, many of which are completely irrelevant, whereas some of the more interesting answers are not found. [2] Some answers are also not in the proper format. Though, the current information retrieval systems provide information for each domain but still it is difficult to provide the appropriate information to users what they seek. There are two fundamental issues with current information retrieval systems; information excess and information irrelevancy. [1] Also the searching systems cannot be able to distinguish the meaning of the statement in terms of the user’s requirement.

III. PROPOSED SYSTEM

Ontology is a knowledge representation technique for the semantic web.[1] Ontologies are not intended just for storing knowledge about the subject domain but can be used by semantic web agent for inference, data integration decision making etc. So our aim to deriving logical consequences using OWL language, make ontological database from static relational database using protégé tool, and extract semantic meaning from query entered by the user, map this meaning into ontological database and after that display the result on user screen. [2]

In this paper, we discuss the details of an OWL based system that works on domain and exploits the domain ontology. We limit our discussion to search domain related query only.

IV. BASIC TERMS OVERVIEW

A. Logical Consequences
Relationship between statements that holds true when one logically "follows from" one or more others. Logical
consequence is taken to be both necessary and formal with examples explicated using models and proofs. [6] A sentence is said to be a logical consequence of a set of sentences, for a given language, if and only if, using logic alone (i.e. without regard to any interpretations of the sentences) the sentence must be true if every sentence in the set were to be true. For example, Sachin is not at work is a logical consequence of Sachin is not both at home and at work and Sachin is at home.

B. Semantic Web
The Semantic Web is a web that is able to describe things in a way that computers can understand. Different from normal web technologies – More intelligent than formal web technologies. [3]

• The Beatles was a popular band from Liverpool.
• John Lennon was a member of the Beatles.
• "Hey Jude" was recorded by the Beatles.

Sentences like the ones above can be understood by people. But how can they be understood by computers?

Statements are built with syntax rules. The syntax of a language defines the rules for building the language statements. But how can syntax become semantic? This is what the Semantic Web is all about. Describing things in a way that computers applications can understand it. The Semantic Web is not about links between web pages. The Semantic Web describes the relationships between things (like A is a part of B and Y is a member of Z) and the properties of things (like size, weight, age, and price). The aim of the Semantic Web is to allow much more advanced knowledge management systems:

• Knowledge will be organized in conceptual spaces according to its meaning. [3]
• Automated tools will support maintenance by checking for inconsistencies and extracting new knowledge.
• Keyword-based search will be replaced by query answering: requested knowledge will be retrieved, extracted, and presented in a human-friendly way. [5]
• Query answering over several documents will be supported.
• Defining who may view certain parts of information (even parts of documents) will be possible.

C. Ontology Web Language
For the web, Ontology is about the exact description of web information and relationships between web information. OWL is a part of the "Semantic Web Vision" - a future where: Web information has exact meaning. Web information can be processed by computers, and Computers can integrate information from the web. OWL was designed to provide a common way to process the content of web information (instead of displaying it). [1] OWL was designed to be read by computer applications (instead of humans). OWL and RDF are much of the same thing, but OWL is a stronger language with greater machine interpretability than RDF. OWL comes with a larger vocabulary and stronger syntax than RDF.

OWL has three sublanguages:

• OWL Lite
• OWL DL (includes OWL Lite)
• OWL Full (includes OWL DL)

By using XML, OWL information can easily be exchanged between different types of computers using different types of operating system and application languages. OWL became a W3C (World Wide Web Consortium) Recommendation in February 2004. A W3C Recommendation is understood by the industry and the web community as a web standard. A W3C Recommendation is a stable specification developed by a W3C Working Group and reviewed by the W3C Membership.

D. Protégé
Protégé is a free, open source ontology editor and knowledge based framework. The Protégé platform supports modeling ontologies via a web client or a desktop client. Protégé ontologies can be developed in a variety of formats including OWL, RDF(S), and XML Schema. Protégé is based on Java, is extensible, and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development. The Protégé-Frames editor enables users to build and populate ontologies that are frame-based, in accordance with the Open Knowledge Base Connectivity protocol (OKBC). The Protégé-OWL editor enables users to build ontologies for the Semantic Web, in particular in the W3C’s Web Ontology Language (OWL). [2]

E. SPARQL
An RDF graph is a set of triples; each triple consists of a subject, a predicate and an object. SPARQL is a query language for getting information from such RDF graphs. It provides facilities to: 1. Extract information in the form of URIs, blank nodes and literals. 2. Extract RDF sub graphs. 3. Construct new RDF graphs based on information in the queried graphs. [6] The SPARQL query language is based on matching graph patterns. The simplest graph pattern is the triple pattern, which is like an RDF triple but with the possibility of a variable in any of the subject, predicate or object positions. Combining these gives a basic graph pattern, where an exact match to a graph is needed to fulfill a pattern. [7]

F. XML (Extensible Markup Language)
XML allows mark-up and structuring arbitrary content by means of nested, attributed elements. The structuring has no
particular semantics to indicate what the structure means. XML plays the role of just a syntax carrier and this layer corresponds to a basic syntax layer. [1]

G. RDF (Resource Definition Framework) Model

RDF allows encoding, exchange, and reuse of structured metadata. In principle, information is represented by generic means, say by directed partially labelled graphs that may be serialized using XML. Contrary to XML, RDF allows assigning global identifiers to such information resources and allows one resource document to refer to and extend statements made in other resource documents.

V. SYSTEM ARCHITECTURE AND FUNCTIONS

The Deriving Logical Consequences project is nothing but the efficient way to find the accurate result of query fired by the user. The domain on which the query is asked is the college department. The system helps to find out the efficient relative information about department which is actually not provide by the current search mechanisms. If you give any word or sentence to the system it will not search that word or the sentence by the keyword matching it will give the proper relative result. The complete overview of the system is as shown in the overview diagram below.

![System Architecture Diagram](image)

The product to be developed has interactions with the users through the GUI (Graphics user interface) only: Anyone or Members who want to search information about college are free to use the system. Fired query is handle by the Query handler and send to the protégé. Protégé-OWL editor enables users to build ontology for the Semantic Web. Using this tool the mapping between local ontology to reference ontology is done on that basis the relative information is retrieve from the database. Final result is send on the user interface as a result.

![Mapping Process Diagram](image)

VI. SYSTEM MODELLING

As part of the system requirements and design activities, the mash up created in this work consists of several components that are associated, namely: Component to design ontology model. This component is used to design ontology model based on domain expert. In modeling the ontology we used an open source tool known as a Protégé. Result from the design model is structured in the taxonomy hierarchy. The process of designing the model can only be done by a domain expert (in our case a developer) in other words people who really control a domain. The role of domain expert is to create a taxonomy hierarchy (super class-subclass), property and instance by using the Protégé. Once a complete ontology model is created, the ontology models are stored in XML file. [2]

In this work we decided to transform the OWL XML files from Protégé directly into the database. Previously we decided to use RDF [4, 5, 6, 7, and 8], however, it was difficult to classify the sentence format and parse the query due to the fact that the format is in subject, predicate, and object. In the XML format, we could decide our own format, and we could parsed the title and the author information. XML format makes it clearer and easy to create the query. [1]

VII. ONTOLOGY DESIGN

Ontology design provides information of the stages in the development of ontology and explains the components needed in a representation of information. Stages in the development of ontology are as follows:

- Determining the domain. Domain ontology is College Department domain.

```xml
<rdf:RDF
  xmlns:owl = "http://www.w3.org/2002/07/owl#"
  xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs = "http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xsd = "http://www.w3.org/2001/XMLSchema#">
  <owl:Ontology rdf:about=""/>
</rdf:RDF>
```
An example OWL ontology

- Defining ontology class and arrange them in taxonomy hierarchy (subclass-superclass) using the process of top down starts with defining the common concept followed by more specific concept.

- Defining the slot or property.

- Defining facets in the slot or axiom in properties. Properties have specific domain and range. Properties connect instance in domain and instance in range.

- Creating instance.

- Filling the instance’s slot value.

VIII. CONCLUSION

The proposed system is an effort to retrieve relevant information in a domain using domain specific knowledge captured in the form of OWL ontology. OWL is the proposed standard for Web ontologies. It allows us to describe the semantics of knowledge in a machine-accessible way. Formal semantics and reasoning support is provided through the mapping of OWL on logics. Predicate logic and description logics have been used for this purpose. While OWL is sufficiently rich to be used in practice, extensions are in the making. They will provide further logical features, including rules.

REFERENCES


