International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-4 Issue-2, December 2014

An Approach for Semantic Information Retrieval from Ontology in Computer Science Domain

Ritika Bansal, Sonal Chawla

Abstract- Ontology plays a pivotal role in exchange of information, use of knowledge and its re-use, shared and common understanding of a domain specific knowledge that can be communicated between people and across application systems which is the goal of semantic web. Ontology is used to capture knowledge about any domain of interest with the objective of incorporating the machine understandable data on the current human-readable web. Ontology is a broad term including a wide range of activities, complexities and issues in which Ontology Development is one of the most fundamental and significant concern[1]. There may be various methodologies or tools for ontology development. This paper has three main objectives. Firstly, it considers the computer science domain and demonstrates the development of Ontology in this domain using Protégé 3.4 Editor. Secondly, this paper focuses on the techniques and query language SPARQL for data retrieval from Ontology. Thirdly, this paper will discuss an approach for retrieving information from Ontology through natural language queries by demonstrating the layout of IRSCSD (Information retrieval system for computer science domain).

Keywords: Ontology, RDF, Semantic Searching, SPARQL, NLQ, Protégé, Jena API, Query.

I. Introduction

The current web is based on HTML which is not able to be exploited by information retrieval techniques and hence processing of information on web is mostly restricted to manual keyword searches which results in irrelevant information retrieval. So, there is a need of an intelligent and meaningful web which is efficient in relevant information retrieval. This limitation may be overcome by a new web architecture known as semantic web which is an intelligent and meaningful web. In semantic search system, the concept of ontology is used to search results by contextual meaning of input query instead of keyword matching [2]. Ontology provides a knowledge-sharing framework that supports the representation and sharing of domain knowledge [3]. An increasing number of Ontology are being developed, and their reuse and sharing offers several benefits. One important benefit is that we can significantly save time and effort by reusing existing Ontology instead of building new ones every time.

Manuscript published on 30 December 2014.
***** Correspondence Author (s)
Ritika Bansal, Research Scholar, Department of Computer Science and Applications, Panjab University & Assistant Professor, M.C.M. D.A.V. College for Women, Chandigarh, India
Sonal Chawla, Associate Professor & Chairperson, Department of Computer Science and Applications, Panjab University, Chandigarh, India

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

Another advantage is that heterogeneous systems and resources can interoperate seamlessly by sharing a common knowledge [4]. In the proposed IRSCSD system, the semantic information is extracted from users input query in natural language. Using this semantic information, query is translated into a format which can directly retrieve information from ontology. In the proposed system, input query in natural language is converted into a SPARQL query which is a query language for RDF based database. SPARQL query is then fired on to the RDF database and accesses the relevant information [5].

II. Ontology and Information Retrieval

2.1 Ontology

Ontology in Computer Science is a way of representing a common understanding of a domain. It allows for machine-understandable semantics of data, and facilitates the search, exchange, and integration of knowledge. Ontology is different from traditional keyword-based search engines in that they are metadata, able to provide the search engine with the functionality of semantic matching. Ontology is able to search more efficiently than traditional methods [6]. Typically, ontology consists of hierarchical descriptions of important concepts in a domain and the descriptions of the properties of each concept. Ontology, which is the heart of semantic web, with concept instantiations serves as a domain knowledge base i.e. semantic web technology information center. The ontology is designed for the system to incorporate the domain information in the form of instances and data type values, classes and object properties i.e. to model the knowledge domain [7]. Ontology allows for machine-understandable semantics of data, and facilitates the search, exchange, and integration of knowledge. Ontology is always constructed with a certain task in mind; this task focus restricts the content and structure of the ontology. Many tools have been developed for implementing metadata of Ontology. Ontology tools can be applied to all stages of the ontology life cycle including the creation, population, implementation, and maintenance of ontology (Polikoff, 2003) [8].

Ontology can be developed for various domains like: Information system design, Biomedical, Media, Linguistics, Business, Travel, Web Services, Logic Puzzles, Engineering, Education, Construction, Entertainment, Government, Home security etc. If the domain (or part of it) changes, the conceptualization must also change and consequently the ontology that represents this mini-world changes too.

© Copyright: All rights reserved.

Ontological Engineering refers to the set of activities that concerns the ontology development process, the ontology life cycle, and the methodologies, tools and languages for building Ontology. Ontology development is a complex and largely domain-oriented process that can be benefited from tool support.

2.2 SPARQL

As more data is being stored in RDF formats, a need has arisen for a simple way to locate specific information. SPARQL, a powerful query language fills that space, making it easy to find the data being searched in the RDF haystack [9] .SPARQL stands for SPARQL Protocol and RDF Query Language. It is the standardized query language for RDF, the same way SQL is the standardized query language for relational databases. A SPARQL query consists of a set of triples where the subject, predicate and/or object can consist of variables. The idea is to match the triples in the SPARQL query with the existing RDF triples and find solutions to the variables. A SPARQL query is executed on a RDF dataset. SPARQL Protocol and RDF Querying Language are used as the channel that accomplishes the retrieval of that information. SPARQL builds on previous RDF query languages such as rdfDB, RDQL and SeRQL.

III. Proposed Semantic Web based IRSCSD architecture

In the proposed Information retrieval system for computer science domain (IRSCSD) system, interface will accept natural language queries to extract data from domain specific Ontology and retrieve the desired results. Ontology is a RDF/OWL based database whose query language is SPARQL, so there is a need of conversion from natural language query to SPARQL query to retrieve data from Ontology. In the proposed system, input query in natural language is converted into a SPARQL query which is a query language for RDF based database. SPARQL query is then fired on to the RDF database and accesses the relevant information. Thus, the proposed semantic web based IRSCSD architecture is comprised of three main phases:

Phase1: Ontology building

Phase2: NLQ to SPARQL Conversion

Phase3: Running SPARQL query on Ontology and fetching desired results.

So, the High Level Design of IRSCSD Architecture showing its three main phases is:

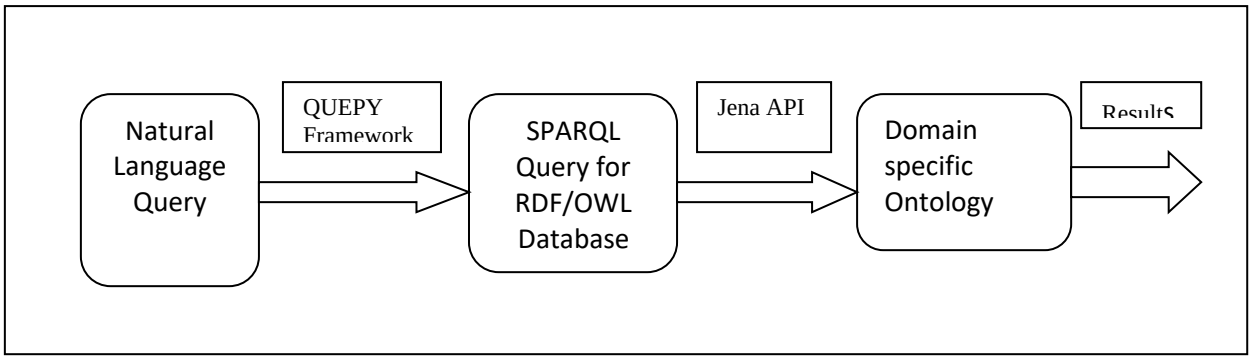


Figure 1: High level design of IRSCSD architecture.

The high level design of proposed architecture shows the three main phases of the system. The core part of the design is domain specific Ontology building. Interface accepts queries in natural language which are converted into SPARQL query language through Python based QUEPY framework. Then the converted SPARQL queries are being fired to the Ontology through Apache's Jena API to fetch the results.

The entities are extracted from Ontology to build the lexicon library and their synonyms are added using Word-Net to expand the Lexicon library [10]. Parsing and disambiguation of the natural language query is done after the recognition of named entities and parse tree is formed. Then the translator will match the query terms with Ontology concepts and properties and triples are generated. After the integration of theses triples, SPARQL query is generated. This generated SPARQL query is fired to the domain specific Ontology and get the desired results at the interface. Thus, when the architecture is seem at a detailed level and each phase is expanded into its components, low level design of IRSCSD architecture is:

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

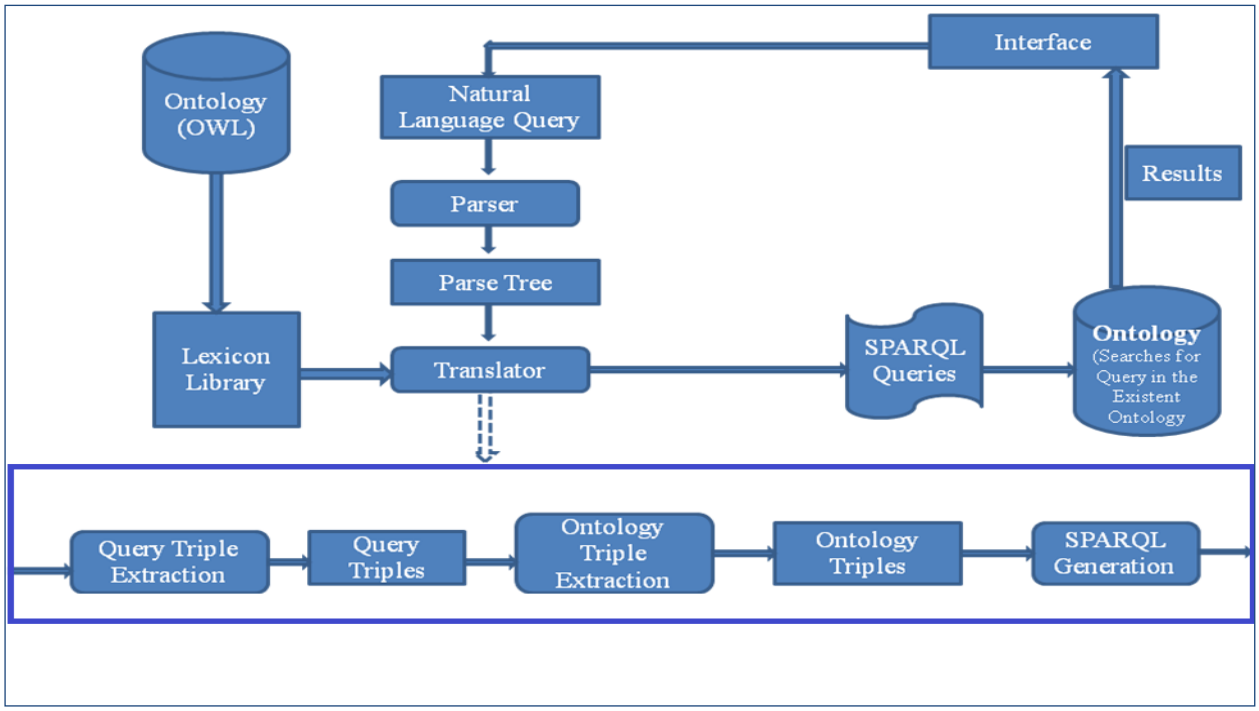


Figure 2: Low level design of IRSCSD architecture.

3.1 PHASE I: Ontology Building

Ontology development is a complex and largely domain-oriented process that can be benefited from tool support. In (Bansal.R.,Chawla.S.,2013)[7] ,various tools for ontology development are compared based on certain features such as modeling features/limitations, base language, web support and use, import/export format, graph view, consistency checks, multi-user support etc. It was found that Protégé tool 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 [11]. Protégé is a free and open- source ontology editor and framework for building intelligent systems [12]. Protégé is a tool which creates data into RDF data format. We have used Protégé_3.4.8 tool to create ontology for computer science domain. We have taken stack topic of data structure from computer science to create the prototype of proposed system. Various stages are there for developing ontology. First stage is to gather the detailed information of the domain. Second stage is to identify all the classes and subclasses for the ontology to be developed. Third stage is to set the properties between classes and subclasses. Properties are of two types: Object properties and data properties. Object properties usually describe relationships between two instances or two individuals of classes. Data properties describe relationships between instances and data values. Every property has domain and range. Fourth stage is to set the domain and range of every property. Comments can also be added to classes and properties for the domain explanation. Fifth stage is to create instances of classes and set their data and object properties to define relationships between the instances of various classes and subclasses Sixth stage is for consistency checking.

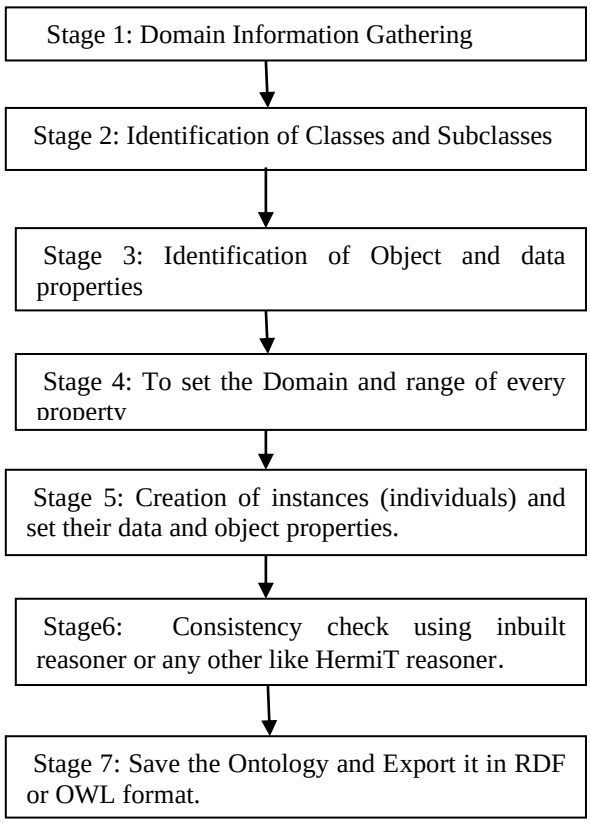


Figure 3: Steps for Ontology Building

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

Inbuilt Reasoner can be used to check the consistency of an ontology.Addition plug in like HermiT reasoner can also be used to check the consistency of the developed ontology. Sixth stage is to save the ontology in RDF/OWL format. Finally export the ontology in RDF or OWL data format to the required interface for execution of queries.

3.1.1 Creation of Classes and Subclasses

In computer science, the core concept of stack from data structures subject is considered to build the prototype for IRSCSD system. Stack ontology consists of various classes and subclasses as shown in figure below. The root node contain classes of various macro topics related to stack domain such as ADTs(abstract data types), Applications, Datastructure_category , datastructures, Examples, Implementation_through, Location_in_datastructure , Operations , Principle , Stack_terminology. These classes have various members which act as their instances. Like, ADTs class has three members dequeue, queue and stack.

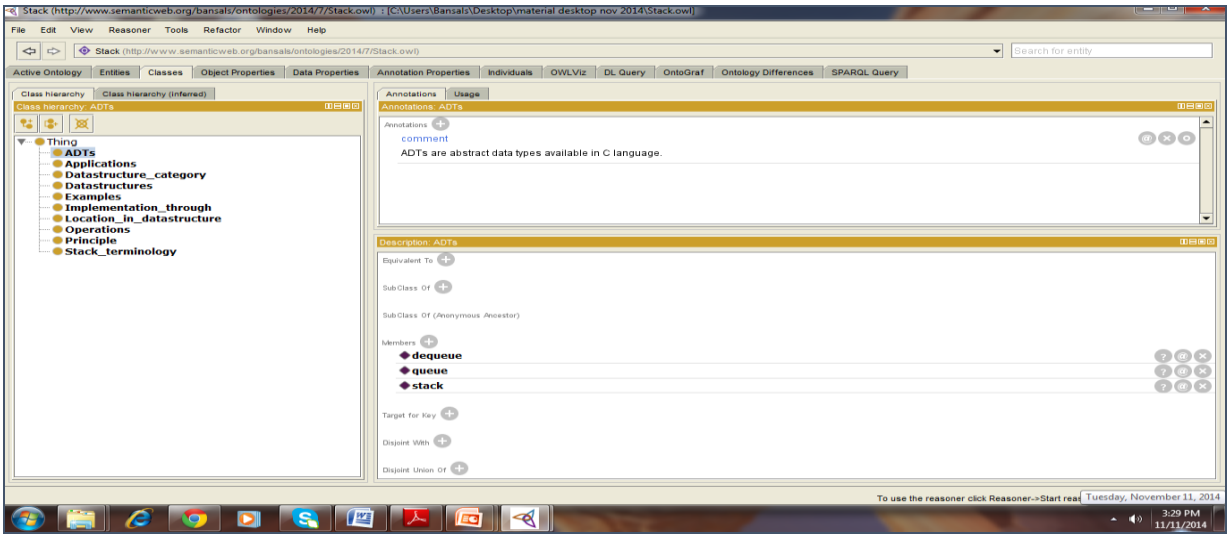


Figure 4: Classes and subclasses of Stack Ontology

3.1.2 Identify Object and Data Properties

Properties of different classes and subclasses are shown in figure below. Various object properties which describe the relationship between two instances or two individuals of classes have been set. Various object properties are core_operations, deletion_at , insertion_at, also_known_as , isa , stack_example , support_operations. Similarly , data properties are number_of_operations , principle_used. Every property has domain and range which needs to be set while developing an ontology. For example, Deletion_at object property has domain ADTs and range Location_in_datastructure.

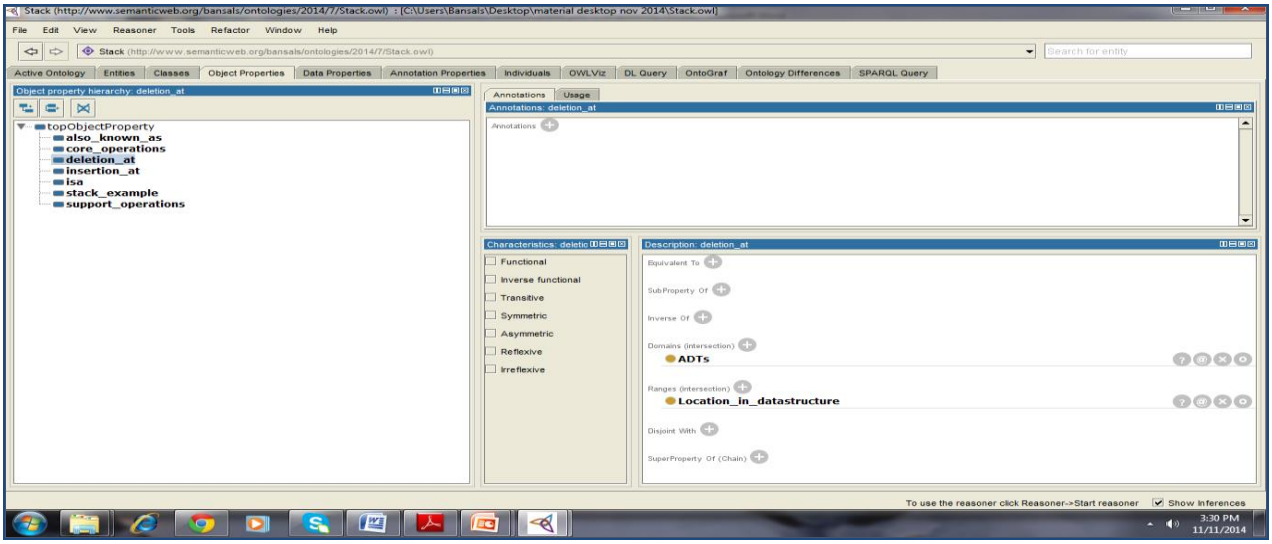


Figure 5: Object and data properties of Stack Ontology

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

3.1.3 Creation of Individuals

Individuals act as an instance for classes and subclasses. Through individuals (Instances), relationships are formed between all the entities of the respective Ontology. Figure shows stack as an instance of two classes ADTs and Datastructures. For the data property number_of_operations value given is 6 and principle_used is LIFO. Various object properties are also been set like core_operations are new, peek, pop, push .Likewise all the individuals are set for all the classes and subclasses.

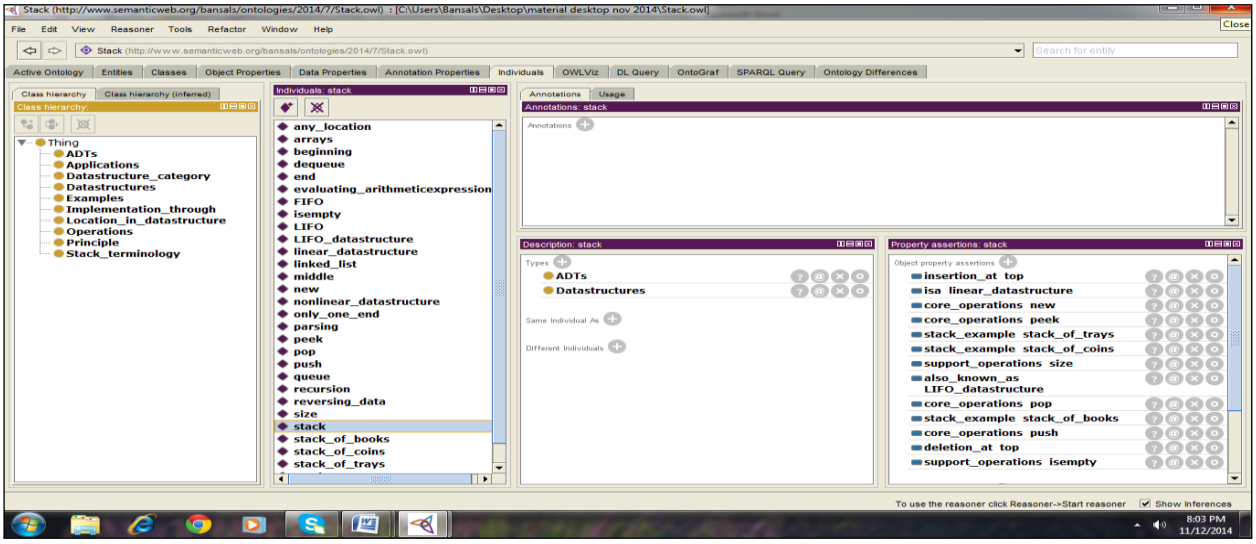


Figure 6: Instances (Individuals) of Stack Ontology

3.1.4 Consistency Check of developed Ontology

Protégé tool provide reasoner to check the consistency of the developed Ontology. You can start the reasoned and after consistency check, you can stop the reasoned. Various extra plugins for consistency check are available like HermiT reasoner. RDF validator is also their through which you can check the triples formed in the developed Ontology.

3.2 PHASE II: NLQ to SPARQL Conversion

Information retrieval system for computer science domain (IRSCSD) will provide an interface which will accept user queries in natural language. So, this system will be user friendly and there is no need of learning SPARQL language for retrieving data from RDF/OWL based database. The natural language query (NLQ) is converted into SPARQL query using QUEPY framework.

QUEPY is a python framework to transform natural language questions to queries in a database query language like SPARQL. It can be easily customized to different kinds of questions in natural language and database queries. The transformation from natural language to SPARQL is done by first using a special form of regular expressions and then using a convenient way to express semantic relations [13]. The rest of the transformation is handled automatically by the framework to finally produce this SPARQL.

Approach used for translation of natural language query into SPARQL query is comprised of five steps. First step is the recognition and disambiguation of Named entities. Second step is of parsing and disambiguation of the NL query and third step is to match query terms with Ontology concepts and properties. After this, generation of candidate triples is the fourth step. Finally, last step is the integration of triples and generation of SPARQL queries.

QUEPY is installed on Ubuntu 12.04 which is a Linux platform. Linux commands for QUEPY installation:

```
$yum  install python-setuptools
$easy_isntall pip
$pip install –U nltk
$pip install quepy
$pip install docopt
Pip install –U numpy
```

After execution of above linux commands, installation checking is done by following command:

```
 $ quepy version
quepy0.2
```

If successful, it will return the version number of QUEPY.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

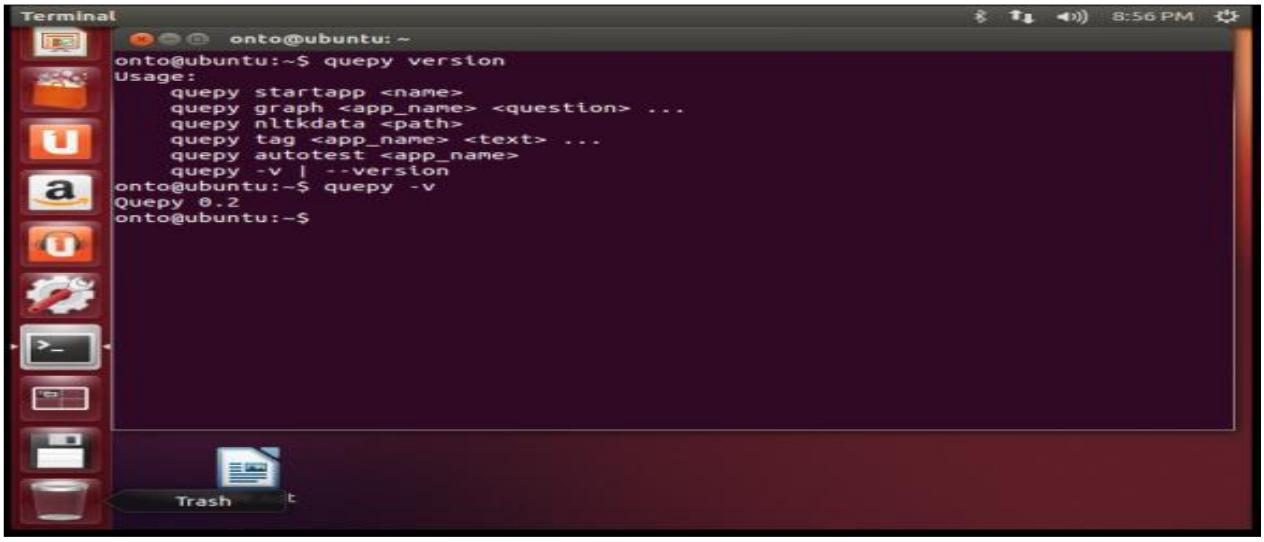


Figure 7: Installation of QUEPY

As shown in figure 8, in the input path of"who is Tom cruise" which is a natural language query is given and SPARQL query is generated.

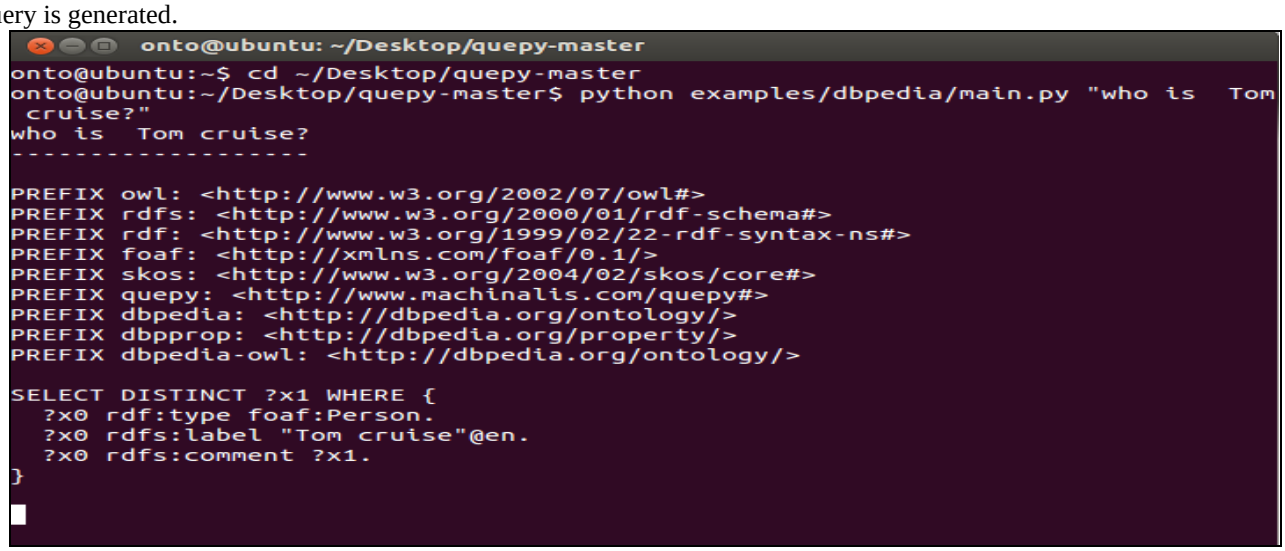


Figure 8: Natural language query converted to SPARQL query

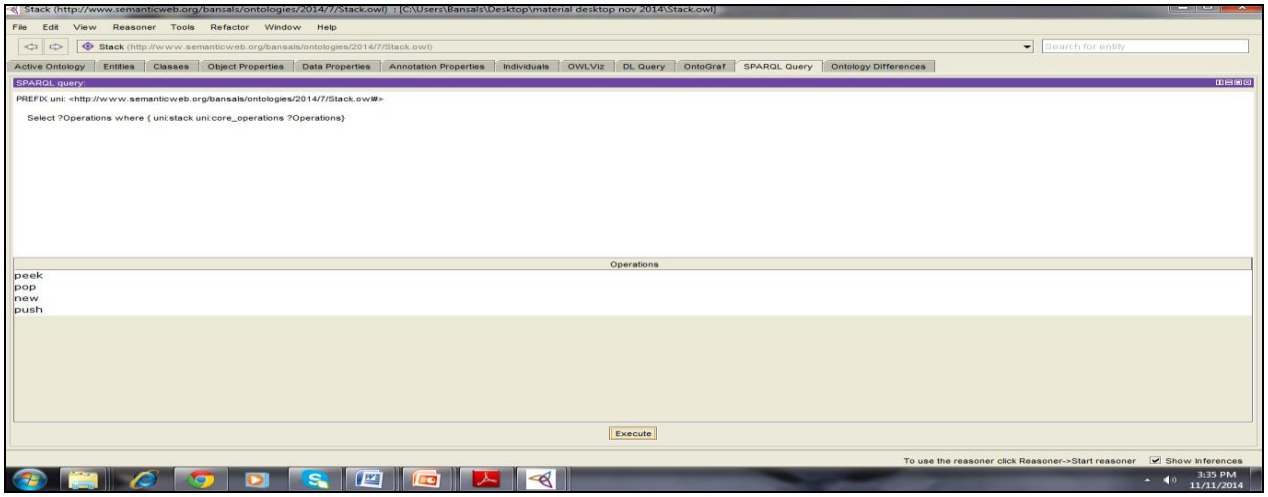
Similarly, for Ontology in computer science domain, if natural language query "What are the various operations on stack" is fired at the interface, it is converted to following SPARQL query:

PREFIX
uni:<http://www.semanticweb.org/bansals/ontologies/2014/7/Stack.owl#>
Select ?Operations where{
Uni:stack
Uni:core_operations
?Operations
}

3.3 Phase III: Running SPARQL query on Ontology and fetching results

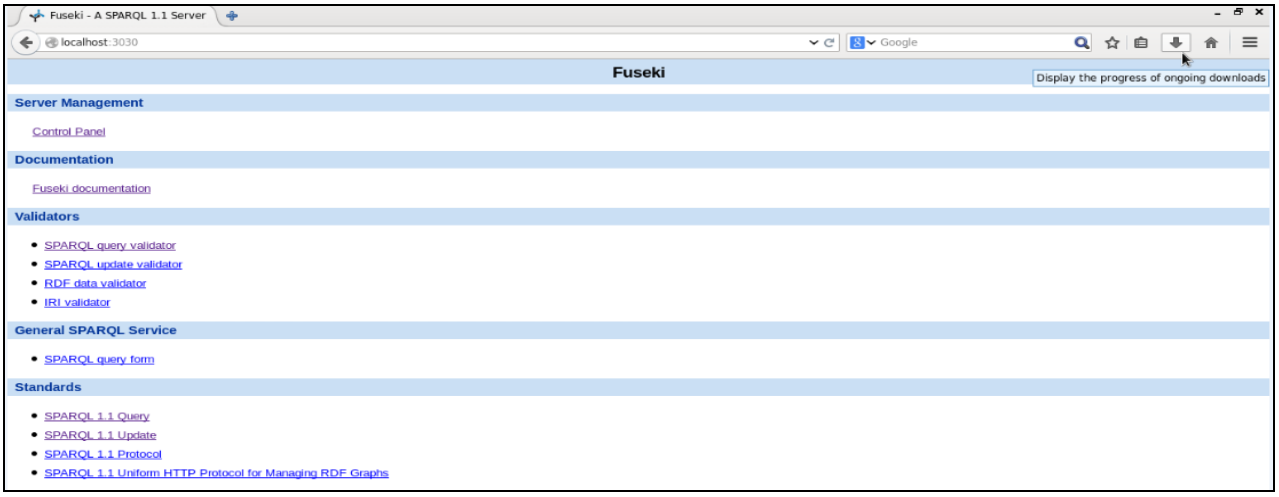
Third phase of Proposed Semantic Web based IRSCSD architecture is to fetch the desired results from Ontology. Three approaches were formed for running SPARQL queries on Ontology. The first approach is to run SPARQL queries through protégé. In, second approach SPARQL queries are executed through Apache's Jena Fuseki server which is GUI based. Third approach is to execute through Apache's ARQ which has Command line interface. Both ARQ and Jena Fuseki server are SPARQL engines and are open source. So, these are query engine for Jena that supports the SPARQL RDF Query language.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.



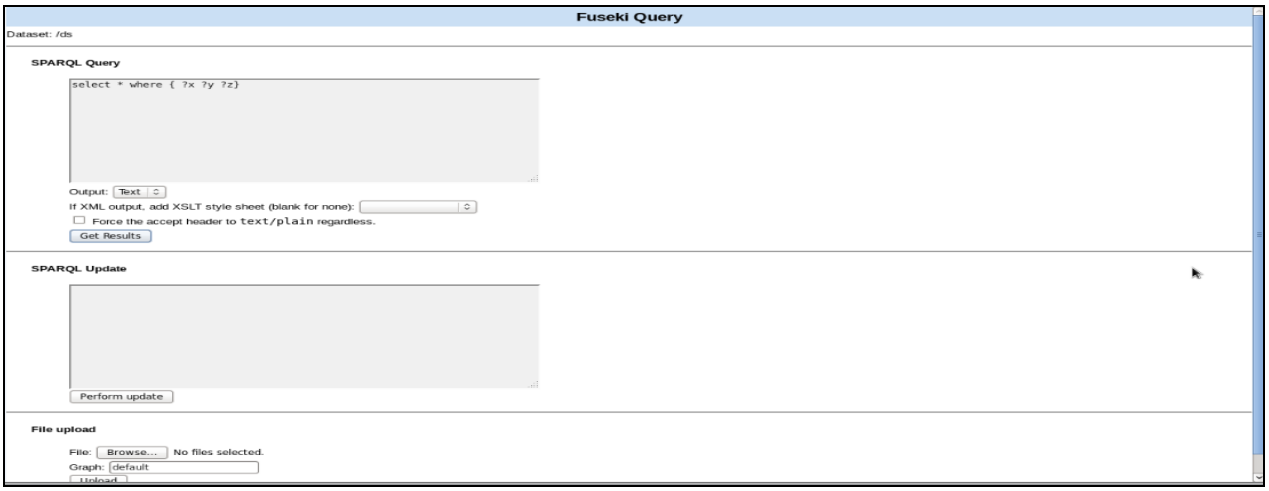
Approach 1: Running SPARQL query through protégé

Interface of Apache's Jena Fuseki server which is a Graphical user interface



Approach 2: Interface of Jena Fuseki Server

Through control panel, we can load the Stack Ontology and run the SPARQL queries and get the desired results. Approach 3 is not user friendly and is not shown due to some technical reasons.



Approach 2: Running SPARQL queries through Apache's Jena Fuseki Server

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

IV. Conclusion

Ontology is used to capture knowledge about any domain of interest with the objective of incorporating the machine understandable data on the current human-readable web. Ontology development is the most fundamental step and has been illustrated with a prototype developed in computer science domain. This proposed IRSCSD (Information retrieval system for computer science domain) architecture is for overcoming the limitation of keyword-based searching. The proposed system extracts relevant information instead of giving list of all the documents containing related information. The work can be extended to the development and deployment of large and complex Ontology and providing a solution for various other critical ontology issues towards semantic web.

References

[1] Malik, S. K., Prakash, N., & Rizvi, S. A. M. (2010). Developing an university ontology in education domain using protégé for semantic web. International Journal of Science and Technology, 2(9), 4673-4681.

[2] Guowei Chen,Pengzhou Zhang, "Keywords Retrieval Based On Ontology Inference", Communication University of China, International Conference on Industrial Control and Electronics Engineering 2012.

[3] Miriam Fernández , Iván Cantador , Vanesa López , David Vallet , Pablo Castells , Enrico Motta ,"Semantically enhanced Information Retrieval: An ontology-based approach "Web Semantics: Science, Services and Agents on the World Wide Web (2011).

[4] B.Chandrasekaran;John R.Josephson; What Are Ontologies,and Why Do We Need Them? IEEE Intelligent Systems, [J], 1999. PP20-25.

[5] IT .Kanimozhi, Dr.A.Christy," Incorporating Ontology and SPARQL for Semantic Image Annotation" Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT), 2013.

[6] Sonakneware.P.S.,Karale.S.J., " Ontology Based Approach for Domain Specific Semantic Information Retrieval System",IJERA,ICIAC,April2014.

[7] Bansal,R. , Chawla.R.,"Semantic Web Tool:For Efficient retrieval of Links and Required Information",IJITEE,Vol 3,Issue 4,September 2013

[8] Youn.S, McLeod.d,"Ontology Development Tools for Ontology - Based Knowledge Management,2006

[9] Wikipedia http://en.wikipedia.org/wiki/Semantic_Web

[10] Wang, C., Xiong, M., Zhou, Q., & Yu, Y. (2007). Panto: A portable natural language interface to ontologies. In The Semantic Web: Research and Applications (pp. 473-487). Springer Berlin Heidelberg.

[11] http://protege.stanford.edu/

[12] RDF Primer. W3C Recommendation. Feb, 2004. http://www.w3.org/TR/rdf-primer/

[13] Quepy. https://pypi.python.org/pypi/quepy/

[14] Gladun.A.,,Rogushina.J.,Shtonda.V.," Ontological Approach To Domain Knowledge Representation For Information Retrieval In Multiagent Systems", International Journal "Information Theories & applications" Vol.13.

[15] Dan.Z.,, "Research on Semantic Information Retrieval Based on Ontology", Library of Wuhan University of Technology, Wuhan, P.R. China, 430070.

[16] Lijun.T., Xu.C.,,"The Study of Semantic Retrieval Based on the Ontology of Teaching Management",Advancedin Control Engineering and Information Science CEIS 2011.

[17] Li, Y., Yang, H., Jagadish, H.V.: NaLIX: an interactive natural language interface for querying XML. In: SIGMOD Conference. (2005) 900{902

[18] Chen, H., Finin, T., Joshi A.: An ontology for context-aware pervasive computing environments. J. Knowl. Eng. Rev. 18(3), 197–207 (Sept 2003). Cambridge University Press, USA (2003). ISSN:0269-8889

[19] Bechhofer, S., Horrocks, I., Goble, C., Stevens, R.: OILEd: a reasonable ontology editor for the semantic web In: KI2001, Joint German/Austrian conference on Artificial Intelligence, volume LNAI Vol. 2174, pages 396-408, Vienna (2001)

[20] Xinhua.L., Xutang.Z.,, zhongkai.L.," A Domain Ontology- based Information Retrieval ApproachforTechniquePreparation international Conference on Solid State Devices and Materials Science 2012.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.