A Network of Heterogeneous and Distributed Ontologies for Health and Nutrition Information System

BY

ALI MAZHAR IQBAL

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER NETWORKS

MAY 2014

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN- 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **Ali Mazhar Iqbal** under the direction of his thesis advisor and approved by his thesis committee, has been presented and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER NETWORKS.**

Tarek Helmy

Dr. Tarek Helmy El-Basuny
(Advisor)

Dr. Basem Madani
Department Chairman

Dr. Mahmoud Elish
(Co-Advisor)

Dr. Salam A. Zummo
Dean of Graduate Studies

Dr. Mohamed El-Attar
(Member)

2/6/14

Date: May 2014

عمادة الدراسات العليا
DEANSHIP OF GRADUATE STUDIES

Dedication

I dedicate my work to my parents who are always praying for my success and to my wife who had to sacrifice during my studies and was fully supportive. I would also dedicate it to my sons who could not spend enough time with me during my work.

ACKNOWLEDGMENTS

My deep appreciation and heartfelt gratitude are to my advisor Dr. Tarek Ahmed Helmy EL-Basuny whose excellent guidance has been invaluable and provided me with an excellent atmosphere for doing this research. He always encourages me and the OSAPIR project's team members to be innovative. My heartfelt gratitude is also to my thesis committee members Dr. Mahmoud Elish and Dr. Mohamed El-Attar for their caring, patience and insightful comments. I would like acknowledge the support provided by King Abdulaziz City for Science and Technology (KACST) through the Science & Technology Unit at King Fahd University of Petroleum & Minerals (KFUPM) for funding this work through project No.10-INF1381-04 as part of the National Science, Technology and Innovation Plan. Thanks are extended to the project consultants Prof. Yuri Tijerino (Director of Web Science Lab, Kwansei Gakuin University, Japan), Prof. Jeffrey M. Bradshaw (Florida Institute for Human and Machine Cognition, USA). I would also like to acknowledge other members of the OSAPIR project where we evaluated and kept aligned our studies and approaches to achieve the common goal to support the project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

OWL	:	Web Ontology Language
RDF	:	Resource Description Framework
URL	:	Unified Resource Locator
HTML	:	Hyper-Text Markup Language
XML	:	Extensible Markup Language
RDFS	:	Resource Description Framework Schema
USDA	:	United States Department of Agriculture
WHO	:	World Health Organization

ABSTRACT

Full Name : Ali Mazhar Iqbal

Thesis Title : A Network of heterogeneous and distributed ontologies for health and nutrition information system

Major Field : Computer Networks

Date of Degree : May 2014

Web publishing is commonly done by the content writers independently to generate Web contents and linking them together. The main goal of the semantic Web is to extend the current human-readable Web by annotating the Web resources (i.e., attaching semantic metadata to a Web resource) to encode some semantics and to make them in a machine-readable form that can be accessed by the applications based on the predefined ontologies. The ontologies which actually help building this meaningful information are normally designed specific to domains and independent of each other e.g. Food, Health, and Nutrition. The landscape of ontology research is getting increasingly keen on the questions dealing with multiple heterogeneous ontologies which can help correlate the knowledge from different domains making it further useful as an integrated knowledge. Language barriers also limit the access of information to the users for various domains or services. This thesis investigates the work in the semantic integration between networks of heterogeneous ontologies and presents a framework for integrating of cross-domain multilingual ontologies. The thesis also studies the mechanisms to further enhance existing ontologies to support the integration process. The thesis presents a framework for the management of the enhanced ontologies and explains how it utilizes these ontologies to extract the knowledge from different Web resources and to make the extracted knowledge searchable by other systems inquire about food, health and nutrition assistance as a case study. Moreover, the thesis investigates the language barriers and proposed approaches to remove these barriers to make the information from various domains and languages integrated into one common knowledgebase to serve user's queries. All necessary APIs of the framework have been developed, tested and evaluated with the other components of the main framework to answer more specific queries about food, health and nutrition domains. Experimental results are encouraging and show that the management services provided by the proposed framework enable better semantic annotation of Web sources and queries to precisely answer inquiries about food, health and health issues.

ملخص الرسالة

الاسم الكامل: على مظهر إقبال

عنوان الرسالة: شبكات الأنتولجى الموزعة والمتباينة لدعم نظم المعلومات الغذائية والصحية

التخصص: شبكات الحاسب

تاريخ الدرجة العلمية: مايو 2014

عادة مايتم النشر على شبكة الإنترنت بشكل مستقل دون وجود ترابط بين محتويات الويب الموزعة. لهذا فإن الوب الدلالي يهدف الى تطوير الويب الحالى وزيادة الترابط الشبكي لمحتويات الوب الموزعة بناءا على الأنتولوجى مما يجعل محتوياته قابلة للفهم آليا بواسطة الحاسوب ومن ثم يمكن للتطبيقات المختلفة الإستفادة من هذه المعلومات المترابطة. من المعلوم أن الأنتولوجى الذى يساعد على الربط بين محتويات الوب وفهمها عادة مايتم تعريفه وتصميمه بشكل مستقل فى كل مجال. هناك العديد من البحوث التي تتناول كيفية الربط بين الأنتولوجى لعدة مجالات غير متجانسة والتي يمكن أن تساعد في ربط المعارف من مختلف المجالات مما يجعلها مفيدة وغنية بالمزيد من المعرفة المتكاملة. من المعلوم ايضا أن الحواجز اللغوية تحد من وصول المعلومات إلى المستخدمين في المجالات المختلفة. ولذا فان هذه الأطروحة تهدف الى دراسة التكامل الدلالي بين شبكات الأنتولوجى غير المتجانسة والموزعة وتقدم إطارا لدمجها والتغلب على حواجز اللغة. تدرس الأطروحة آليات لتطوير الأنتولوجى الموجود لتعزيز التكامل بينها فى المجالات المختلفة وبلغات متعددة. تقدم الأطروحة أيضا إطارا لإدارة البيانات الدلالية لإستخراج المعرفة من موارد الشبكة الموزعة مما يساعد في الاجابة على الإستفسارات المختلفة. علاوة على ذلك، فإن الأطروحة تساهم في ازالة حواجز اللغة لجعل المعلومات من مختلف المجالات واللغات في متناول الجميع ودمجها في قاعدة معرفية عامة واحدة لخدمة استعلامات المستخدم. تم تطوير جميع واجهات برمجة التطبيقات اللازمة للإطار وإختبارها وتقييمها مع مكونات أخرى من الإطار الرئيسي للرد على الإستفسارات الذكية وتحديدا فى المجالات الحرجة مثل الغذاء والصحة. النتائج التجريبية للإطار المقترح مشجعة و تظهر أن الخدمات التي تقدمها إدارة الإطار المقترح لمصادر الويب تساهم في الرد على الاستفسارات الذكية بشكل مقنع فى مجالات الغذاء والصحة.

CHAPTER 1

INTRODUCTION

A lot of study and research have been done since the emergence of the term Semantic Web by Timer Lee [1] to improve the content of Web by adding semantics such that these documents could be processed by software agents. Technologies like XML, RDF, and Ontology (OWL) allow presenting the information in the structured way which could be inferred by the software agents. Organizations use these technologies and create various ontologies for different knowledge domains. This thesis deals with the ontology management with respect to the integration of cross-domain knowledge. Ontology representing a domain holds the semantically understandable structure to store the knowledge which could be processed by humans or software agents. The research and application of ontologies still continue to be getting of more interest even after a decade. Various researches are done in ontology management area which include the development languages likes RDF [2] , OWL [3] and ontology engineering tools like Protégé as well [4]. With the abundant use of the ontologies, a lot of issues have been identified and different researches addressed these management issues of ontologies. These different issues such as creating ontologies, growing size of the ontologies, structure complexity, and aligning or merging multiple ontologies of a domain are addressed by different studies.

Building a framework for integration of cross-domains ontologies to integrate their knowledge from multilingual knowledge sources is the main objective of this thesis. There are different aspects of ontology management which should be handled by the management framework. This sub-framework is a part of **O**ntology-based **S**emantic **A**nnotation and **P**ersonalized **I**nformation Retrieval (OSAPIR) framework which can be used to develop an end to end portal for semantic integration of domain knowledge by extracting and annotating domain related knowledge to precisely answer the user's queries. The objective of this thesis is to build an ontology management framework to assist the whole process of the OSAPIR framework. Details about the OSAPIR framework are covered in Chapter 3. In the next subsections, I will discuss about motivation behind this research work, the problem statement for the research, thesis contribution and finally the organization of the thesis chapters.

1.1 Motivation

The Web we use today is full of continuously growing large collection of documents. These documents which are linked to each other cannot be interpreted by software agents. These documents are based on raw textual information which can be understood by human beings only. Providing the ontology management services that make the Web documents understandable by the software agents is one motivation behind our work.

With continuously growing Web resources with no semantics, the search engines are limited to keyword-based matching techniques and provide answers which are not all relevant to the user's specific need. This motivates us to use ontologies for adding semantic layer to these documents by providing ontology management services to annotate the documents and queries. Such annotated documents with the semantic layer could be reasoned to retrieve more relevant and precise answers.

After a decade of researches and studies, different organizations have begun creating ontologies for different knowledge domains like health, food and etc. The issue of interdependencies is becoming more complex with globalization such that it is no longer enough for the subject matter experts to develop this compartmentalized knowledge independently. Different ontologies have been developed of knowledge domains with no link to each other while interdependencies exist between them in real-world. This motivates us to develop a framework to manage these different heterogeneous ontologies to integrate the cross domain knowledge.

For integration of the knowledge sources of different domains, respective ontologies have to be integrated and managed such that these ontologies could be efficiently used for the purpose of knowledge integration. In any semantic Web application, management of the ontologies is required in order to annotate and reason the knowledge or due to evolution or improved versions of ontologies.

Language barriers often limit the access of information to the users for various domains or services. English is usually considered a main language for providing the information on the Web while it covers only 28.7% of all user of the Web [5]. Researches and studies being done all around the world and a lot of useful information is being published in different languages which should be equally accessible by all users. One of our motivations is to remove these language barriers while making the information from various domains and languages integrated to one common knowledge base to serve user's queries.

1.2 Problem Statement

To keep up pace with the growth of information on the Web, mechanisms are needed to allow efficient querying on the diverse information sources. In heterogeneous environment with diverse and segregated knowledge sources, ontology based manipulation to integrate knowledge of multilingual and cross-domain sources such as Food, Health and Nutrition, is probably the most desirable approach for semantic reconciliation. A framework that builds an ontology based semantic integration among diverse and cross-domain knowledge areas is essential. Such an ontology management framework which could resolve interdependencies and consider interrelationship among knowledge domains to make knowledge interoperable would be of great benefit to build knowledge-based search engines.

1.3 Thesis Contribution

This thesis contributes in developing a framework for the management of network of heterogeneous and cross-domain ontologies for semantic knowledge integration. Hereafter we summarize different points of contributions:

1. Providing an intensive literature review of existing ontology management tools and frameworks.
2. Presenting an ontology integration model to link cross-domain ontologies using the properties based on the real-world relations among them.

3. Developing management services of the integrated ontologies to provide services for OSAPIR framework to extract, to annotate and to eventually reason on the annotated knowledge for retrieving relevant and precise answers to the user's queries.
4. Developing the framework to be independent of any domains and flexible enough to be configured for integration among any set of domains.
5. Implementing of the framework in the area of health, food and nutrition to prove the concept of knowledge integration and provide precise answers to the users in such critical domains.
6. Providing multilingual support for knowledge annotation, query processing and reasoning to allow building complete multilingual system with unified knowledge from different languages.

1.4 Thesis Organization

The rest of this thesis is organized as follows. Chapter 2 provides the background of the semantic Web technologies and related work. Chapter 3 presents the OSAPIR framework requirements and components. Chapter 4 presents the ontology management framework which is the main contribution of this thesis. Chapter 5 presents the prototype and implementation details of the ontology management framework in the Food, Health and Nutrition domains. Chapter 6 presents the experimental results and analysis. Finally Chapter 7 concludes the thesis and highlights the future work direction.

CHAPTER 2

BACKGOUND & RELATED WORK

This chapter provides the background of our work and reviews related work. Section 1 & 2 defines the current Web and its search problems related to relevant information searching. Section 3 provides background about semantic Web & its technologies to solve the issue of no semantics in the current Web. Section 4 provides some details of well-known tools for ontology engineering and management. Section 5 discusses the approaches for integration of cross-domain ontologies. Section 6 discusses different management frameworks to evaluate integrated knowledge management from multiple domains. Section 7 discusses different approaches from different research to support multilingual ontologies.

2.1 World Wide Web

The World Wide Web was launched in 1991 by European Organization for Nuclear Research (CERN) to enable information sharing among computers [6]. Later it begun to grow and with explosion of personal computers and major advances in the telecommunication field were the triggers of the Web that we see today. The Web consisting of large amount of distributed resources of mostly the HTML documents linking many other media resources as well. Newer versions of HTML[7][8][9] came with many layout and design supports with different scripting languages support and Java applets, all elevated the interactive capabilities of Web pages.

2.2 Searching the Current Web

Information on the Web has grown too large which could not be easily browsed when looking for information. Information can be searched by using search engines which index the information spread across domains to provide easy search facility. These searches are based on the keywords matching on the documents which don't guarantee to bring the relevant results which user is looking for. Web documents which are based on HTML language have weak semantic support.

Let us consider the food and health scenario where the lazy lifestyle and modern meals play a role in causing a lot of diseases such as diabetes, hypertension, cardiac failure, and arthritis. The information on Web related to food and health is available by different sources and is segregated without any semantic interlink to each other. The difficulty of finding relevant and trustworthy information in this kind of heterogeneous environment creates an obstacle for citizens concerned about their health and nutrition. This situation highlights the need of intelligent search for relevant and precise health and food information that cannot be done by traditional Web search engines. Semantic processing techniques can help in better understanding the users' queries in addition to better structuring the scattered information on the Web. This results in more accurate and relevant search results from specific trusted sources meeting user's need.

2.3 Semantic Web Technologies & Tools

The semantic Web intends to add the meanings to the current Web resources. It attempts to make improved architecture for WWW which adds semantics to the content. If such semantic-based content resources are available on the Web, unlike current meaningless HTML contents, then automated software agents can be built for taking intelligent actions or tasks on behalf of users. A multilayer architecture for semantic Web has conceptualized by Timber Lee is shown in [10]. These different layers are syntax, data, ontology, logic and proof. The syntax layer deals with structure of elements where they are nested with other elements or attributed to other elements. XML [11] language is a markup language which is used in this layer as carrier for semantic information. The next is data layer where RDF [2] is used which allows encoding, exchanging or reusing of information. The third basic layer is ontology layer which is one basic component of semantic Web. Ontologies describe the formal structure knowledge, a hierarchy of concepts in a given domain. The next layer is the logic layer which consists of rules that enable the reasoning on the knowledge allowing intelligent answering by the automated agents. The last is the proof layer which provides the explanation and provenance of the answer which means that the fact is extracted from particular source or origin.

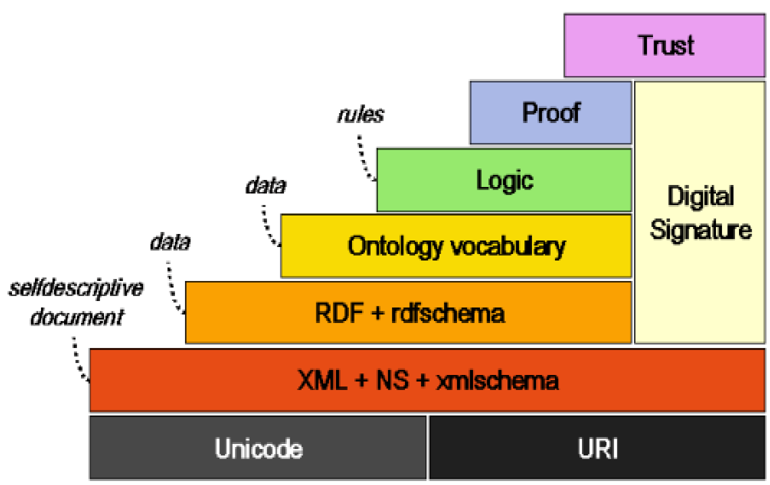


Figure 1 - Semantic Web Layers

2.3.1 eXtensible Markup Language (XML)

eXtensible Markup Language (XML) [11] has provided the semantic by embedding some metadata in the form of human-readable tags describing data. XML documents can also include additional information such as author, relevant keywords for search engine optimization, and the software tools used to create the XML file.

Before XML, data was stored in flat file and database formats, where most of data was proprietary to an application. XML came along and made data interoperable within a single domain, i.e., within the domain defined by a schema or a set of related schemas. By itself, XML provides syntactic interoperability only when both parties know and understand the element names used. If I label an element <price>12.00</price> and someone else labels it <cost>12.00<cost>, there’s no way for a machine to know that those are the same thing without the aid of a separate, highly customized application to map between the elements. Semantic Web technologies address this problem by making tags understandable not just to humans – but to machines as well.

The first step required for machines to understand data is to get that data into a uniform format, where, for instance, a field labeled “street” always has the same format and contains the same type of information, and so on. This type of functionality can be found today on Web sites that use forms that allow users to enter information and run a query, such as airline Web sites that allow visitors to search for and book flights based on a variety of criteria. However, considering the amount and variety of data available from different sources today, this method of data typing does not scale beyond very specific applications. The next step towards the semantic Web requires that data from multiple domains is classified based on its properties and its relationship with other data. This is where semantic Web technologies such as RDF, RDFS, and OWL come in.

2.3.2 Resource Description Framework (RDF)

Resource Description Framework (RDF) [2] is recommended by W3C for defining the resource and considered as first level of knowledge representation formalism. RDF is built using XML and URI technologies to make statement about the resource. RDF statements describe the properties and values of a resource and are often referred as triples. The triple consists of subject; predicate and object which correspond to a resource (subject) a property (predicate), and a property value (object). Figure 2 is an example of an RDF statement in plain English:

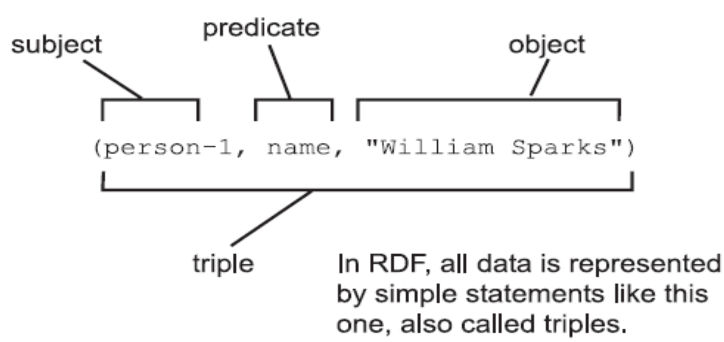


Figure 2 - RDF Triple

2.3.3 Resource Schema

Resource Description Framework Schema (RDFS) [12] is a data modeling vocabulary for RDF [2] that describe the RDF resources and relationship between the resources. An RDFS vocabulary defines the allowable properties that can be assigned to RDF resources within a given domain. RDFS also allows you to create classes of resources that share common properties. Using the same triples paradigm defined by RDF, RDFS triples consist of classes, class properties, and values that define the classes and relationships between the resources within a particular domain.

In an RDFS vocabulary, resources are defined as instances of classes. A class is a resource too, and any class can be a subclass of another. This hierarchical semantic information is what allows machines to determine the meanings of resources based on their properties and classes.

2.3.4 Web Ontology Language (OWL)

Ontology is the key technology for semantic Web. Web Ontology Language (OWL) [3] is a recommended language of semantic Web by W3C which represents complex knowledge in a structural form. It's a computational logic based language allowing software agents to interpret the knowledge. OWL based documents are also known as ontologies. Ontology defines the structure of knowledge and provides common vocabularies to share the information for a given domain. It consists of concepts and relation among different concepts which are machine interpretable. Both humans and machines share the knowledge using ontologies. Ontologies provide common understanding of the knowledge of particular domain which allows reusing and sharing it across different organizations or applications. It defines terms and relationships among the terms and various properties of these terms to formulize the domain.

To compare the knowledge between two knowledge bases they should have common understanding of the terms and structure of the knowledge. A software agent should be able to discover the common meanings but unfortunately the Web knowledge bases are not machine understandable. Ontologies provide solutions to these problems by encapsulating the knowledge into the ontology itself.

OWL documents are independent and modular. Multiple ontologies can be referred dynamically to read or understand. Software agents can access these documents to interpret and find the relationships among the concepts being inquired by looking at multiple facts to drive the required fact.

2.4 Ontology Engineering & Management Tools

There many ontology engineering and managements tools available from both open source and commercial communities. The most commonly used tool in the research for ontology engineering is Protégé [4]. It's an open source tool for editing and managing the knowledge of the ontology. Protégé supports two way of modeling ontologies, Protégé-Frames and Protégé-OWL editors. Protégé is integrated software tools which used by many knowledge experts for building the ontologies.

OntoStudio [13] is one widespread and commercial modeling tool supporting creation and maintenance on the ontologies. It stands out due to its comprehensive function of ontology modeling. It has mapping tools which can be used to match heterogeneous ontologies intuitively.

Jena [14] is a Jena is a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFS and OWL, SPARQL and includes a rule-based inference engine. It's an open source project and provides RDF and OWL APIs along with in-memory persistence storage and SPARQL query engine.

PowerAqua [15] is a multi-ontology based question answering system which supports query in natural language and draw results from distributed knowledge sources. PowerAqua system that is able to answer queries by locating and integrating information, which can be distributed across heterogeneous semantic resources.

TopBraid Composer [16] is another enterprise class ontology engineering tool for developing semantic Web ontologies. It’s a leading industrial standard RDF and OWL ontology editor as well as one the best SPARQL tool. It also includes flexible published APIs for building semantic client server based applications. Different versions are available from free to commercial with varying features.

BigData [17] is a horizontally-scaled, general purpose and computing for ordered data. It is designed to support single server environment and the cluster environment for scalability. It has no scalability limits and can be even deployed in the thousands of the servers. It supports RDFS and OWL reasoning.

Sesame [18] is standard de-fact framework for processing RDF data since most of the researches and studies using the framework for research or semantic Web applications. It includes parsers, persistence storage, reasoning and querying, by using the SPARQL language. Sesame is used worldwide by the large companies, government agencies and research industries. It has a very flexible architecture and adaptable architecture which is one the main reasons of its popularity.

OWLIM [19] is a family of semantic repositories or RDF database management system which provides robust support for RDFS and OWL with native RDF engines built using Java [20] language. OWLIM is used in large number of researches and semantic products. It supports reasoning on large datasets and enables us to query billions of facts [21]. OWLIM comes packaged as storage and inference layer (SAIL) for Sesame and openRDF Framework [18].

We discussed some of the surveyed software and tools for ontology engineering which can help us in building the proposed framework. In this thesis, we present ontology for the integration where we choose TopBraid Composer due its features and ease of use. We went through many cycles of ontology reviewing and engineering for the case study implementation. We built the framework for managing the integrated ontologies for different purpose and in order to store and reason on the ontologies we used Sesame for the persistence of the ontologies and knowledge in the repository. Advantages of Sesame are that it has very flexible architecture and supports reasoners like OWLIM and Jena which are beneficial for the proposed framework.

2.5 Integration of Heterogeneous Cross-Domain Ontologies

With continuous growth of studies on the semantic Web, the interest of ontologies has increased. Ontologies are being created by different organizations based on the different point of view from the subject matters experts. They use different methodologies and tools for creating these ontologies. Even being the same domains, these ontologies are heterogeneous and require some sort of mapping or integration among them for interoperability [22]. The spreading of ontologies over the various research communities has all together produced rising variety of tools and techniques to construct, maintain, manage, merge, map, as well as match these ontologies. Different techniques to resolve the heterogeneity issue among domain ontologies are as below:

- **Ontology mapping**, mapping concepts from different ontologies based on similar relation.
- **Ontology alignment**, a set of agreement between two or more ontologies which is normally an output of the ontology matching.

In the above two approaches the ontologies are updated to work together while the approaches below produce new ontology based on the existing ones.

- **Ontology merging**, where ontology is produced from two sources with overlapping sections.
- **Ontology Integrating**, where a new ontology is developed by reusing other available ontologies.

Above provided approaches for interoperability among the domain ontologies can be applied to ontologies belonging to one domain. If the ontologies belong to different domains then there is no mapping between them and merging these ontologies by identifying the concepts and relationship of different domains ontologies is like merging two knowledge domains into one which is not the right solution.

Siddharth Taduri in his research [23] for integrating different information domains to patents system, created a new ontology based on the two ontologies for Patents and Courts. In their approach, they proposed the ontology for Patent system and defined the semantics expressed in both domains to provide unified knowledge base. This approach of integration limits us to reuse the ontologies and knowledge actually created by the experts in their domain. In addition, for creating such ontology for any two domains to be integrated one needs the expertise from both domains to come up with such ontology.

Tejal and Hethi in their research [24] proposes a cross-domain ontology (OSHCO) semantic interoperability across Medical and Oral domains. They designed an ontology which covers the relationships among the medical and oral health domains. This approach is very similar to the Patent system case which we discussed earlier. In both scenarios, domain experts from different domains have to work together to re-engineer the ontology to cover the concepts and relation from both of these ontologies.

G. Vadivu and S. Hopper in [25] linked ontologies for food, chemical and diseases by bringing them to common agreement between the instances of knowledge, which allows user queries to be semantically answered.

We discussed different approaches for interoperability of ontologies from different domains. Merging the domain ontologies into single domain ontology violates the idea of the domain ontology. We want to reuse existing ontologies by selecting the ontologies based on the criteria of trusted knowledge and extend the ontology for language improvements if needed. The approach we took is to link the ontologies based on the relations among domains using an upper layer ontology which links them using properties defining the relationship. Such ontologies are required to be management for the process of semantic analysis, annotation and reasoning of the content. So we evaluated some methodologies which could provide similar nature of ontology management to help us in building the proposed framework.

2.6 Ontology Management Issues

There are some proposed frameworks of the ontology management that deal with different issues of management for processing of data using ontologies.

Zhan Cui and Paul O'Brien in their study [26] highlighted the importance of ontology management and proposed an ontology management framework (DOME) which is suit of tools for single domain ontology management.

Alexander Maedch and Raphael Volz in [27] emphasized the significance of the ontology building and management systems and proposed a framework for producing ontologies semi-automatically with the text of any specific domain by applying the machine learning approaches. The authors introduced the architecture of the framework and explained the way it is utilized for extraction of ontologies. The framework supports several ontology engineering tasks which fall under two categories of algorithms: ontology extraction and ontology maintenance. The framework is also deals with producing ontologies for the same domains.

A. Aldea in the his study [28] discussed multi-agents based platform which uses ontologies and apply learning techniques to extract the information and discover the new concepts in the Web. The framework utilizes different sorts of agents for different tasks, use domain ontology for retrieved knowledge and updates the domain ontology. All the information is merged into single ontology.

There is no comprehensive framework available for building semantic Web application by the use of ontologies from different domains in order to make interoperable. Although there are approaches used in the ontology integration to merge the ontology into single domain, we opted to keep the ontologies separate. Ontologies should be re-used which are provided by some well-known publisher and extended if required to improve such as translations. This approach will not force us to merge the domain ontologies and will also allow us to update the newly available ontology from the publisher to be used easily. But we have identified the requirements of such framework for managing the upper layer and domain ontologies for integration of knowledge.

2.7 Multilingual Ontologies

Information available on the Web is language independent and generally user prefers the availability of the information related to the language of his choice, so the availability of the language independent knowledge is the need of today. Utilizing the various languages in the study of Ontology can also be a challenge to many attempts of the Web designs to cater the thousands of users in the WWW.

In research communities, multilingual ontologies have become vital need to support global knowledge understanding. The most wide spread technique is the use of labels and description to embed the translation and provide language description. Elena and Guadalupe [29] in their study proposed a technique to link the ontologies to linguistic model stored externally. They called it Linguistic Information Repository (LIR). With this approach, it provides multilingual information of all elements in addition to unified access to ontology for heterogeneous multilingual information.

Deryle and David in [5] proposed an approach to have multilingual extraction ontologies to resolve the issue of language barrier in the information available on the Web. With this approach there is a separate ontology for each language identical to each other allowing extraction system to use ontology based on the language of the information being extracted.

Based on our study we learned few approaches that are currently being followed by ontology providers of different domains. As we intend the reuse the existing ontologies from trusted publishers, we must adapt these common approaches for ontologies in our framework. The proposed framework supports ontologies with embedded translation as labels also it supports independent ontology for each language. In addition to that we implemented another approach to provide external linguistic information which is similar to LIR approach which is discussed earlier.

2.8 Survey of Food, Health & Nutrition Ontologies

The OSAPIR framework is used as a case study in the domains of Food, Health and Nutrition where the ontologies of these domains were reviewed and evaluated to be used for knowledge integration. Different aspects of semantic integration were considered such as:

- If the ontology fit for the acquisition and annotation or not.
- Does it have enough sufficient vocabulary to process the annotation?
- Does Arabic ontology exist or how the Arabic support is designed with the ontologies?

Above are few questions which guided us while reviewing these ontologies or extending ontologies later for use by the OSAPIR framework.

Semantic Diet [30] by Evan, intends to help people with healthier diet. It provides Food and Nutrition ontologies which are based on the USDA [31] database for food and nutrition data and relationship. USDA database is trusted source containing a comprehensive list of all types of food and its nutrition values. It also includes some useful ontologies for measurement of food and serving sizes. Advantage of using Semantic diet is that it's built using USDA database and is used in many semantic applications. One problem with the ontology is that Arabic is missing and it exists in English only.

AGROVOC [32] provides rich ontologies for Food and Nutrition with multilingual support. The problem we see is that food and nutrition information in the ontology is not aligned with USDA database.

FOODS [33] ontologies provides different ontologies which include food, nutrition and disease as well. It doesn't have Arabic versions and it's also aligned with USDA database.

With the above described advantages and disadvantages for each ontology source, we selected ontologies from Semantic Diet as these are aligned for USDA database and we extended these ontologies to add Arabic translation.

ICD10 [34] ontology is an OWL-DL is the International classification of diseases which was published by World Health Organizations (WHO) [35]. It is used for health management and clinical purpose to maintain the history of occurrence and frequency of diseases. A positive aspect of the ontology is that it's available in different languages including the Arabic. The ontology is designed to categorize diseases and health issues which could be based on the various types of health and important records. The ontology is hierarchical in nature and classifying all these concepts into many levels, such that the concepts are not self-explanatory unless a complete parent hierarchy is observed to understand the actual vocabulary a set of concepts. Moreover, the ontology uses the technical names of disease and do not embed synonyms in the ontology. Such ontology makes the text processing less effective as more work is required to map the ontology concepts to the text being annotated.

Disease Ontology (DO) [36] is open source ontology for the integration of biomedical data that is associated with human disease. Terms in DO are well defined, using standard references. These terms are linked to well-established, well-adopted terminologies that contain disease and disease-related concepts. Each concept has a reference for most common health related ontologies with different synonyms or alternative names for the same concept. It is very useful for semantic annotation for two reasons; self-contained names used for each concept and rich set of synonyms for each concept. For those reasons, we have selected this ontology for our case study for semantic annotation of disease concepts. The only limitation of DO is related to multilingual support since it is only provided with English names only.

We evaluated different ontologies for case study implementation where we targeted the domains of food, health and nutrition and considered the support of Arabic language to cover the aspect of multilingual knowledge. We used Food, Health and Nutrition Ontologies from Semantic Diet [30] as they are aligned with USDA database which could be considered as trustful source. For the disease ontologies we opted DO [36] and engineered to address the limitations of language by extending with translation.

CHAPTER 3

A FRAMEWORK FOR ONTOLOGY-BASED SEMANTIC ANNOTATION FOR PERSONALIZED INFORMATION RETRIEVAL (OSAPIR)

In this chapter we briefly discuss the OSAPIR framework which covers various aspects of the framework utilized to build a complete portal for the knowledge integration of any domain. We provided little background of the problem and highlighted few goals which motivate us to propose the OSAPIR framework, followed by the architecture and three main components of the framework. The third component in the last section of this chapter is the main focus of this thesis.

3.1 Framework Objectives

The Web content is growing exponentially which brings a lot of challenges to access the information. With this growth of the Web content, the users' demands to find the relevant information have increased. Most people use the traditional Web search engines to locate any information, such as Bing, Google and Yahoo. Not all users are satisfied with the current search engines as they do not find the search results relevant to their needs. There is a need to have a fast and automatic ontology-based semantic manipulation of Web sources content. This is important in critical domains, such as health, food and nutrition where users need to retrieve precise and relevant health, food and nutrition information that fit their needs from trusted sources. To achieve this semantic reconciliation of knowledge from different domains, there is a need of a platform which could help us achieve it through different processes like modeling ontologies, extraction and annotation, inferring knowledge and personalizing the responses. Next are few main objectives behind the OSAPIR framework.

3.1.1 Language Independent Knowledge

Although a huge percentage of the Web content is presented in English, still there is a lot of content in other languages [5]. In traditional Web, access to cross-lingual content is only possible if websites are translated into the corresponding languages. There is a lacks of explicit mechanisms to automatically reconcile information expressed in different languages. This leads to situations in which data expressed in a certain language is not easily accessible to speakers of other languages. Semantic Web offers a great opportunity to make Web information broadly accessible, independent of culture and native language. One of the main objectives behind OSAPIR is to remove the barrier of language for use of information while providing semantically processed answers to user

3.1.2 Cross-Domain Integrated Knowledge

Different knowledge experts are working independently in the area of their expertise with no link to each other. Such non-integrated knowledge when searched using current Web search engines, can answer users' questions with no relation and semantic understanding between these domains. Semantic Web can play a very important role by providing the understanding and the semantics of a given domain. But we are challenged and motivated by a requirement where knowledge from these heterogeneous sources with different domains could be semantically integrated. This cross domain integrated knowledge should enable us to answer user's question referring to multiple domains by semantically understanding the query and reasoning the answer based on the relation among the domains. Cross-Domain Integration of the Knowledge is the main objective of the OSAPIR framework.

3.1.3 Semantically Relevant Results

The search engines crawl the Web content and create indices that are used to retrieve the results for users' search queries. The users write their queries using natural language while the current search engines are keyword-based. This leads to a challenge to understand the user's queries correctly. Moreover, the users might not be able to express all their needs explicitly while the search engines are limited to the provided query to bring the matched results. So, because user's needs are different, the relevancy of the retrieved results varies from a user to another user. This leads to a challenge to get the relevant and personalized information based on the user's needs. Semantic Web addresses the relevancy by semantic understanding of the users' queries and the reasoning with the annotated Web sources based on the integrated domain ontologies. Moreover, the personalization technologies help in understanding the users' needs better which can support in semantically enriching the queries and retrieving personalized results. This raises the challenges of semantically manipulating the users' queries, reasoning and annotating the Web content based on the domain ontologies. OSAPIR framework helps us achieve the semantic understanding of Web contents as well as user's queries by utilizing semantic Web technologies.

3.1.4 Need of a Framework

Some domains are more critical such as the health and food domains which make these challenges more obvious. So, there is a need to have an integrated infrastructure that handles the above challenges. An infrastructure in a form of framework with supporting semantic Web and personalization technologies will help the Web developers to develop semantic applications for different domains.

A framework is a software platform for developing the application. It provides basic foundation for software developers to create application for a given platform. Generally frameworks provide application programmable interface (API) for accessing its components where the framework itself serves as pillars for building up the application where developers don't have to do everything from scratch. A framework may also include additional software libraries and other programs used in the software development process. So these can be considered basic requirements for any common framework for development.

We propose a framework for **O**ntology-based **S**emantic **A**nnotation for **P**ersonalized **I**nformation **R**etrieval (**OSAPIR**). Below, we present the proposed framework that is capable to handle multi-lingual cross domain Web content and can be easily adapted to any domain such as the health and food domains. We start with discussing the requirements of such framework then we show the proposed framework architecture. Then, we briefly describe each component of the framework.

3.2 Requirements

We aim to build a multi-lingual cross domain personalized semantic Web search framework that can adapt to any domain such as the health and food domains. Below we present the requirements for such semantic Web search framework.

a) The framework should be applicable to any domain with minimal customization.

b) The framework should support multilingual with respect to ontologies, Web sources, knowledge-bases, and user's queries.

c) The framework should facilitate cross domain integration of ontologies and knowledge-bases.

d) The framework should support acquiring and annotating Web sources in heterogonous formats.

e) The framework should provide a mechanism to decide the trust level of the acquired Web sources.

f) The framework should generate standard semantic annotation formats for the acquired Web sources based on the domain ontologies.

g) The framework should semantically manipulate the user's queries.

h) The framework should provide reasoning capabilities for answering user's queries.

i) The framework should capture and model the user's preferences.

j) The framework should personalize the retrieved results.

k) The framework should support standard ontology representation format.

The framework should provide the required ontology management services to achieve the desired objectives, i.e. alignment of ontologies from different domains and languages.

3.3 Proposed Framework

Based on an intensive literature review and discussions among the project team members including the consultants, we propose a framework that addresses the above requirements, framework for Ontology-based Semantic Annotation for Personalized Information Retrieval (OSAPIR). The proposed framework is capable to adapt to any domain by defining the domain ontologies, lexical resources, trust level and seed Web sources. Furthermore, the framework supports multilingual on ontologies, Web sources and user's queries. Figure 3 shows the architecture of the proposed OSAPIR framework.

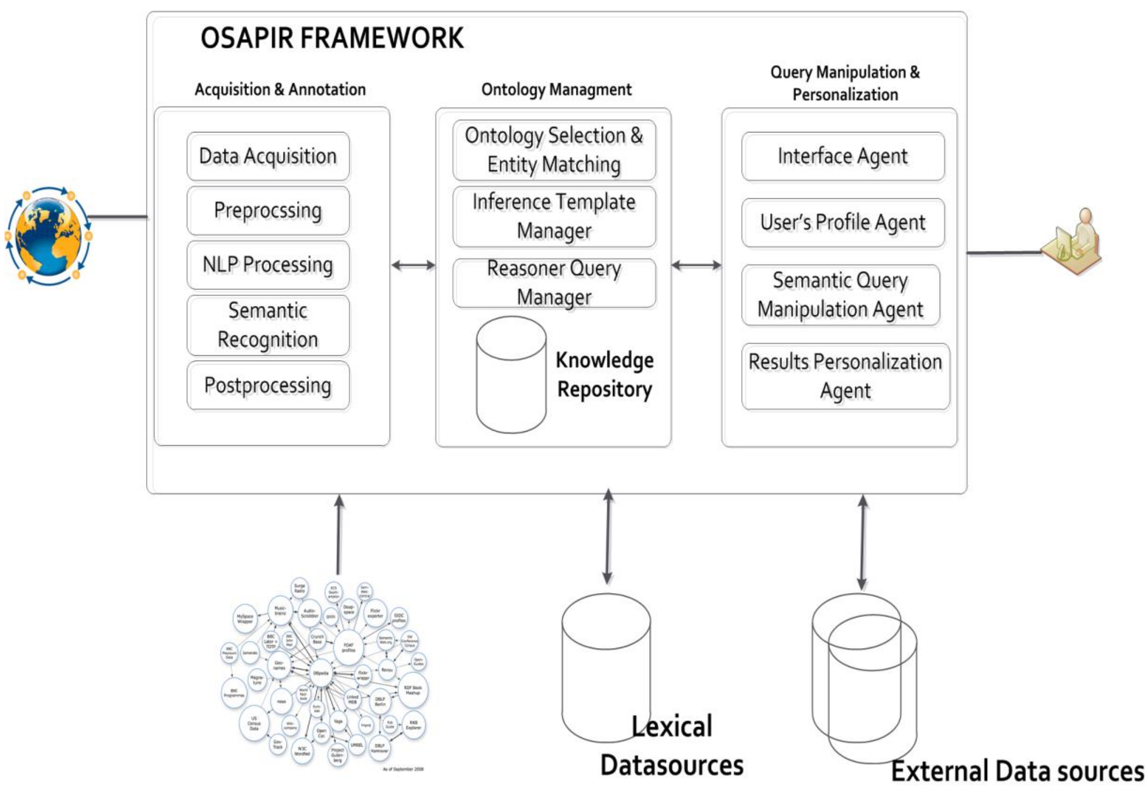


Figure 3 - OSAPIR Framework

Figure 3 shows the Architecture of the framework for Ontology-based Semantic Annotation for Personalized Information Retrieval (OSAPIR).

There are three dimensions of the requirements that work together to achieve the framework's objectives. First, users' queries need to be semantically understood according to the domain ontologies. The retrieved results from the knowledgebase should be personalized based on their needs. Second, the Web content needs to be annotated according to the domain ontologies in order to populate the knowledgebase. Third, the cross domain ontologies and knowledgebase need to be managed in efficient and effective way. As a result, the proposed framework is divided into three major components: Data Acquisition & Semantic Annotation Component, Ontology Management Component and Semantic Query Manipulation & Personalization Component. Below is a brief description for each component.

3.3.1 Data Acquisition and Semantic Annotation Component

The main goal for this component is to collect and annotate the contents of multi-lingual Web sources based on the pre-defined domain ontologies. This component consists of two major layers; the acquisition layer and the semantic annotation layer.

The acquisition layer consists of multiple data integration tasks for the purpose of collecting data from Web sources related the targeted domains. The collected data from Web sources are then used by the annotation layer for semantic enrichment. The acquisition layer can be configured to collect data from specific websites based on certain criteria such as trust level or pre-defined seeds websites. The relevant Web sources are collected based on the relevancy to the domain ontologies. This layer supports processing of all common Web document formats such as HTML, XML, PDF, Office document and multimedia format.

The semantic annotation layer annotates the acquired Web sources based on the domain ontologies and the predefined cross-domain integration. Moreover, it provides multiple mechanisms to perform automated annotations for semi-structured (i.e. tables) and un-structured (i.e. paragraphs) Web sources. This layer can produce embedded annotation inside the Web document using standard annotation languages such as RDF. This component of framework is taken care by another research “Multilingual Framework for Ontology-Based Semantic Annotation of Health & Nutrition Websites“ [37] .

3.3.2 Semantic Query Manipulation and Personalization Component

This component is used to interface with the end-user, captures and models the user’s preferences into a user’s profile. It semantically manipulates the multi-lingual user’s queries and enriches them with more information from the user’s profile. This component interacts with the Ontology Management Component for query reasoning based on the domain ontologies and knowledge-bases. Moreover, it personalizes the retrieved results and captures the user’s interactions to enhance the user’s profile and provide more relevant answers. This component of OSAPIR framework is taken care by another research “Agent-based Framework for Semantic Query-Manipulation and Personalized Retrieval of Health and Nutrition Information” [38]

3.3.3 Integrated Ontology Management

The Ontology Management component which is the main contribution of this thesis takes care of managing a network of heterogonous ontologies and knowledge-bases required by the main framework, i.e. integration model for cross domain and/or multi-lingual ontologies. It also provides different ontologies management tasks for processing of information, i.e. mapping of various ontologies for more efficient sharing and reuse. This component can process any standard ontology representation languages. It also provides API interfaces to access the ontologies by other two components of the proposed framework. In addition, it provides reasoning capabilities on the knowledge-bases to allow semantic answering to the user's queries. In the next chapters, we will elaborate more on the architecture of framework with details of management's tasks handled by the framework.

CHAPTER 4

A FRAMEWORK FOR MANAGING HETEROGENEOUS AND DISTRIBUTED ONTOLOGIES NETWORK

4.1 Introduction

Ontology management is a core part of the OSAPIR framework which is discussed in the previous chapter. This chapter focuses on the ontology management component of the OSAPIR framework. Ontology management framework takes care of different management tasks related to ontologies network which allow the cross domain integration of knowledge. In our case, where we are required to integrate the knowledge of different domains, we need ontologies to represent each domain. Ontology represents the structure of knowledge for a given domain. As we intend to integrate the knowledge from different domains, different heterogeneous ontologies representing each domain are required to be used together in an integrated manner. Such integration of ontologies of different domains for knowledge integration makes it necessary to have ontology management framework to handle these tasks. Since our scope is not only to retrieve the information from these different sources independently but in fact, is to retrieve the knowledge which relates to each other and then can be inferred based on the respective ontologies. The ontology management framework facilitates the management tasks required for the semantic annotation process of user's queries and Web sources. Such information when annotated is again persisted and managed by the framework which allows intelligent answering to questions based on the semantic reasoning. The following section highlights the goals of designing the ontology management framework and later sections discuss with more details about various components of the framework.

4.2 Goals of the Ontology Management Framework

Ontologies are becoming ubiquitous in the information systems [39] and these can be considered as the backbone technology for the semantic Web. Ontologies are being developed in massive scale using various editors as well as in multiple languages. These huge number of ontologies consist of many unrelated domains and sometimes overlapping but with different granularity or levels. These huge numbers of ontologies raise problems of managing such kind of ontologies. Ontology evolution, ontology versioning and persistence are few basic management tasks that one needs. In case of OSAPIR, the ontologies from different domains have to be integrated through one upper layer (integration) ontology to build the relation between two different domain ontologies.

The framework should provide a mechanism to support ontologies from different domains in order to extract correlated knowledge from different domains. Moreover, the framework should provide semantic reasoning to get more relevant answer of the user's queries. It should take care of the ontology persistence, integration of the domain ontologies, should assist annotation process through integration ontologies and also should take care of persistence of the knowledge base as well as reasoning capabilities on the integrated knowledge base. It should support multilingual ontologies to allow multilingual knowledge extraction and reasoning results for the cross-domain knowledge. It should also support standard ontology formats for persistence, reuse and exchange between other components. In summary, it should take care of all necessary tasks to maintain the new integrated ontology and knowledgebase for its possible usage by the other components of OSAPIR framework.

4.3 Ontology Management Framework

The ontology management framework is a core part of OSAPIR framework. Ontologies, which represent the conceptual structure of the knowledge, are required to be managed in order to be accessed, reused, reasoned across multiple ontologies and many more management tasks. In our proposed ontology management framework, we proposed an ontology integration model that integrates a set of given domain ontologies in order to integrate cross-domain knowledge which is one of the main goals of OSAPIR framework. In this integration model, these different ontologies are brought into relation by the upper layer integration ontologies which map the possible relations among the domains. These integrated domain ontologies via upper layer integration ontology allow integration of such different knowledge sources. This integration ontology model addresses the problem of cross-domain integration of the ontologies which will be discussed in the upcoming sections of this chapter. With this integration ontology in place, more management tasks are expected to enable the efficient use of these ontologies by the framework. The proposed framework defines the relationship across these domain ontologies using the additional integration ontology which helps building the relations between the knowledge bases of different domains and allows reasoning over the produced knowledge base.

The framework is designed as domain independent which means it can be tailored to support any given domains for knowledge integration. The integration ontology can be created for the given domains and with some customizations for query processing and configurations the framework can support ontology management of those domains. It can then be consumed by the other components of OSAPIR framework for annotation of Web sources and support semantic reasoning. The framework supports the multilingual knowledge management for annotating and reasoning on extracted multilingual knowledge from different sources. There are different components to serve other management tasks which are required by the other components of OSAPIR framework. Figure 4 shows the architecture of the management services framework for integrated ontologies and knowledgebase.

4.4 Ontology Integration Model

In order for the framework to integrate the heterogeneous cross-domain knowledge, ontology is required to map these different ontologies. There are couples of approaches which either map the ontologies or integrate them into one core ontology. In case of two different domain ontologies, it’s not a good idea to integrate them into one because that will be a violation of the whole idea as they will be no longer two different domains. Our objective is to identify the relations among different domain ontologies and bring them into an agreement using the upper layer ontology. This upper layer ontology (integration ontology) relates the two domains with each other and can possibly capture any annotation required for information extracted from these domains.

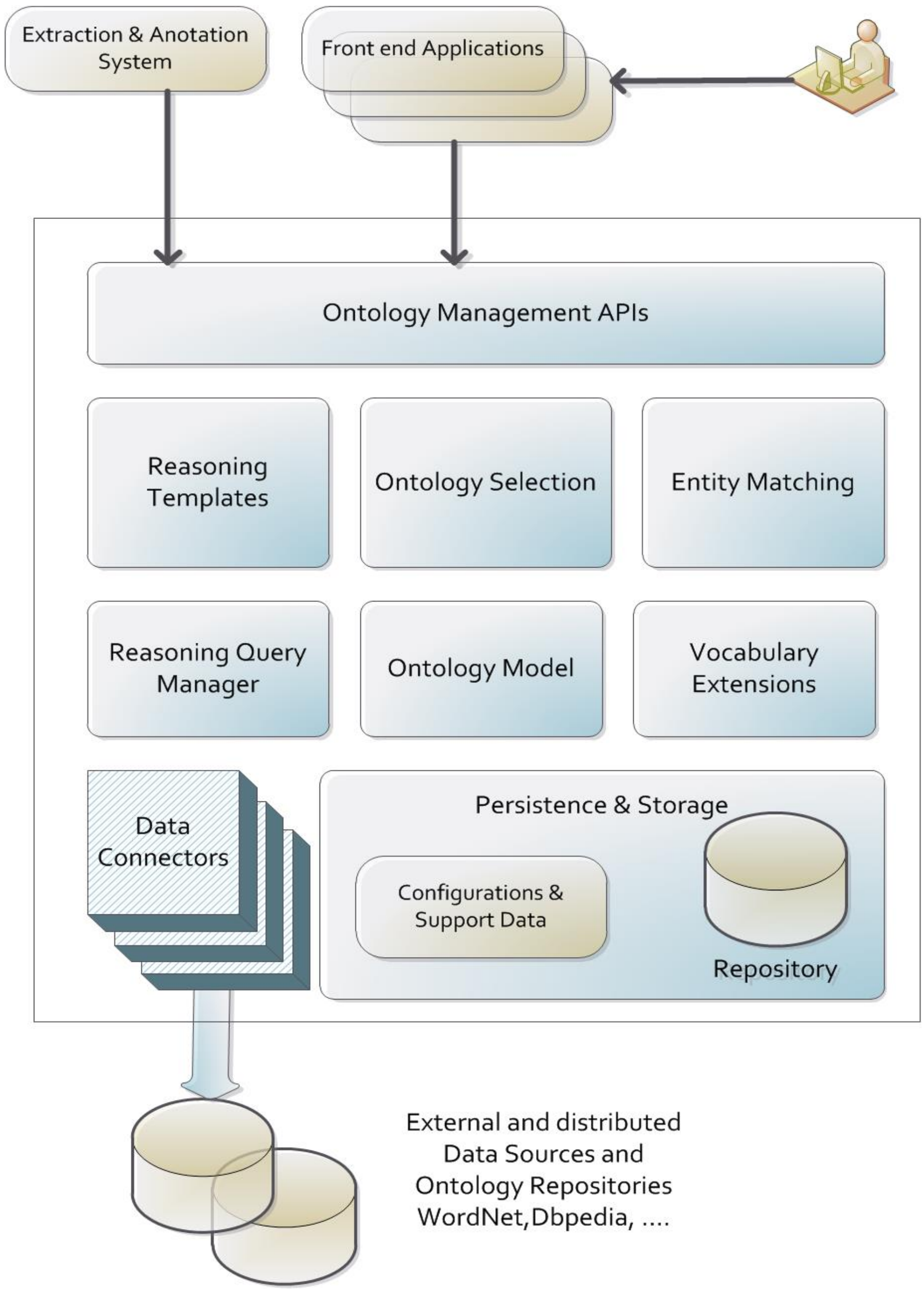


Figure 4 - Ontology Management Framework

The integration ontology model is the heart of the framework as it makes it possible to relate all these separate ontologies to each other for extracting the data as well as reasoning on the data. Figure 5 shows the conceptual representation of Integration Ontology Model. The document, sentence and relation are the main concepts of integration ontology which are mapped to lower domain ontologies based on the possible relations of such domains. All the real-world relations which can link the Ontology A to Ontology B should be created as functional properties to map to these domain ontologies. The document ontology is used to maintain the reference of the extracted information from Web resources.

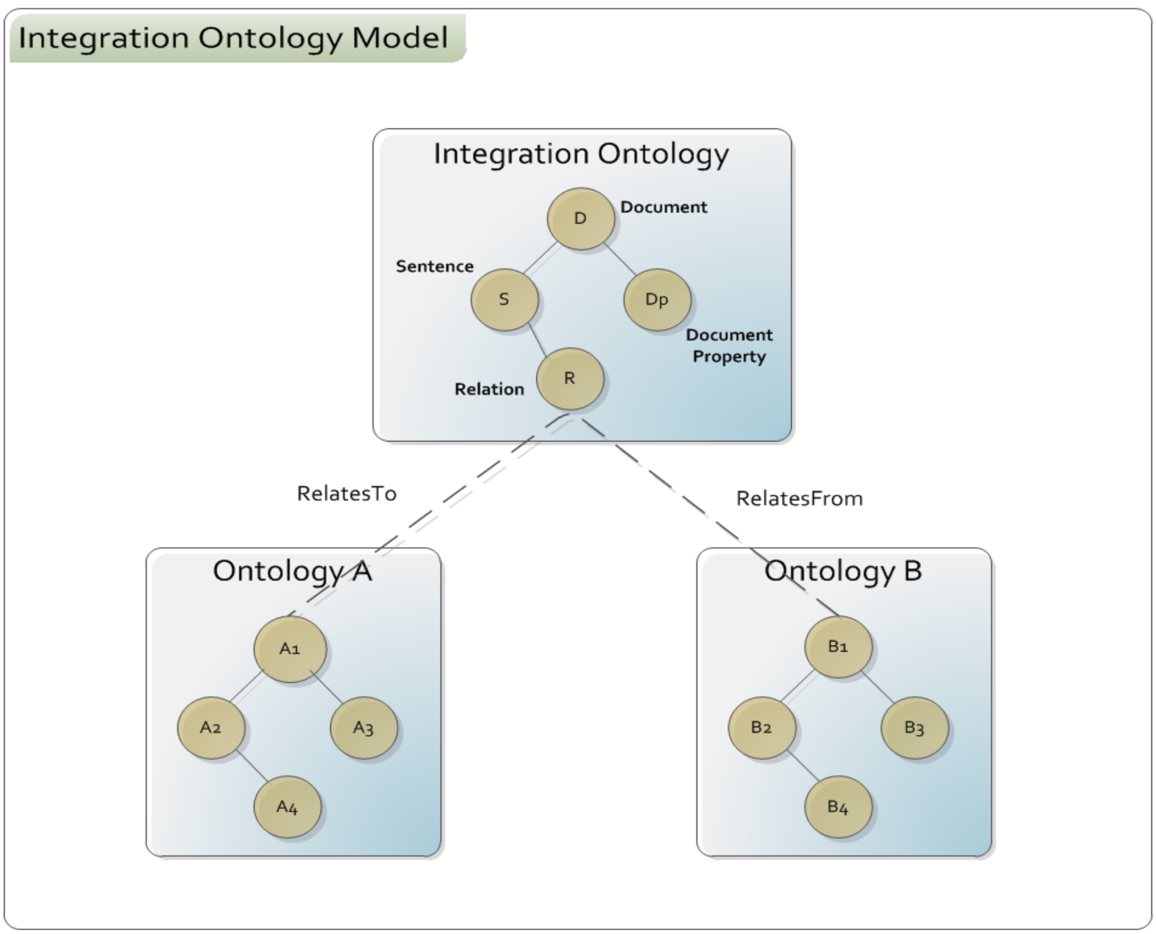


Figure 5 - Integration Ontology Model

For example, in the case of food and health domains where the user might ask about foods which are not recommended for certain type of diseases. Such questions pose a gap where there is no such relation among the ontologies exists and that means we cannot semantically reason since ontologies have no relation. We can bridge this gap by creating integration ontology which links these domain ontologies with all possible relations among them without actually modifying the domain ontologies so that the question can be reasoned. Figure 6 shows the example of integration ontology in which relation ontology declaration is specified with link of food and nutrition domains with health domain.

```
<owl:Class rdf:about="http://www.kfupm.edu.sa/ontology/integration/Relation">
   <rdfs:label rdf:datatype=http://www.w3.org/2001/XMLSchema#string>Relation</rdfs:label>
   <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
 </owl:Class>

  <owl:ObjectProperty rdf:about="http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo">
   <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/health/Health"/>
   <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
   >Relation</rdfs:label>
   <rdfs:domain rdf:resource="http://www.kfupm.edu.sa/ontology/integration/Relation"/>
  </owl:ObjectProperty>

<owl:ObjectProperty rdf:about="http://www.kfupm.edu.sa/ontology/integration/cause">
   <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/health/Disease"/>
   <rdfs:domain rdf:resource="http://www.kfupm.edu.sa/ontology/integration/Relation"/>
   <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
   >Relation</rdfs:label>
  </owl:ObjectProperty>

<owl:ObjectProperty rdf:about="http://www.kfupm.edu.sa/ontology/integration/hasNegativeEffectFrom">
   <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/food/FoodItem"/>
   <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
   >Relation</rdfs:label>
   <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/food/Nutrient"/>
   <rdfs:domain rdf:resource="http://www.kfupm.edu.sa/ontology/integration/Relation"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:about="http://www.kfupm.edu.sa/ontology/integration/prevent">
   <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/health/Disease"/>
   <rdfs:domain rdf:resource="http://www.kfupm.edu.sa/ontology/integration/Relation"/>
   <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
   >Relation</rdfs:label>
</owl:ObjectProperty>
```

Figure 6 - Sample Integration Ontology for Food, Nutrition and Disease domains

Integration ontology as proposed by OSAPIR framework, allows us to integrate or map these ontologies by using possible relations among them in the real world. Foods that are harmful to certain health conditions should not appear in personalized results which can be accomplished with this integration ontology. Such results could be semantically answered by linking health and food ontologies with such relations e.g. Recommended, prevents, good for, causes, treats. These relations could be used as properties in the upper layer ontology to link these lower domain ontologies.

4.5 Distributed Domain Ontologies Support

In addition, the ontologies from different domains need not be stored together in the same repository. The framework supports distributed repositories based on the repository abstraction layer provided by the Sesame which is the core part of the framework implementation. Due to the flexible architecture support of Sesame, it can work with not only locally stored RDFs but also with any network based service that allow and supports query RDFs for retrieving and storing [40]. In such way it works with distributed repositories where ontologies are not just locally stored but in fact could be stored in the distributed repositories. In the proposed framework, we have an approach of integration of cross domain ontologies, where actually a mapping is created in the integration ontology which refers to the domain ontologies independently with respect to the relation among them. In the proposed approach integration, ontologies are actually linked via integration ontology and exist independently which can be stored in distributed repositories. The proposed framework with the aid of Sesame, it supports to inquire these different ontologies stored in distributed repositories and allows us to reason the information in the interoperable manner out of the distributed repositories.

4.6 Ontology Management Tasks

There are many viable tasks that ontology management systems in general have implemented which are mostly common basic tasks and represent the core functionalities. Some of the tasks are ontology editing, ontology management APIs, reasoning support, persistence & storage mechanism and interface for querying the knowledge base. In our framework, we are providing the management tasks for integration ontologies and reasoning from integrated knowledge sources. The proposed ontology management framework is designed to assist the extraction and annotation processes and then store the annotated information in the internal knowledge base so that it can be used for reasoning to answer various queries related to given domains. The framework supports various ontology management tasks like importing and reusing of other ontologies. It also provides versioning support to ontologies for storing and querying the knowledge. It also supports aligning and mapping between ontologies using the integrated ontologies. It provides different management tasks like Ontology Selection, Vocabulary Extensions, Entity Matching, Knowledge Persistence, and Reasoning Query Manager. As the main goal of the OSAPIR framework is to semantically answer user's queries, so that ontology management framework should support the inference across multiple ontologies to allow reasoning for cross domain information sources. This framework also provides predefined Reasoning Query Templates to help in achieving the main goal of the OSAPIR framework.

4.6.1 Ontology Selection

When dealing with multiple ontologies either cross-domain ontologies or multilingual independent ontologies, there is a need to manage such ontologies by central component which can provide the other components of OSAPIR framework with required ontologies in different formats. During the extraction and annotation process by annotation component [37] of the OSAPIR, ontologies are required to be accessed individually or partially of any domain or language. The framework manages these domain ontologies

Input: TargetOntology (O_t_N), Algorithm ($Alg_{1)}$, Target Language ($L_{t)}$
Output: Complete/Partial Ontology with enriched vocabulary ($Or_{)}$

```
Procedure
begin
   O_n[] = ListOntologyNames(L_t)
   foreach (O_n[])
   begin
         if (O_n[index] Equal O_t_N)
            O_t = LoadOntology(O_n[index])
         endif
   end
// Strategy Pattern based user provided algorithm to search
   O_t_matched_index = algorithm_search(O_t,Alg1)
   if (O_1_matched_index Not Equal To NULL)
      O_r = LoadPartialOntology(O_t)
   else
         O_r = O_t
   endif
   O_r = LoadVocabularyExtensions(O_r)
   return O_r
end
```

Figure 7 - Ontology Selection Algorithm

with different languages and the integration ontology. It allows other components such as annotation component [37] of OSAPIR framework to access the ontology on demand for any domain in any language. Figure 7 shows the pseudo algorithm used in the ontology selection process which shows that a particular ontology (O_t) is selected based on the provided language ($L_{t).}$ Algorithm (Alg_1) is provided for the searching process to search the required concept tree in the ontology tree. Once the ontology (O_r) is selected, it is enriched with vocabularies to improve the search space for the annotation process.

In the process of the ontology selection the searching mechanism is also needed if particular concepts are required to be searched. Ontologies are accessed by the annotation component and searched by the algorithms of their choice. The component is designed to support different algorithms to be used for searching based on the need. A strategy pattern of software design patterns is used to enable dynamic use of algorithms at runtime. Algorithms can be written as required by the components. The component also supports the fetching of the additional vocabularies from local or any other distributed linguistic or ontology source like DBPedia [41] by the use of Vocabulary extension. These vocabularies could be customized and manually provided or could be based on external service like i.e. WordNet [42]. Figure 8 shows the usage scenario of the Ontology selector.

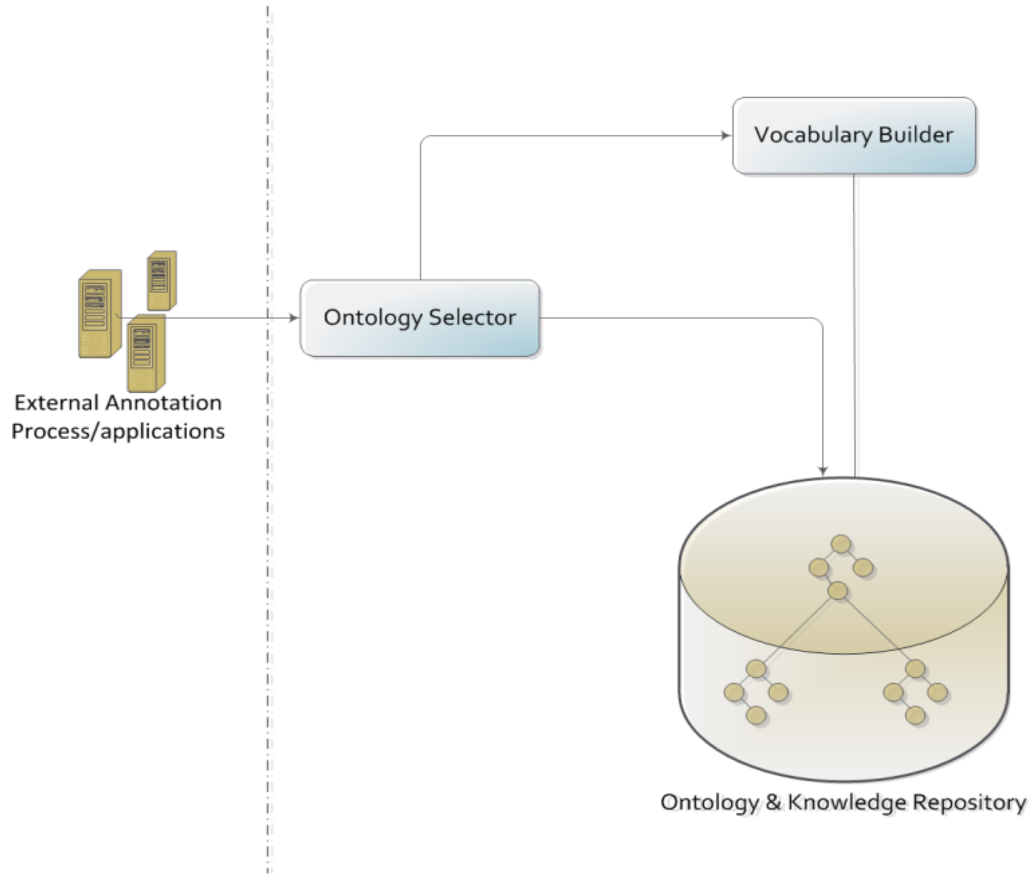


Figure 8 - Ontology Selector

4.6.2 Vocabulary Extensions

In order to improve the process of text matching either for extraction or query processing in different languages, a versatile vocabulary management is needed. A vocabulary could be either fed by language experts or provided by external dictionary services. Vocabulary Extensions component supplements vocabulary of ontology by providing synonyms when required by the caller for matching during text processing. During the lexical analysis of text processing task, either for annotation purposes or user's query processing, additional synonyms are required to assist matching the words to be mapped into correct ontology concepts. Figure 9 shows the architecture of vocabularies extensions provided by the framework.

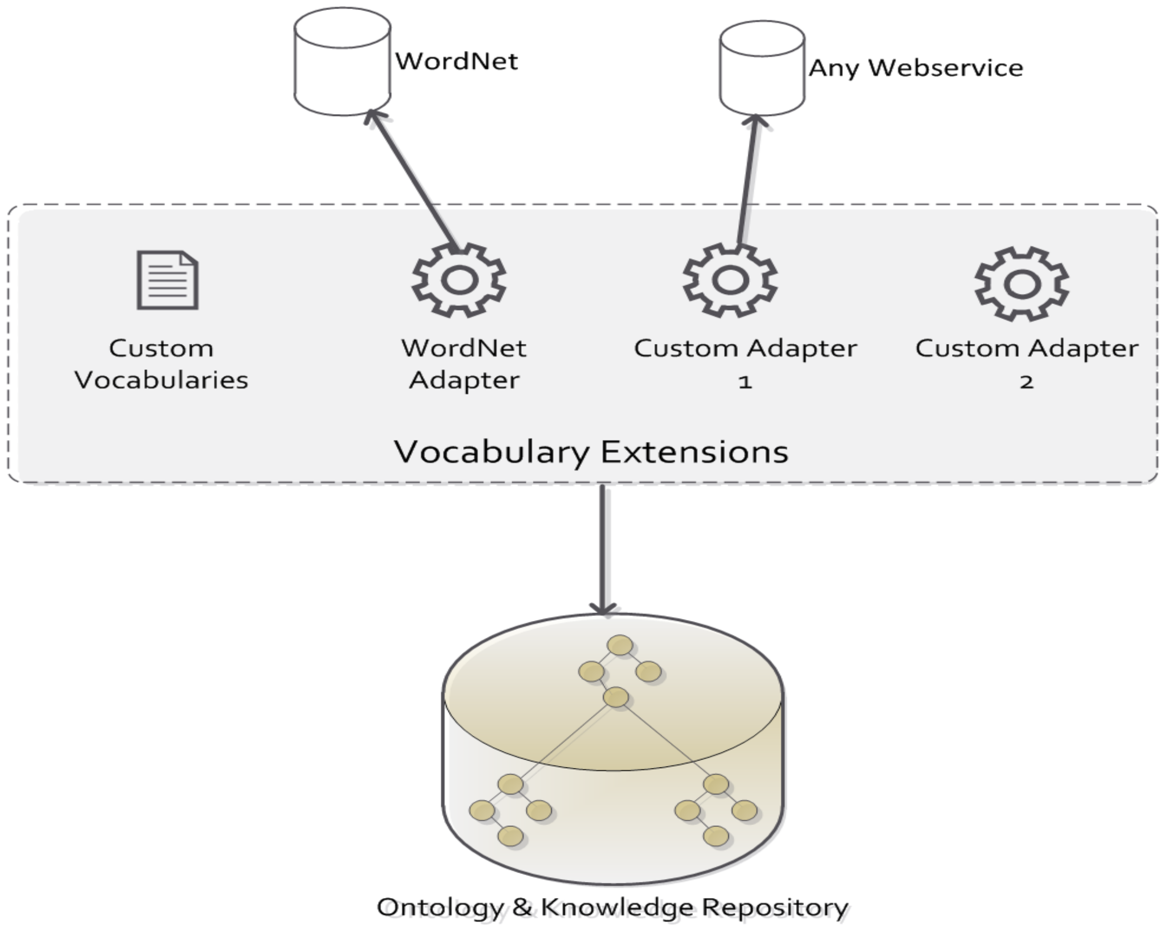


Figure 9 - Vocabulary Extensions

The Vocabulary Extensions has been developed to support manually provided synonyms or dictionary along with automated dictionaries or thesaurus support which could also be based on the some external distributed linguistic database. It has been modeled based on adapters pattern for the implementation where different adapters can be written to provide the synonyms at real-time by the components of OSAPIR. So we have integration to WordNet [42] which is achieved by developing an adapter and more adapters can be developed to support additional web services for dictionaries or thesaurus to support the task. Newly created adapters can be configured in the framework through configuration file as available in the Appendix I.

4.6.3 Entity Matching

This component is responsible for finding the named entity in the pre-selected ontology or in all ontologies if required. The idea is to assist the other OSAPIR framework components in finding the best match where contextual information with lexical

Input: SearchTerm (T_1), Algorithm ($Alg_{1)}$, Language ($L_{1)}$
Output: List of found Named Entities in all ontologies (NE)

```
Procedure
begin
   On[] = ListOntologyNames(L1)
   foreach (On)
   begin
          Os = LoadOntology(On[index])
          Os = LoadVocabularyExtensions(Os);
          NE = SearchTerms(T1, O, Alg1)
        if (NE Not Equal NULL)
            NE[] = NE[] + NE;
        endif
   end
   return NE []
end
```

Figure 10 - Entity Matching Algorithm

processor can help it to judge the best match. The entity matcher also makes use of vocabulary extensions provided by the vocabulary extensions component. As mentioned earlier it's interfaced with external vocabulary service to increase the search space for the ontology in a given language. Figure 10 shows the entity matching algorithm for searching the terms in all ontologies and returns all the entities matched from different ontologies. It also utilizes the vocabulary extensions which can be set by the caller of the method dynamically. Based on the Language (L_1), it loads all the ontologies and loops through each ontology to search the term (T_1) to find all named entities (NE) in all ontologies (O_n) using the provided algorithm (Alg_1). It returns all the matching named entities resolved to ontology concepts in different ontologies. It allows the semantic query manipulation component [38] to analyze the query with context and to pick the right named entity for mapping. User query manipulation component [38] of the OSAPIR framework uses this component for the query understanding process and enriching the query with semantics which are required by the reasoned query manager to reason. This component helps query manipulation component to map the user's input to the ontology.

4.6.4 Knowledge Persistence

This management task allows adding more annotated and validated knowledge to the system. A new or updated ontology version of any domain can be updated through this interface. It takes care of the validation of knowledge data and ontologies in terms of the ontology model rules. Figure 11 shows the process of the knowledge persistence in the framework.

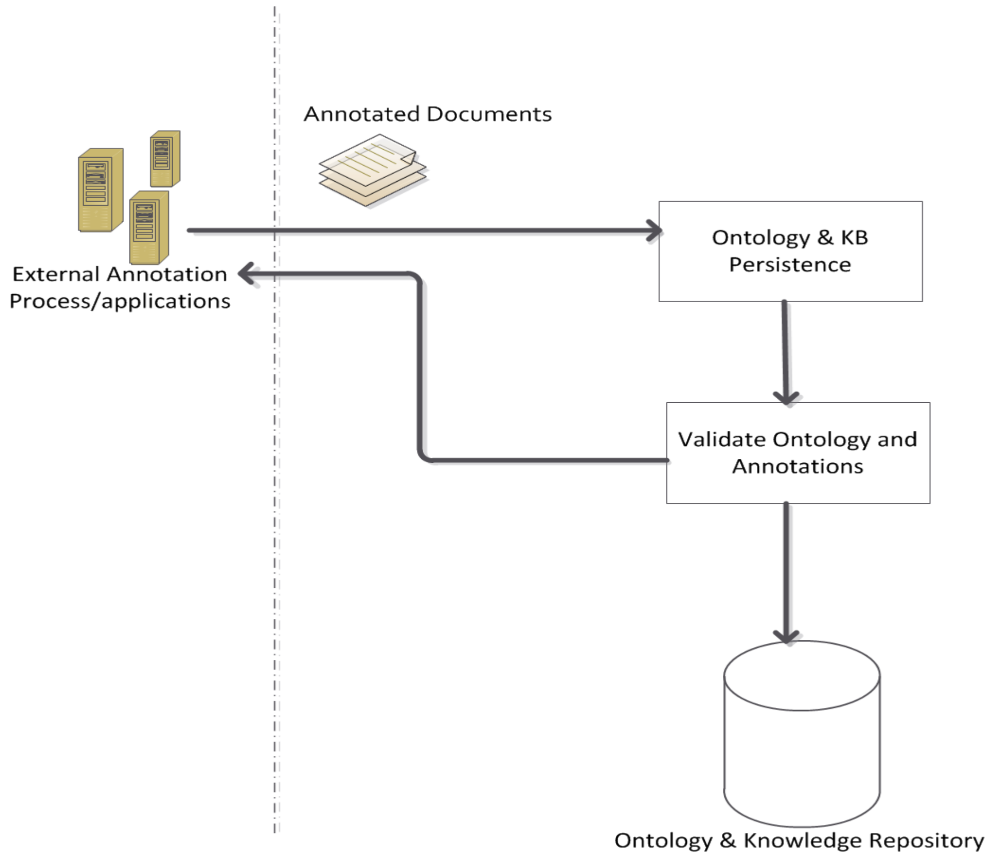


Figure 11 - Knowledge Persistence

4.6.5 Ontology Versioning

Ontology versioning is an important aspect of the ontology management. The framework supports the versioning of ontologies as well as versioning of the documents being annotated. Each annotated document comes with version and refreshed versions are updated for the document. Ontologies evolve with time as the domain experts are continuously working and improving ontologies. Ontologies versioning are supported with backward compatible ontologies and fully compatible ontologies. With fully compatible ontologies, the ontologies may change but semantic interpretation of ontologies remains same while only syntactical representation of descriptions is changes.

With backward compatible ontology where the data interpretation of the ontology is same as if using the old ontology is also compatible.

4.6.6 Reasoning Query Manager

Reasoning Query Manager is a major part of the framework, it provides the capability to infer from cross-domain annotated knowledge, i.e. searching for the benefits for food that support a given health condition or answering the queries with the reference of the sources where the data has been extracted from.

Input: Named Entities (NE_n), Inquired Mapping/Relations (M_n), Query Filters (F_n), Language (L)
Output: Multiple Annotation Results (R_n)

```
Procedure
begin
   S_sparql = InitializeSparqlContext();
   D_n = AnalyzeRelation(NE_n, M_n)    // D_n  are the identified domains
   O_n = ListOntologies(D_n)
   foreach (O_n)
   begin
        S_sparql += ApplyOntolgyContext(O_n[index],M_n[index))
   end
   S_sparql = ApplyFilters(S_sparql,F_n)
   R_n = S_sparql.execute()
   return R_n
 end
```

Figure 12 - Reasoner Engine execution algorithm

Figure 12 shows the algorithmic approach to analyze the input from query manipulation components [38]. The components provide named entities (NE_n), the relations among the entities inquired by the queries (M_n) along with any filters based on the user preference.

The algorithm analyzes the named entities and relations (NE_n, M_n) to find the relevant domains (D_n) to inquire. It loads the ontologies (O_n) for those domains (D_n) and builds SPARQL queries. It then applies the provided filters (F_n) to include or exclude the results and finally executes the updated SPARQL to get the results.

The reasoning query manager works on top of the proposed ontology model and uses the interpretation to deduce facts. The query engine APIs allow to build queries dynamically at runtime based on the needs. The framework also supports creation of design time reasoning templates which maps user queries to SPARQL query template. Templates may not always be sufficient and sometimes there is a need to make a query dynamically and Reasoning Query Manager APIs enables us to make the dynamic queries. Figure 13 shows the process of query processing in the framework.

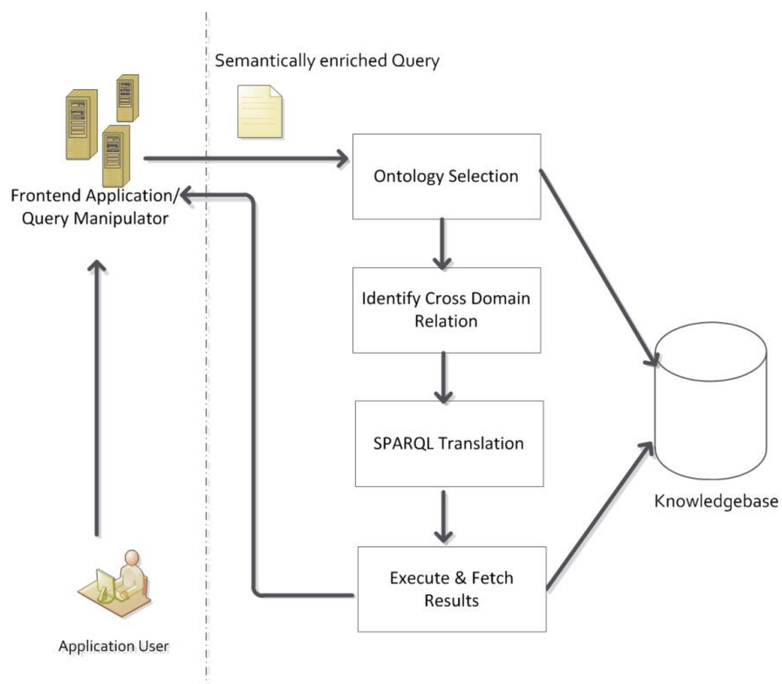


Figure 13 - Query Processing

4.6.7 Reasoning Query Templates

As the knowledge of different sources is integrated by the framework and query engine APIs allows dynamically querying the integrated knowledge. This Reasoning Query Templates is a value added feature to build design time query templates based on the possible relations among the domains ontologies as defined in the integration ontology. In these query templates, Query Engine APIs are used to provide predefined queries at the design time which can be used by the user query manipulation component [38] of the OSAPIR framework. These design time predefined reasoning templates allow the front-end application [38] to build more functional widget for the result pages. As mentioned above, it uses the reasoning querying manager APIs to run the queries and to generate the response. The difference is that queries in this case are not actually dynamic in terms of relations and are available to bind the input and output directly for use.

4.6.8 Multilingual Support of Ontologies

Ontology development has been of more interest in the research and different approaches are being studied for supporting multiple languages. This management framework implemented three types of the ontology models to provide multilingual support for ontologies. It covers standards that most of the ontology developers are following as of today. Ontologies can include labels for each language by default, i.e. OWL supports having labels in different languages. The other option is to have an ontology for each language in a given domain. These ontologies are identical ontologies but in different languages and appropriate ontology is selected when required for processing by the ontology selector process [5].

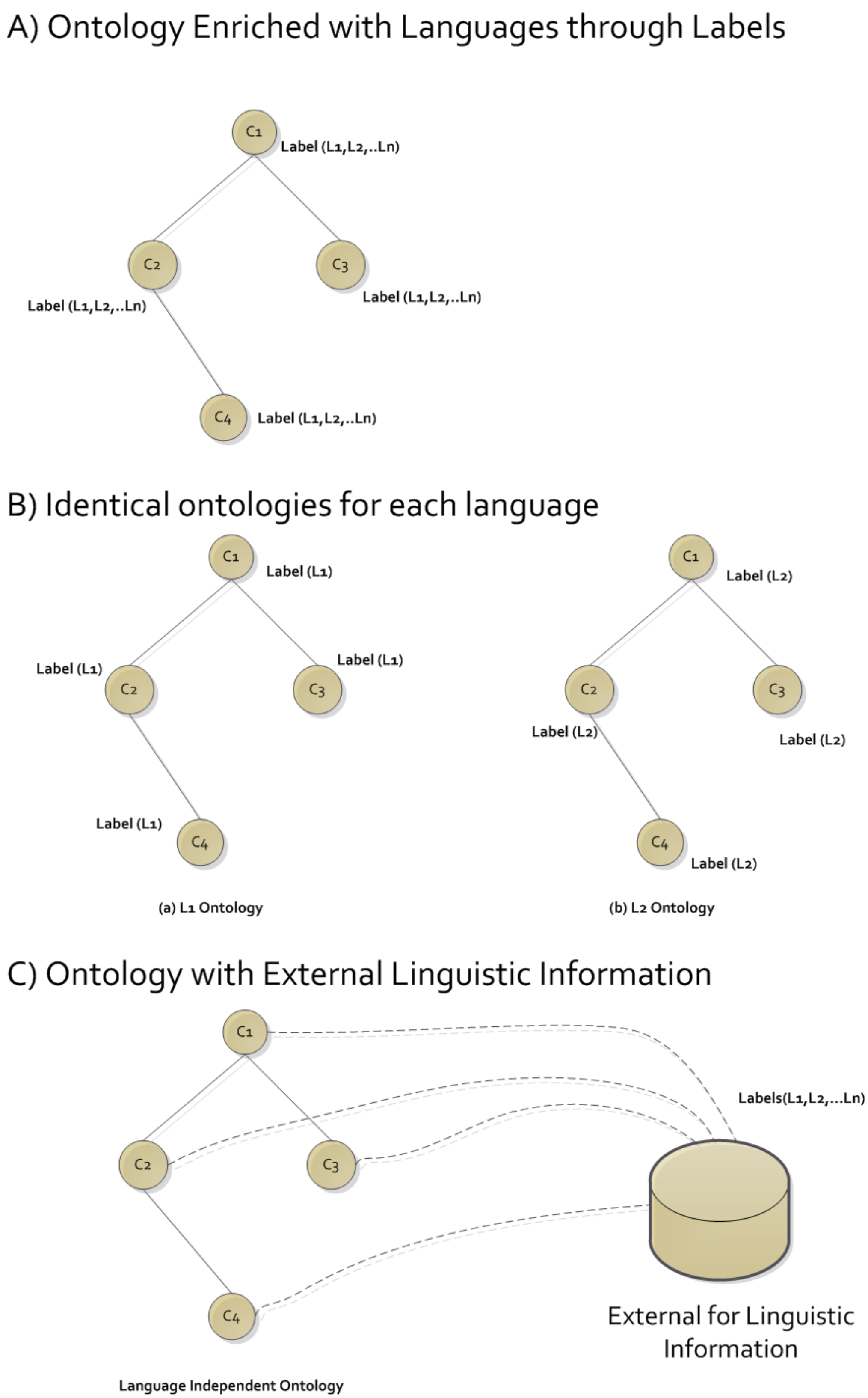


Figure 14 - Multilingual Ontologies Support

The third approach we implemented is by keeping the ontology conceptual knowledge and linguistic knowledge separate as vocabulary extensions which are provided during processing by vocabulary extension component. Vocabulary for different languages can be built manually by language experts or vocabulary extension adapters can fetch them from external Web services dynamically similar LIR model approach [43].

Figure 14 shows three approaches we have supported in our framework. The approach (A) is the use of the ontology with embedded labels for each language. The ontology itself is enriched with all the target languages. This is the widely used approach with mostly available ontologies. As in the approach (B), independent language ontologies which are actually identical to each other in different languages, are supported as well. The third approach (C) is to have only one ontology and keep all linguistic information in the external data source.

4.6.9 Ontology Management APIs

A framework can be considered a basic foundation for any application development process which provides the grounds for the development in a certain platform. A framework should ease the development process and provide access to the internal resources and any external libraries if used by the framework. Our ontology management framework adapts all the qualities expected from a software framework. All services provided by the Ontology Management framework are exposed as application programmable interfaces (API) which allow using the framework to build complete end to end portal. The framework APIs provide access to these different components like Ontology Selection, Entity Matching, Vocabulary Extensions, Knowledge Persistence and various others tasks which helps the information processing by OSAPIR components. The other components of OSAPIR framework use these APIs for the integration of whole framework. There are different APIs packages available for the use by these components. Figure 15 shows top level classes and interfaces to access the APIs and it gives an over view of available management tasks.

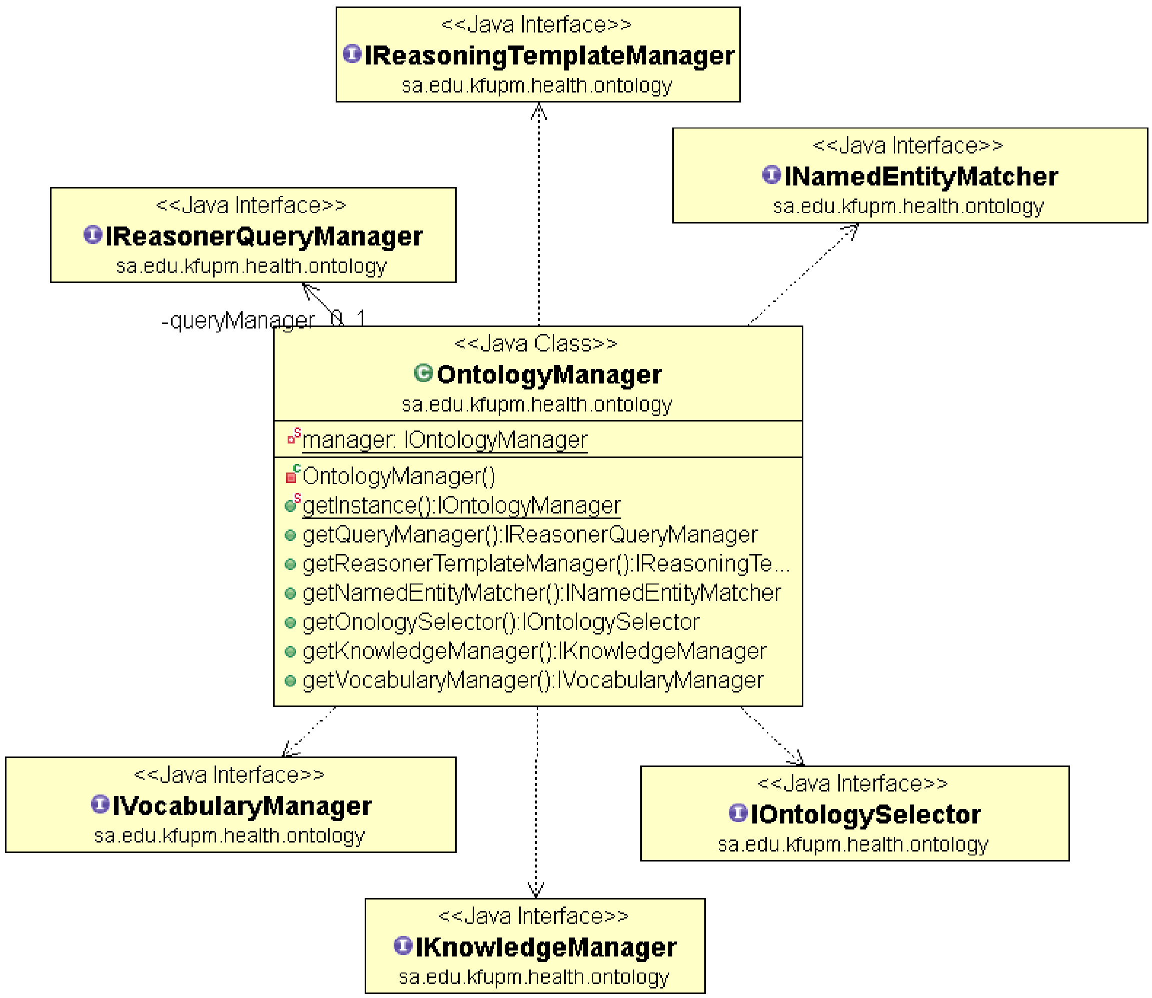


Figure 15 - Ontology Manage API (Facade Pattern)

CHAPTER 5

A CASE STUDY: PROTOTYPE IMPLEMENTATION

In the prototype implementation of the framework, we considered the case of health, food and nutrition domains where a user can inquire about any food or health related information. The inquiries will be semantically understood to provide more relevant and semantic answer. As a first step towards the implementation of the framework it is necessary to bring the ontologies of the chosen domains and to evaluate them from the perspective of OSAPIR framework. Evaluation of these ontologies is based on the need to extract the relevant information from different heterogeneous sources. Predefined criteria are used to evaluate or adapt these ontologies to meet the objective of the OSAPIR framework. Ontology management framework which is based on APIs can be configured by extending certain classes to tailor it to the selected domains. The following sections of this chapter covers the criteria used to evaluate and adapt the ontologies, integration of ontologies and how the framework APIs are implemented to be configured for the food, health and nutrition domains.

5.1 Ontologies

Ontologies are the network of concepts that represents knowledge of a given domain using shared terminologies for the types, properties and relation among different concepts. Each domain has its representing ontology which adds semantics to its knowledge. Our case study implementation considered based on the importance of these domains for general user today. Food and health domains are full of rich knowledge and ontologies for these domains are continually being developed independently by the experts from each domain. These ontologies have no relation to each other while in the real world these domains have relation to each other. The knowledge relating these domains is of great interest to the users concerned to their health conscious diet.

In the following sections we cover different ontologies used for the integration of food, health and nutrition information to bring relation among these knowledge sources and how these integrated ontologies are managed by the framework to achieve the targeted goal.

5.1.1 Criteria for Ontologies

We selected the above ontologies and extended them wherever required to match them with the needs of our framework in meeting the objectives of knowledge integration. We evaluated the ontologies based on varying criteria for these ontologies. Food and Nutrition ontologies were evaluated on the source of the knowledge as part of the ontology. We gave priority for ontologies which were based on the USDA database for food and nutrition. Other criteria like how rich is the ontology in terms of hierarchy and grouping within the domain, vocabulary richness, synonyms availability, multilingual support were used to select the ontologies then improved if required.

5.1.2 Food Ontology

AGROVOC [32] and FOODS [33] are different food ontologies that were evaluated based on the criteria of selection. These ontologies were not selected as they are not aligned with USDA [31] food and nutrition information. The ontology for food is adapted from semantic diet as their ontology for food is based on the USDA database for food items and classifications of the food groups. Ontology is available in English so we added the translation of the ontology in Arabic to have a test case of multilingual support. This ontology is just one main concepts of FoodItem and all the food items instances belong to it which are around 9000. The classification of FoodItem is handled through FoodGroup Concept. FoodGroup in the food ontology has hierarchy of food groups. Almost each instance in the ontology has two type concepts, FoodItem and any child concept of the FoodGroup. Figure 16 shows the English food ontology.

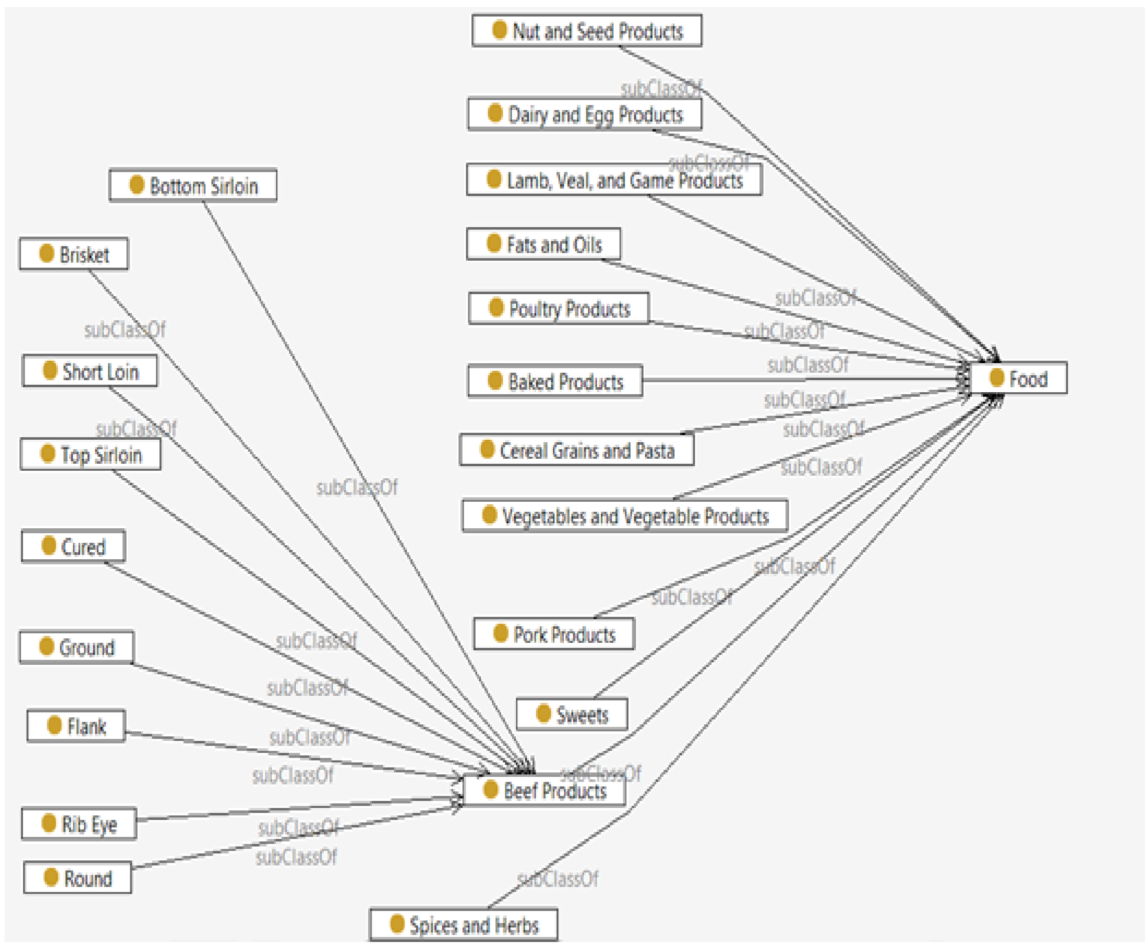


Figure 16 - Food Ontology

5.1.3 Nutrition Ontology

We adapted the nutrition ontology provided by Semantic Diet as starting point and extended it to meet some requirements. One main advantage of the nutrition ontology semantic diet is that it is also aligned with USDA [31] database and linked with food ontology mapping following the USDA [31] database. The Semantic Diet based nutrition ontology contains only one concept with 146 distinct nutrition elements with instances for all food instances.

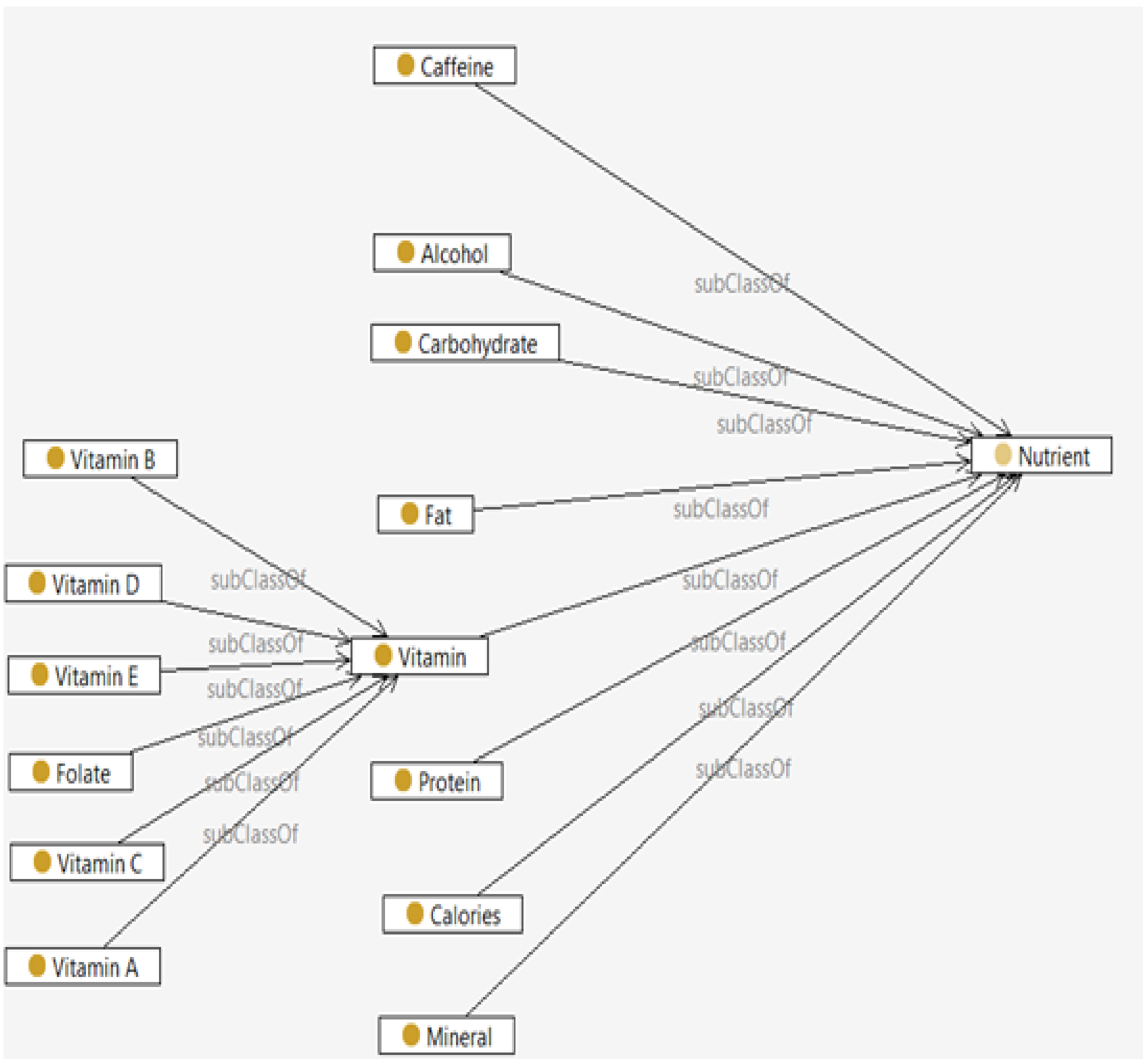


Figure 17 - Nutrition Ontology

We have extended the ontology to multi-levels in order to be able to capture the aggregation of nutrient in the same group. The multilingual support was achieved by embedding the translation of each concept as Arabic label to produce a multilingual ontology that covers English and Arabic languages at this stage. We maintain the same integration with food concepts as followed by USDA database. Figure 17 shows the English nutrition ontology used for the implementation while Figure 18 shows the Arabic version of it.

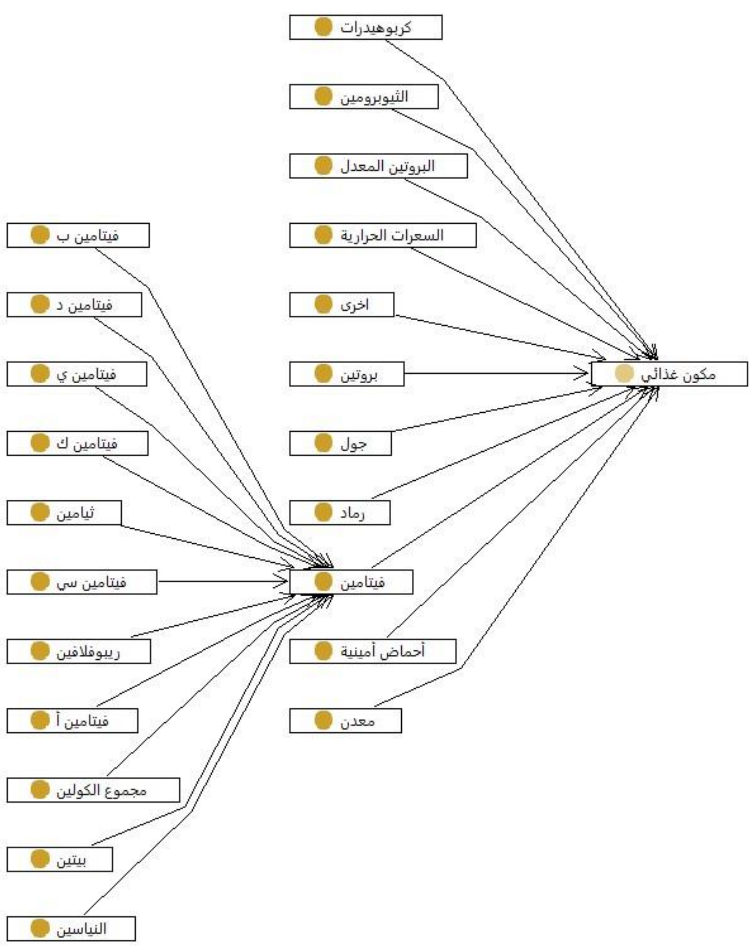


Figure 18 - Nutrition Ontology in Arabic

5.1.4 Recipe Ontology

Similar to food and nutrition ontologies, we have selected recipe ontology provided by Semantic Diet as starting ontology. The Semantic Diet recipe ontology contains only one concept without any instances. We have extended the ontology to multi-levels in order to be able to capture the aggregation of recipes in the same group. The multilingual support was achieved by adapting the approach of building the Arabic translation to produce a multilingual ontology that covers English and Arabic languages at this stage. Figure 19 shows the recipe ontology.

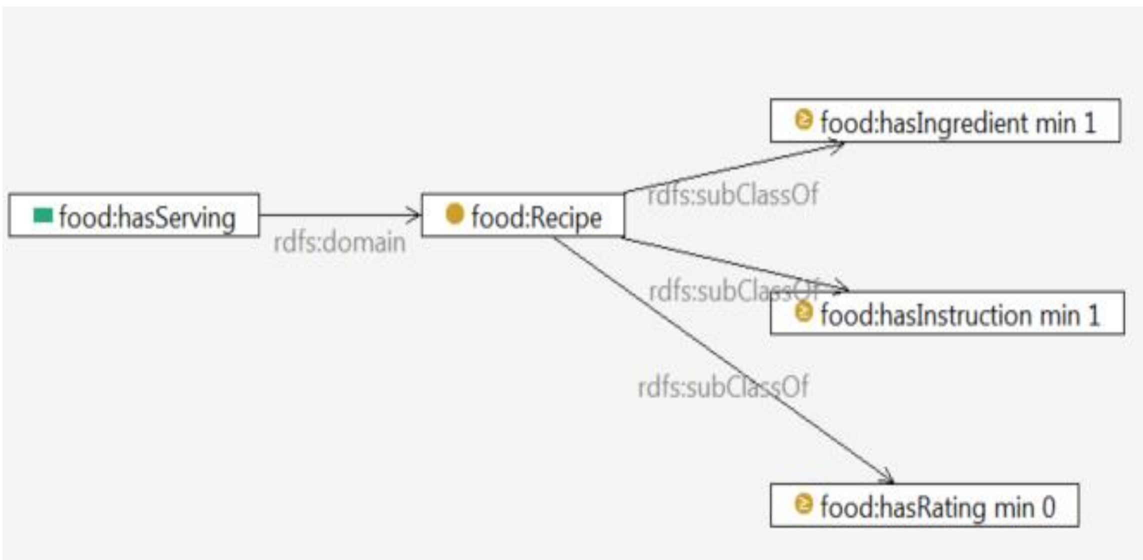


Figure 19 - Recipe Ontology

5.1.5 Body Part and Body Function Ontologies

These are small self-created ontologies for the proof of the concept. Any available ontology could be adopted but unfortunately no comprehensive ontology was available for Body Functions or Body Parts. These are small ontologies with 60 instances for body Functions and 163 for body parts. Figure 21 and Figure 21 show the ontology for body parts and body functions respectively.

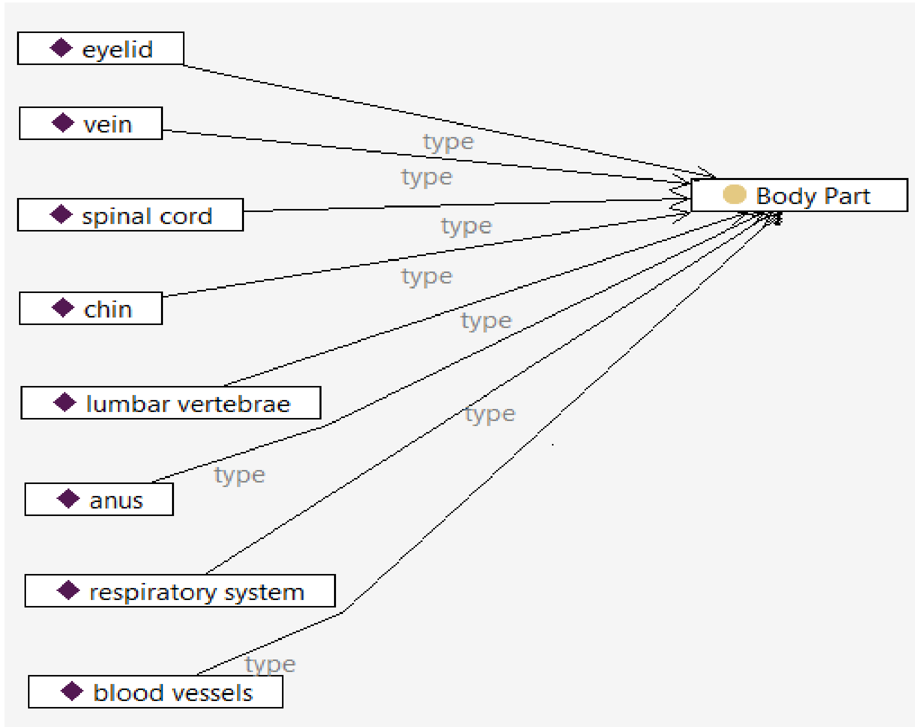


Figure 20 - Body Parts Ontology

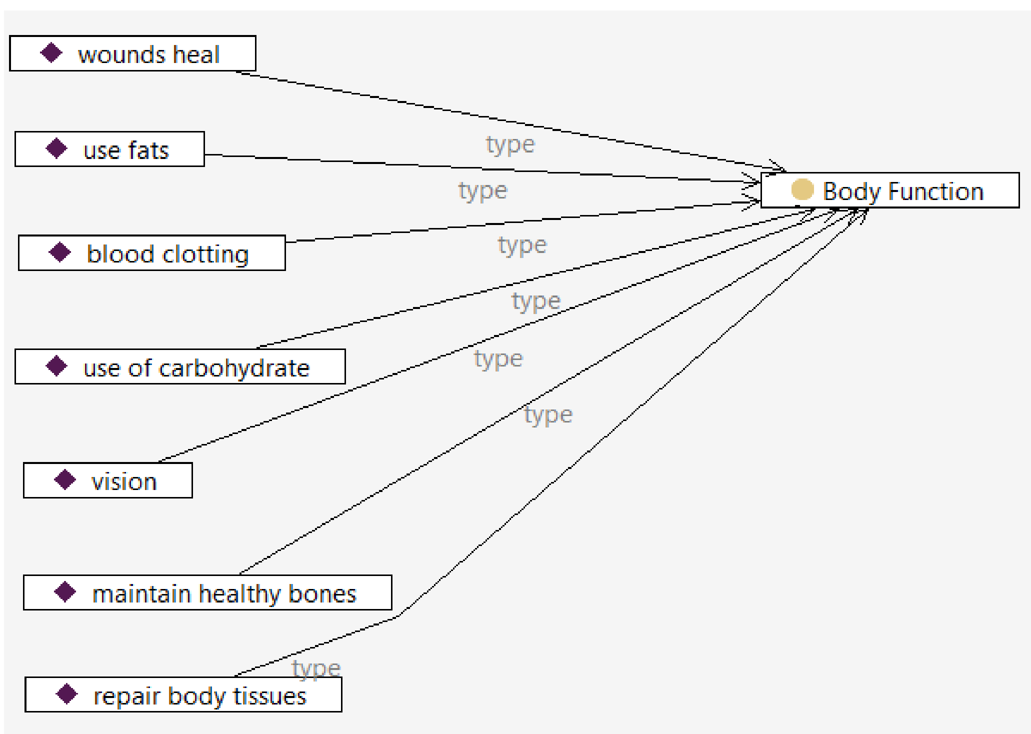


Figure 21 - Body Functions Ontology

5.1.6 Disease Ontology

The Human Disease Ontology [36] is selected because the concepts of this ontology are self-contained concepts unlike the ICD10 [34]. Having self-contained concepts are more suitable for text processing as the concepts are independent of the parent concepts and are meaningful enough to map to the contextual words during the text processing. In general, the disease ontology is a comprehensive vocabulary which is hierarchical in structure. For the description of ontologies in terms of metrics, it has 8685 concepts. It holds 15 properties and the maximum depth of the concepts is 14. On average, there are 3 child concepts for each concept while the maximum number of child concepts is 80. Figure 22 shows the disease ontology used in the implementation.

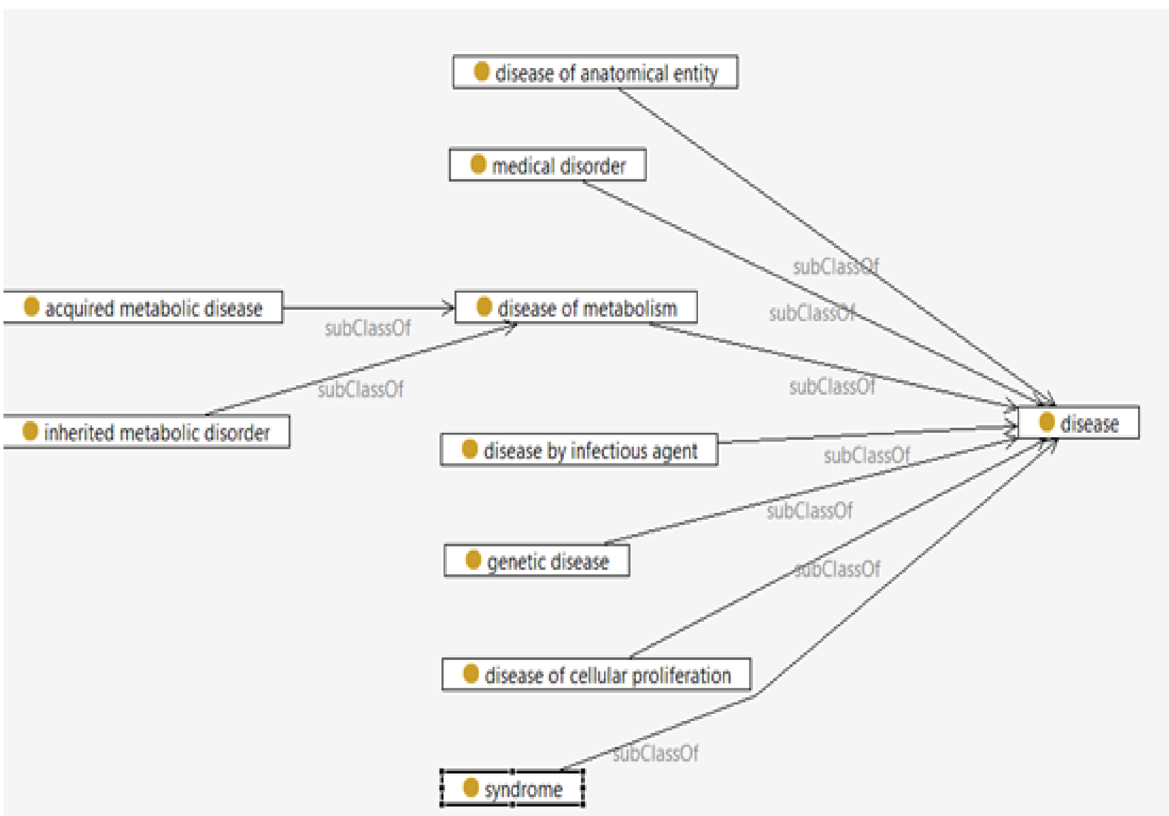


Figure 22 - Disease Ontology

5.1.7 User Profile Ontology

User profile ontology is utilized to capture different attributes of the user, i.e. preferences and medical information. Such information when available to OSAPIR framework allows it to personalize the responses. In order to capture the like and dislikes of the user observed during the searching process, ontology can be beneficial to store the user information as part of the knowledgebase. Representing the profile as ontology also makes it easier to filters out the responses related to food and health domains based on his preference. Figure 23 shows the user profile ontology.

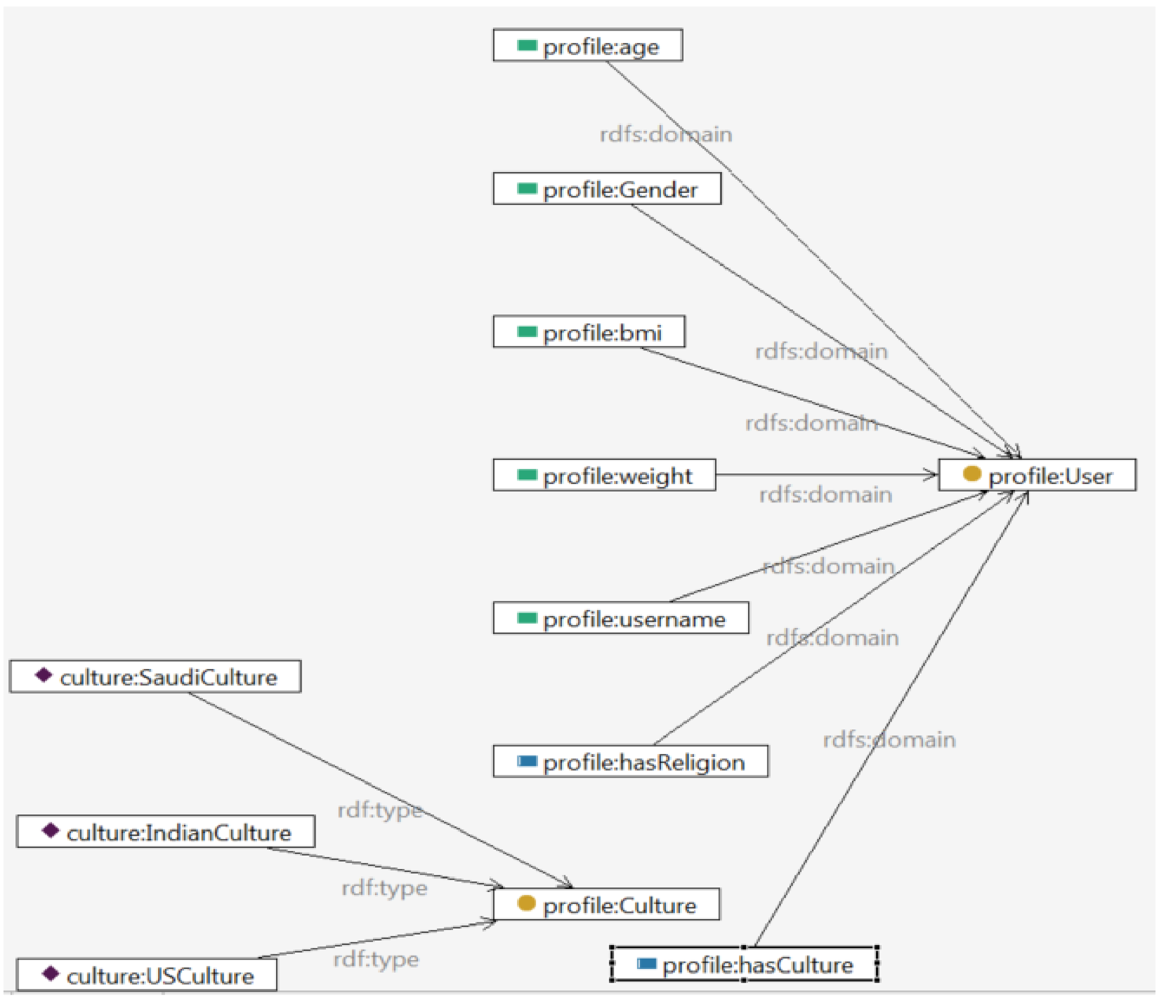


Figure 23 - Profile Ontology

5.1.8 Integration Ontology

The integration ontology is the heart of the framework as it makes it possible to relate all these separate ontologies to each other for extracting the data as well as reasoning on the data. It has three main concepts Document, Sentence and Relation. The document concept is used to maintain the reference of the extracted information from Web resources. The sentence concept is used to maintain the reference of the information in the document level of given Web resource. The relation concept contains the mapping between health and food ontologies to allow us to capture the relation between various Web resources. It captures the main annotation which defines the relation between the instances of different ontology concepts. Figure 25 and Figure 25 show the snippet of OWL description and graphical representation of integration ontology respectively. The relation holds attributes like hasPositiveEffectTo, prevents, treatsFrom, causes, etc., which map to food, nutrition, diseases, body functions and body parts. Document and sentence concepts hold the reference to the source of information that provides the mapping between these domains

The integration ontology is the upper layer ontology which integrates the health ontologies (disease, bodyParts, bodyFunctions) with Food (Fooditem and Nutrient) related ontologies. The mappings for integration ontology are identified by analyzing the common relations among the domains to be integrated.

```xml
<owl:Class rdf:about="http://www.kfupm.edu.sa/ontology/integration/Document">
  <rdfs:label xml:lang="en">Document </rdfs:label>
  <rdfs:label xml:lang="ar">مستند</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
<owl:Class rdf:about="http://www.kfupm.edu.sa/ontology/integration/Relation">
  <rdfs:label xml:lang="en">Relation</rdfs:label>
  <rdfs:label xml:lang="ar">علاقة</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
<owl:Class rdf:about="http://www.kfupm.edu.sa/ontology/integration/Sentence">
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  <rdfs:label xml:lang="en">Sentence</rdfs:label>
  <rdfs:label xml:lang="ar">جملة</rdfs:label>
</owl:Class>
<owl:ObjectProperty rdf:about="http://www.kfupm.edu.sa/ontology/integration/treatFrom">
  <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/food/FoodItem"/>
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Has Negative Effect From</rdfs:label>
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >له اثر ايجابي من</rdfs:label>
  <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/food/Nutrient"/>
  <rdfs:domain rdf:resource="http://www.kfupm.edu.sa/ontology/integration/Relation"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="http://www.kfupm.edu.sa/ontology/integration/preventTo">
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Has Negative Effect To</rdfs:label>
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >يؤثر سلبيا على</rdfs:label>
  <rdfs:domain rdf:resource="http://www.kfupm.edu.sa/ontology/integration/Relation"/>
  <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/health/Disease"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo">
  <rdfs:range rdf:resource="http://www.kfupm.edu.sa/ontology/health/Health"/>
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Has Positive Effect To</rdfs:label>
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >يؤثر ايجابيا على</rdfs:label>
  <rdfs:domain rdf:resource="http://www.kfupm.edu.sa/ontology/integration/Relation"/>
</owl:ObjectProperty>
```

Figure 24 - Snippet of Integration Ontology

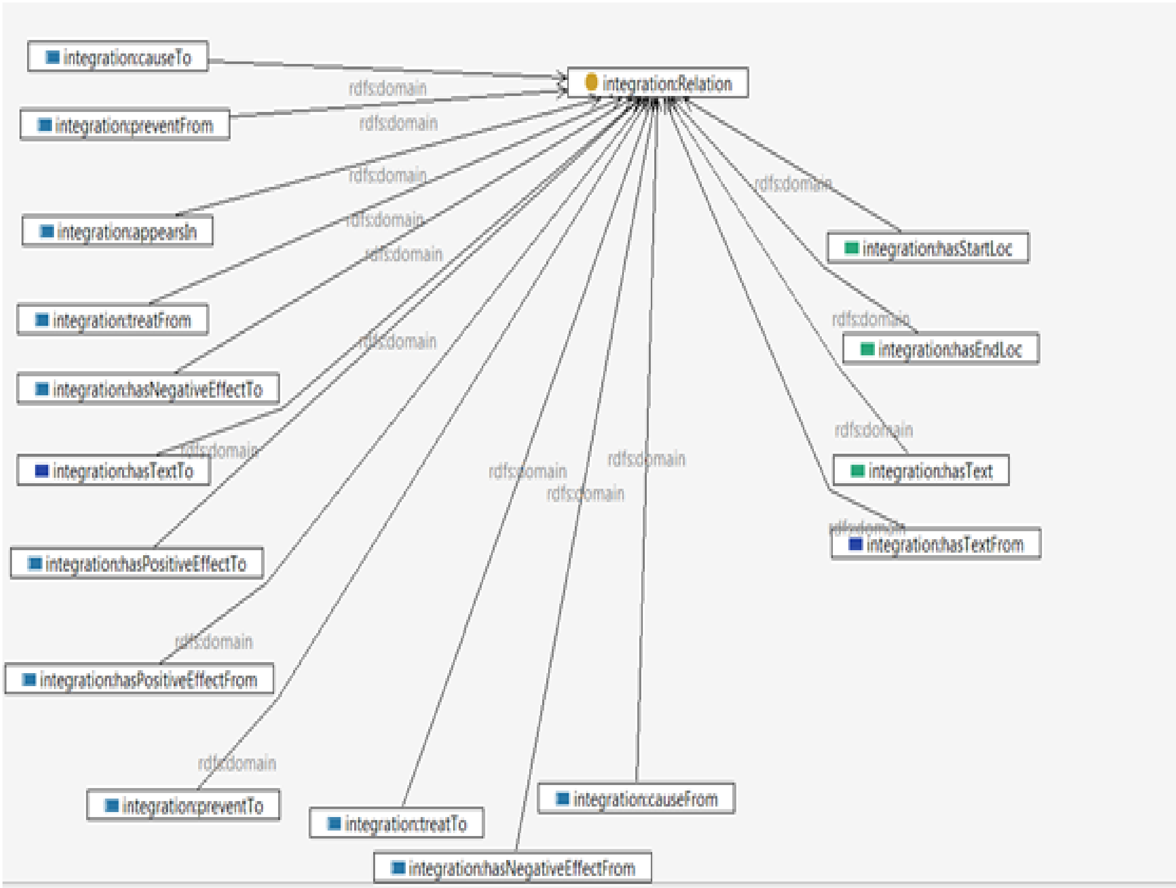
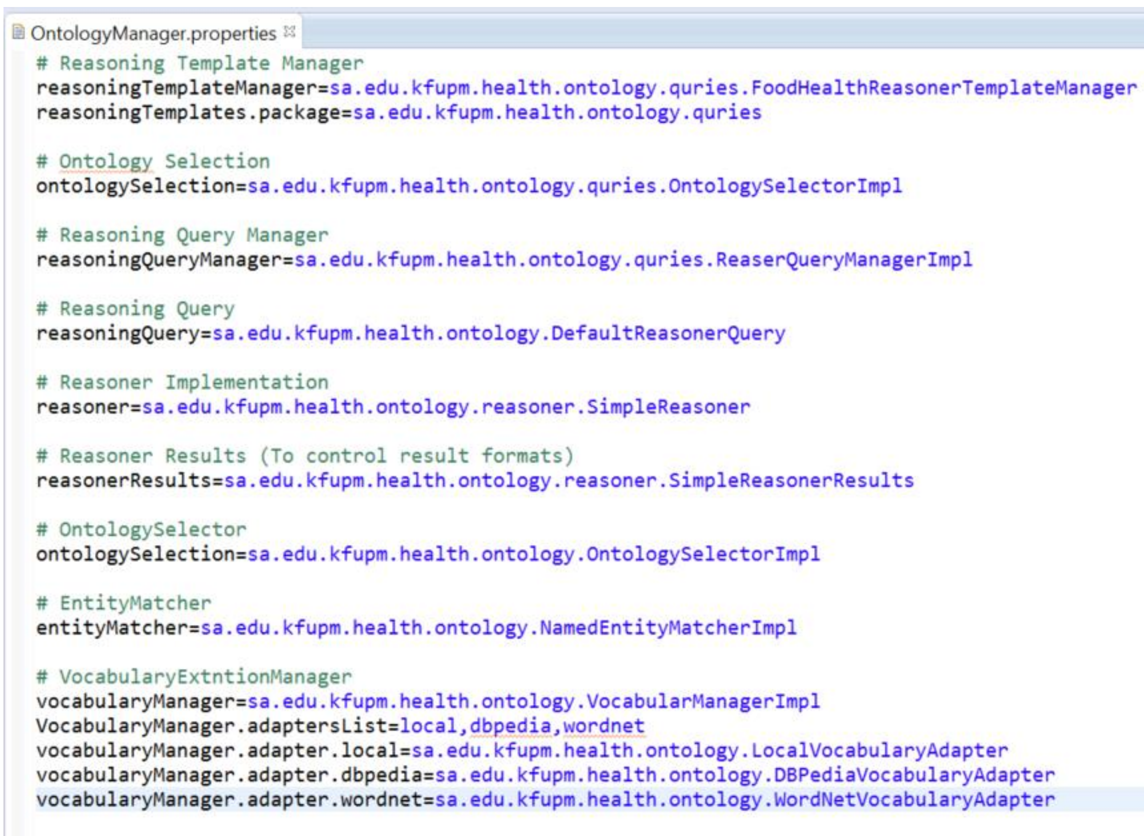


Figure 25 - Integration Ontology

5.2 Ontology Management API: Implementation

Ontology Management is like a pillar to any semantic Web application and this framework is designed to efficiently serve the management tasks which are specific to ontology based knowledge integration from various domains. The framework is designed and exposed as application programmable interface (APIs) for different management tasks. As the framework is designed to be domain independent, these APIs are extendable by the consumer components of OSAPIR for domain specific implementation for configuring the ontologies.

We used the sesame for the storage of the ontologies as it has a very flexible architecture and it's widely adopted and is considered a de-facto standard in the industry for building semantic Web applications. The flexible architecture of the sesame repository allows third party reasoner which is also added benefit of the sesame. OWLIM being an industry ready reasoner product is our choice which is pluggable to sesame. We used the Sesame and OWLIM for the knowledge persistence and reasoning of the annotated knowledge. Figure 26 shows how to configure the Ontology Management framework for domain specific implementation to support OSAPIR in different domains. It shows the sample configuration file used for the case study implementation of the framework.



```
OntologyManager.properties
# Reasoning Template Manager
reasoningTemplateManager=sa.edu.kfupm.health.ontology.quries.FoodHealthReasonerTemplateManager
reasoningTemplates.package=sa.edu.kfupm.health.ontology.quries

# Ontology Selection
ontologySelection=sa.edu.kfupm.health.ontology.quries.OntologySelectorImpl

# Reasoning Query Manager
reasoningQueryManager=sa.edu.kfupm.health.ontology.quries.ReaserQueryManagerImpl

# Reasoning Query
reasoningQuery=sa.edu.kfupm.health.ontology.DefaultReasonerQuery

# Reasoner Implementation
reasoner=sa.edu.kfupm.health.ontology.reasoner.SimpleReasoner

# Reasoner Results (To control result formats)
reasonerResults=sa.edu.kfupm.health.ontology.reasoner.SimpleReasonerResults

# OntologySelector
ontologySelection=sa.edu.kfupm.health.ontology.OntologySelectorImpl

# EntityMatcher
entityMatcher=sa.edu.kfupm.health.ontology.NamedEntityMatcherImpl

# VocabularyExtntionManager
vocabularyManager=sa.edu.kfupm.health.ontology.VocabularManagerImpl
VocabularyManager.adaptersList=local,dbpedia,wordnet
vocabularyManager.adapter.local=sa.edu.kfupm.health.ontology.LocalVocabularyAdapter
vocabularyManager.adapter.dbpedia=sa.edu.kfupm.health.ontology.DBPediaVocabularyAdapter
vocabularyManager.adapter.wordnet=sa.edu.kfupm.health.ontology.WordNetVocabularyAdapter
```

Figure 26 - Framework Configuration File

5.2.1 Ontology Selection

Ontology selector is mainly a component that provides access to core concepts while analyzing the text by other text processing components. During the processing different ontologies being accessed and searched for mapping to the text. It is accessible through the APIs by the other components to access the ontology as flat structured ontology elements. It also provides the capability to search within the ontology and use the synonyms available through vocabulary extensions. The implementation of the class is generic, domain independent and doesn't require rewriting for each domain. The framework doesn't limit you to have this implementation overridden with your implementation and updating the configuration as in Figure 27. Figure 27 shows a class diagram of the provided APIs to be used for the selector and finding concepts. Any new Vocabulary Extensions or addition of new algorithms to support the process can be written and plugged into the system through the configuration file. The framework will automatically load the classes to operate with the new functionalities.

5.2.2 Entity Matching

Entity matching has a vital role in the text processing tasks and is required for entity recognition process to classify the terms to specifics concepts of domains. It requires to access different ontologies to match against all concepts or instances. Ontology Management PIs exposes the public interface to access the Entity Matching component. The implementation of Entity Matching is provided as part of framework and is independent of domains. It works regardless of what domains the Ontology Management framework is configured for. It relies on Vocabulary Extensions of framework to add extra synonyms to improve search. For search, it utilizes the algorithm provided by the OSAPIR components. Vocabulary Extensions and Search Algorithms can be dynamically configured or through framework configuration file Figure 26 as shown in figure. Figure 28 shows the class diagram of Entity Matching.

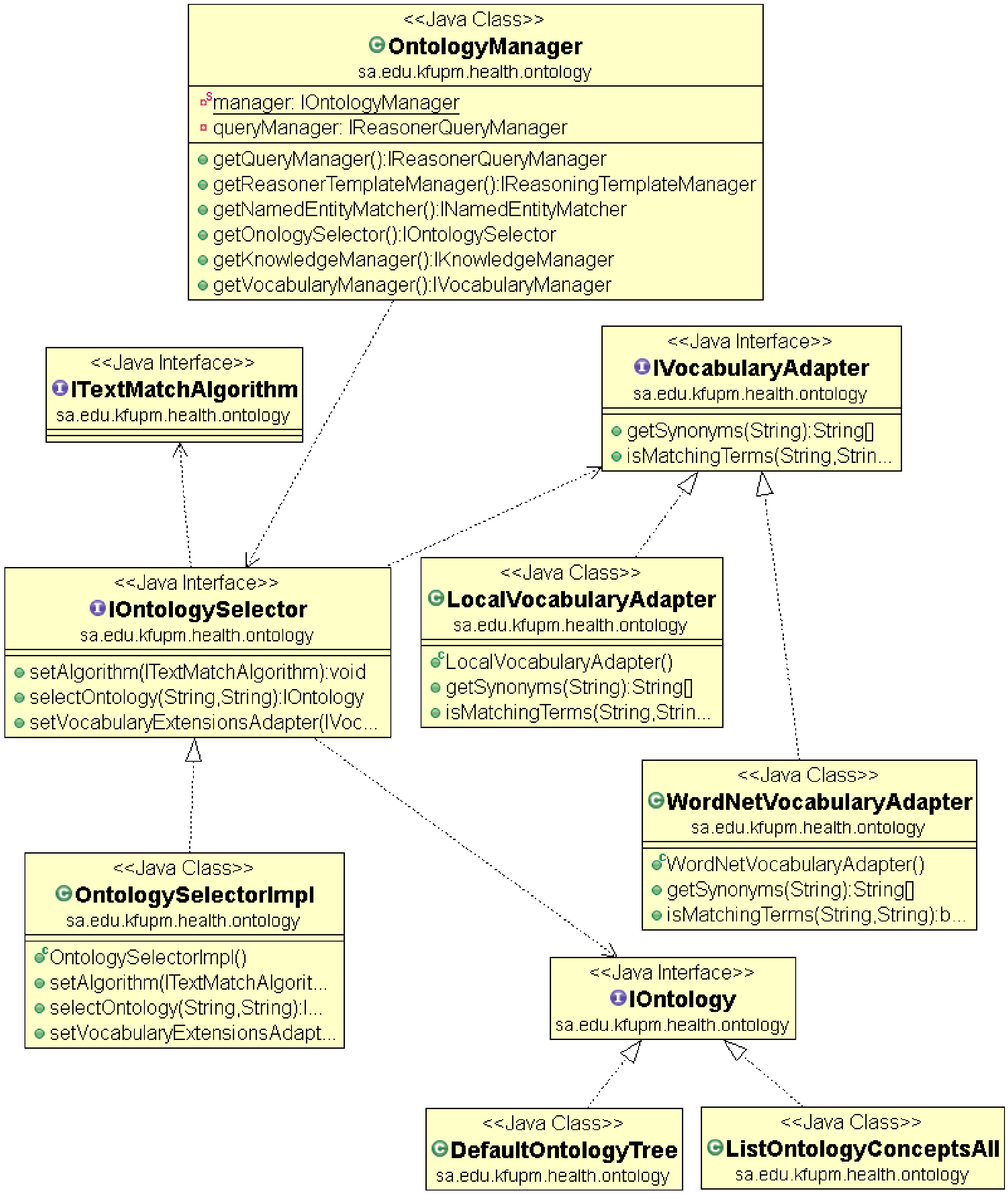


Figure 27 - Ontology Selection Class model

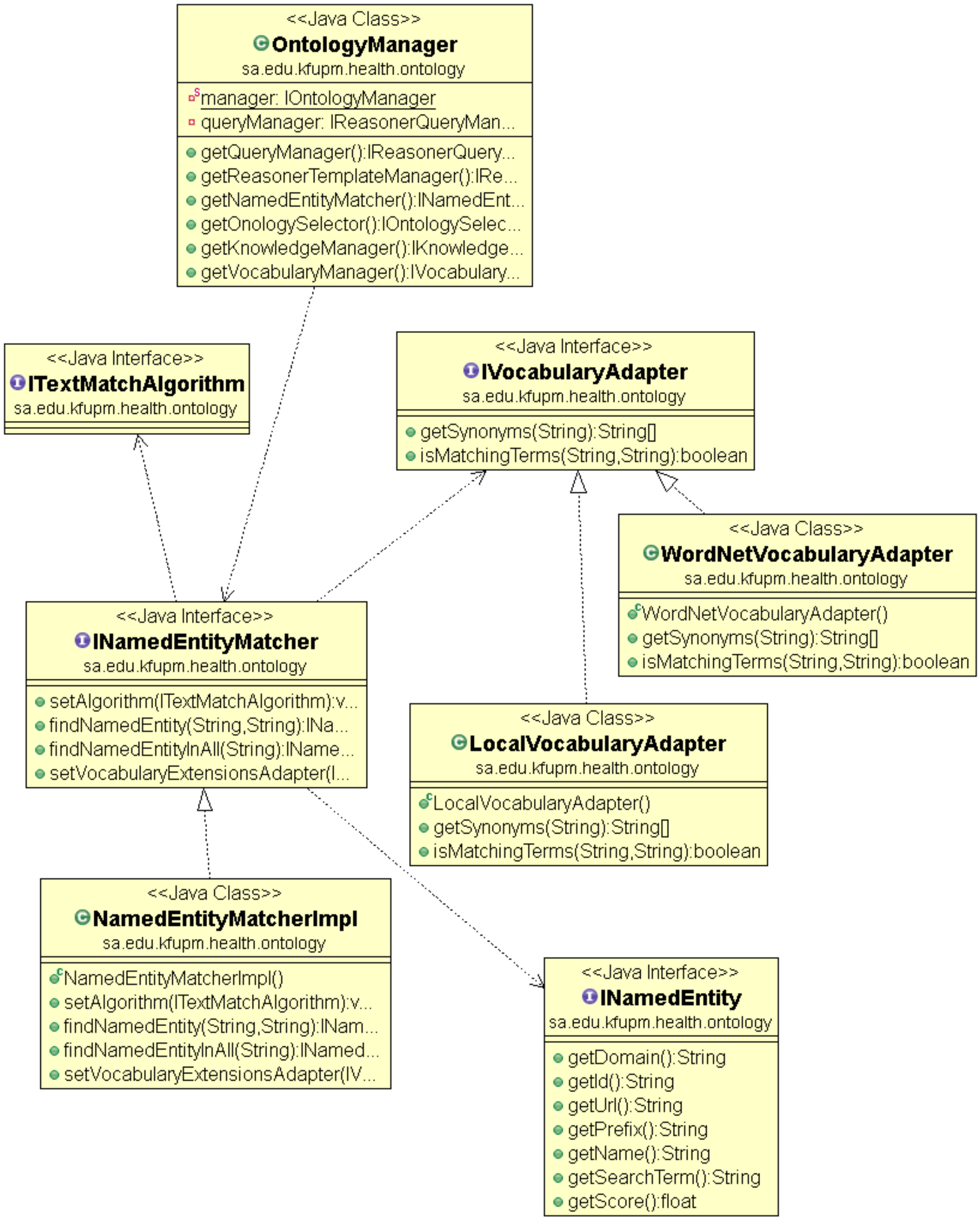


Figure 28 - Named Entity Matcher Class model

5.2.3 Vocabulary Extension

Vocabulary extensions are built based on the adapter pattern of software design techniques which allows it to be integrated with any data source or Web service or even local data source of embedded dictionary. It is up to the other components to implement the required technique by creating the adapter for the data source. Vocabulary extensions are integrated across different components of the ontology management framework. As a default implementation of adapters for vocabulary extensions, two adapters are available one of which is manually maintained vocabularies and the other is WordNet [42] Lexical Adapter. Additional adapters can be written for any lexical data source. The component fetches possible synonyms from the provided sources to enrich the vocabulary at real-time for matching. Figure 29 shows the APIs of the vocabulary extensions. More vocabulary extensions can be provided and can be configured using configurations as shown in Figure 26.

5.2.4 Reasoning Query Manager

The framework is designed using the configurable approach for APIs and implementation to make it as configurable and scalable as possible. Similarly the reasoning APIs are interfaces for the user to access the functionalities of making queries. The implementation classes which are configured in the configuration file of the framework are completely decoupled from APIs used by OSAPIR components and can be changed and reconfigured. As for the current implementation, we used OWLIM-Lite in addition to Open Sesame repository for storing the ontologies and the extracted knowledge base. The reason for selecting the OWLIM is the addition of support OWL 2.0 and reasoning in the sesame repository. Open Sesame provides Java REST APIs for accessing and making queries. In our model, we used these REST APIs to access the repository to manipulate ontologies or knowledge base for reasoning. We used SPARQL queries for the reasoning based on the knowledge base to generate integrated knowledge response for user's queries. Figure 30 shows the APIs for the Reasoner Query Manager.

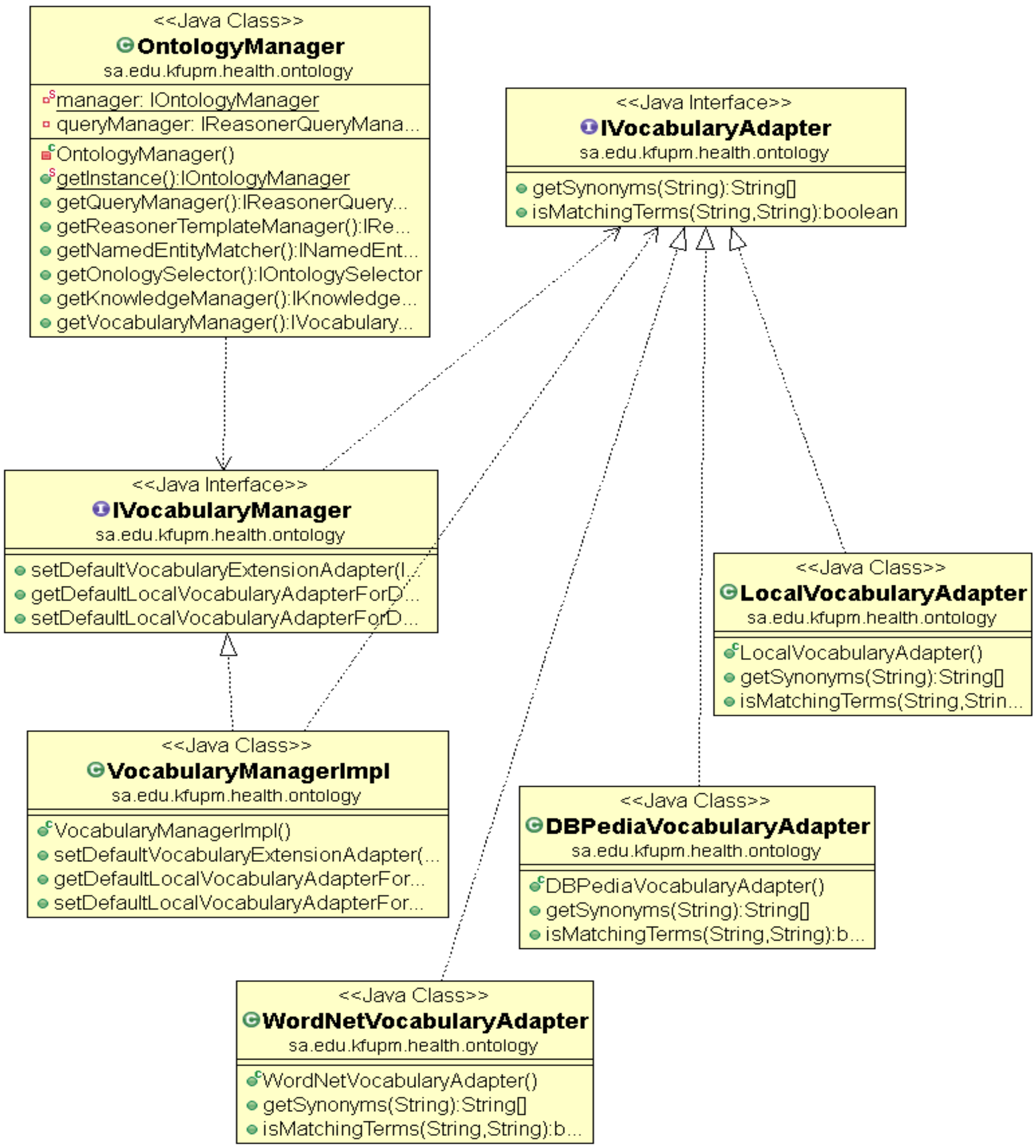


Figure 29 - Vocabulary Extensions

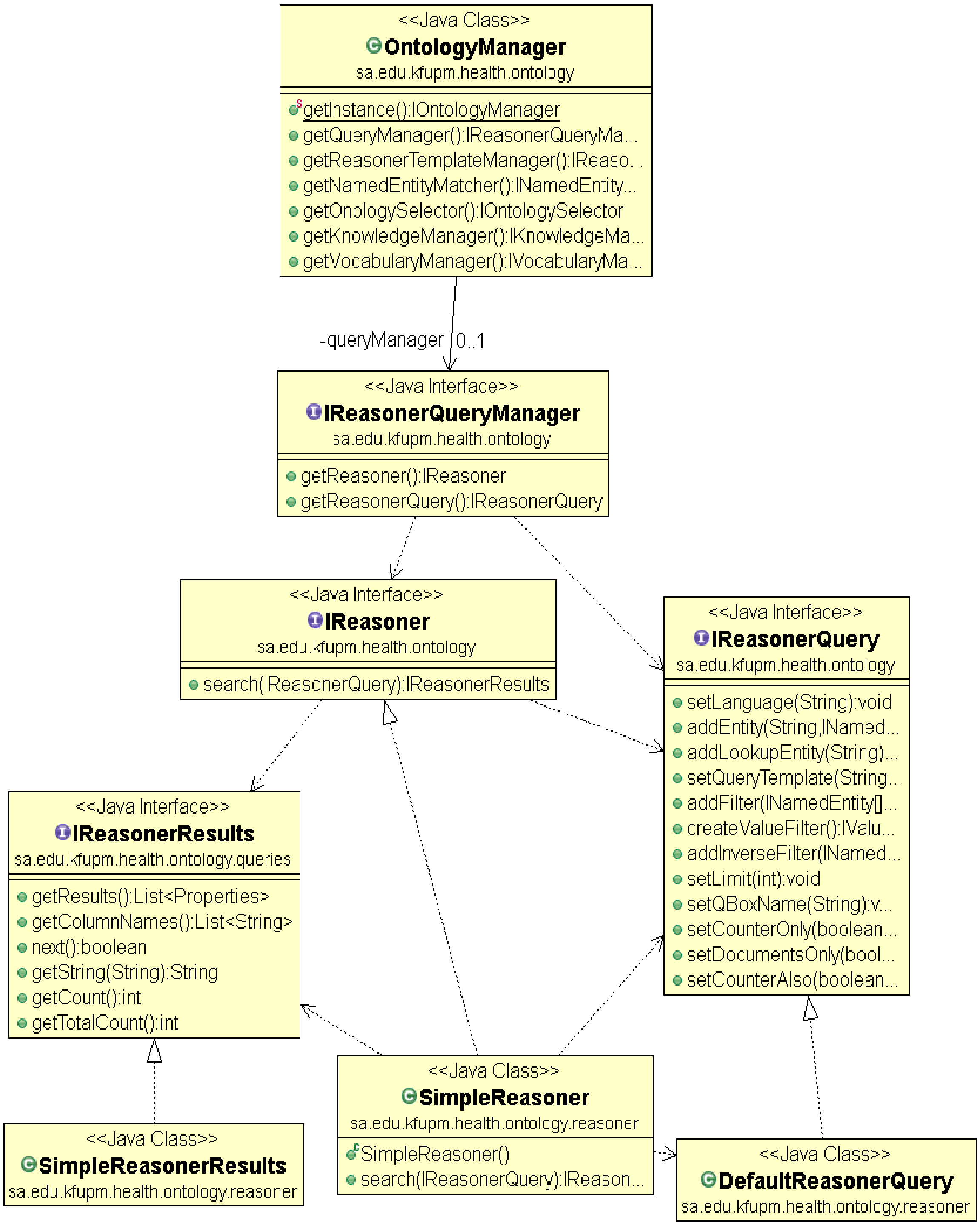


Figure 30 - Reasoner Implementation API

5.2.5 Reasoning Query Template Provider

Reasoning Query Templates are one useful feature of the framework which allows building different possible question templates which are available for direct usage by the OSAPIR components. Figure 31 shows the class hierarchy handling the predefined reasoning templates by extending the class “QBox”. The QBox class internally relies on the Reasoning Query Manager classes to build the query. The new child classes of QBox inherit the features required by template and required customization is added to apply required filters and relations for the query. Each child class represents one reasoning query template to be used by OSAPIR component. For example, “ListTreatsDiseases” is a template which returns the Food or Nutrition which can be useful in treating any disease. Similarly many templates can be created. We implemented many such templates which are utilized by the OSAPIR in the food and health case study implementation.

Secondly, “IReasonerTemplateManager” interface is required to be implemented which hold all the templates list and description for usage by the system. In this case study we created the class “FoodHealthReasoningTemplateManager” class which enables all the templates to available for use. These are provided in the framework configuration file as in Figure 26 which is loaded by the Ontology Manager Implementation automatically.

The highlighted four implementation classes enclosed by the dashed boundary in Figure 31 are the examples of templates for Food and Health reasoning.

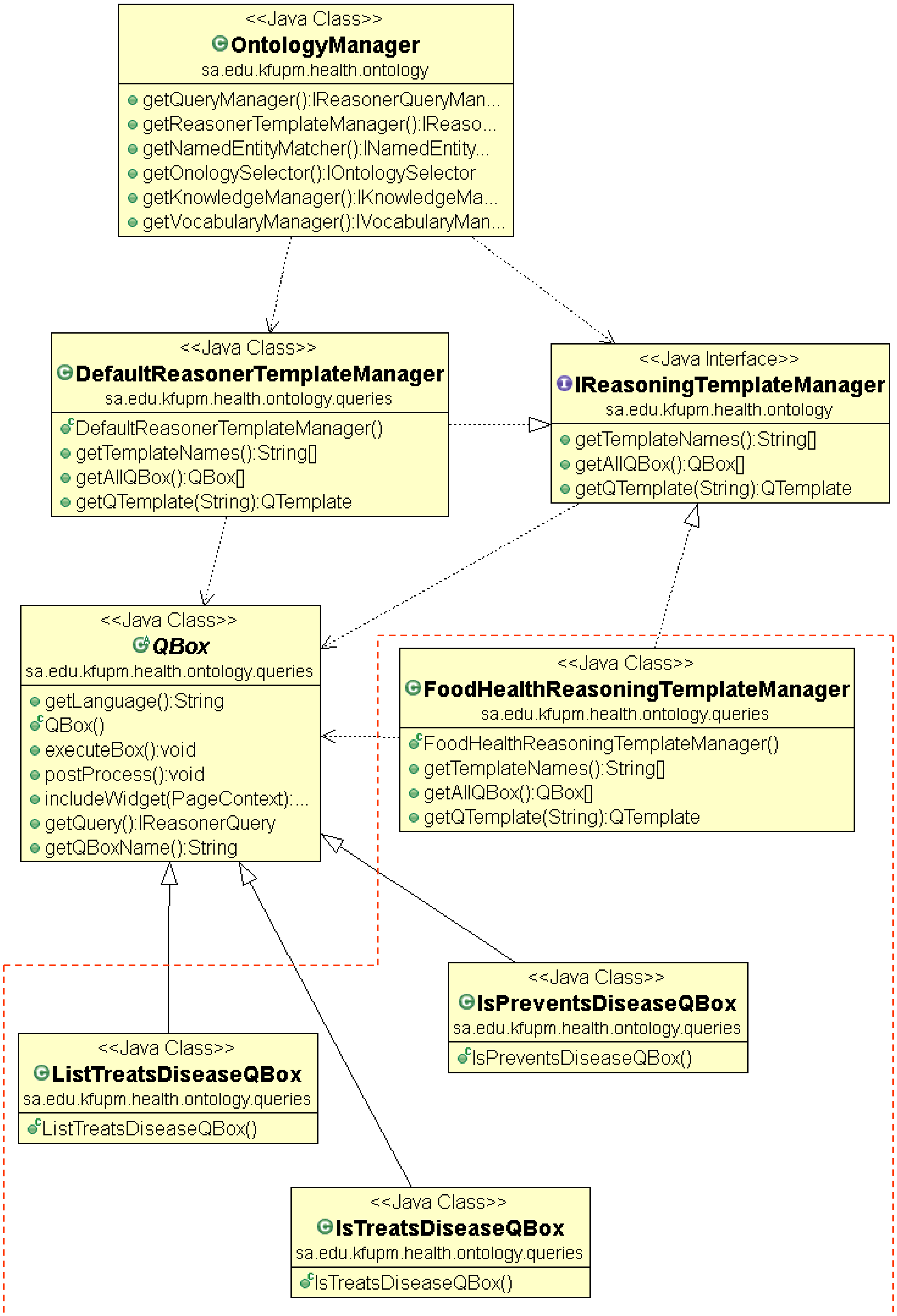


Figure 31 - Reasoning Template Manager - Class Diagram

CHAPTER 6

EXPERIMENTS AND DISCUSSION

We have conducted several experiments to validate the framework performance. We have implemented all the management tasks to support the knowledge integration of the food, health and nutrition domains. This chapter covers the experiments that show how such integration of knowledge is made possible with this framework.

6.1 Ontology Management for the Annotation Process

The Information Annotation component [37] acquires and annotates the data using the ontology selection process provided by Ontology Management Framework to create annotation of entities found from cross-domain ontologies. In this experiment, many documents were annotated and relations between many diseases, food and nutrition have been captured. We will explain how the ontology selection process enables the annotation component to load the ontologies and to perform the string matching on the ontologies. It uses Entity Matching to find different terms for the text processing to annotate the related knowledge of two domains by producing the RDF response based on the integration ontology. Figure 32 shows one relation instance produced by annotation component while utilizing the ontology selection to access the vocabularies of food and disease. Annotation component also makes use of Entity Matching to map the term “blood pressure” to a specific concept of the ontology which is “I_DOID_10762” while the term used in domain ontology is “Hypertension”. This was possible for the Entity Matching by using the vocabulary extensions to provide synonyms for the concepts of specific domain. Similarly food item “banana” is mapped to the “I09040”. These are the IDs of the concepts in the domain knowledge of disease and food ontologies. The relation instance has linked “banana” to “blood pressure” with the relation “treatsTo” in the instance of relation concept of the integration ontology. Such annotated information enables the Reasoning Query Manager to infer the results based on the new captured relations.


```
<integration:Relation
rdf:about="http://www.kfupm.edu.sa/ontology/integration/relation/Relation_13910
70732126">
<integration:appearsIn
rdf:resource="http://www.kfupm.edu.sa/ontology/integration/sentence/Sentence_13
92320506767_5996"/>
<integration:hasText rdf:datatype="http://www.w3.org/2001/XMLSchema#string">can
lower</integration:hasText>
<integration:treatFrom
rdf:resource="http://www.kfupm.edu.sa/ontology/food/foodItem/I09040"/>
<integration:treatTo
rdf:resource="http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_10762"/>
<integration:hasTextFrom
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">bananas</integration:has
TextFrom>
<integration:hasTextTo
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">blood
pressure</integration:hasTextTo>
</integration:Relation>
```


Figure 32 - Annotation RDF - Relation Instance

Figure 33 shows the annotation RDF containing the instances of Document and Sentence for the Relation instance shown in Figure 32. Document instance captures the details such as URL while Sentence instance keeps the actual sentences and start and end position of the text in the document. This information is useful in front-end application if required to highlight the actual source of annotated information. The above experiments demonstrated how the knowledge of two domains is annotated by and produced as RDF which can be reasoned by Reasoner Query Manager.

```
<integration:Document
rdf:about="http://www.kfupm.edu.sa/ontology/integration/document/Document_1392320506767">
<integration:hasTitle xml:lang="en">
        eatthis/8-health-beneits-of-bananas.html_0017F
</integration:hasTitle>
 <integration:hasURL rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
  http://www.healthdiaries.com/eatthis/8-health-beneits-of-bananas.html
</integration:hasURL>
<integration:hasLang
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">en</integration:hasLang>
<integration:containsSentence
rdf:resource="http://www.kfupm.edu.sa/ontology/integration/sentence/Sentence_1392320506767_5996"/>
</integration:Document>
<integration:Sentence
rdf:about="http://www.kfupm.edu.sa/ontology/integration/sentence/Sentence_1392320506767_5996">
 <integration:beginLocation
rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">637</integration:beginLocation>
 <integration:endLocation
rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">832</integration:endLocation>
 <integration:content xml:lang="en">Studies show that the high amounts of potassium in bananas
(over 13% of the RDA) can
   lower one's blood pressure, which in turn lessens the possibility of atherosclerosis, heart
attack and stroke.</integration:content>
</integration:Sentence>
```


Figure 33 - Annotation RDF - Document and Sentence Instances

We performed experiments to evaluate and improve the performance of the Vocabulary Extension services provided by the Ontology Management Components as it plays important role by increasing the search space for processing the documents. During annotation process by the OSAPIR components, we scanned 1000 thousand documents for the information extraction and annotation by use of the Ontology Selection and Entity Matching provided by the Ontology Management Components. It was clearly observed by performing the initial tests without any supplementing vocabularies that the ontology itself doesn't contain enough vocabulary to process the text and additional vocabularies are required. Table 1 shows the experiments results in annotation of documents when using the entity matching with and without the vocabulary extensions.

	Matched (No Vocabulary)	Matched (WordNet)	Matched Local Vocabulary
Food-centric questions	285	543	849
Nutrition-centric questions	78	279	507
Disease-centric questions	45	147	594
Body Part-centric questions	99	210	243
Body Function-centric questions	60	141	318

Table 1 - Entity Matching Comparisons in Annotation Processes

In other experiments, we applied vocabularies extensions adapter for WordNet to see the improvements in the text matching and found better results in matching the terms. In the last experiment, we used vocabulary extension adapter for locally saved manual linguistic support where we provided the exhaustive vocabularies by identifying various concepts and their common synonyms that are provided in the properties file. With the locally provided vocabularies we could improve the quality of the text processing to some more extent.

6.2 Ontology Management for Query Understanding

The semantic query manipulation is very important to respond with relevant answers. In OSAPIR, query manipulation component [38] analyzes the query with ontology matching by using Entity Matching via Ontology Management APIs. In this experiment, we submitted many queries with different commonly used vocabularies for diseases to evaluate how effective is the generated semantic query by the use of vocabulary extensions. Queries like “What food should be avoided with kidney stones?” OR “What food should be avoided in presence of Urinary Stones?” OR “What food should be avoided for diseases like renal calculus?” use three different commonly used names of one disease. The disease ontology has one concept defined which represents this disease. The label used in the ontology for the disease is "nephrolithiasis" which is probably a medical name for the actual disease. Entity Matching by the Ontology Management APIs helps finding the disease terms from the all the three questions in ontology while Vocabulary Extensions brings additional vocabularies of commonly used names of the disease. These vocabulary or synonyms of the disease allows to be mapped to the correct concept. Figure 34 shows example of five different user queries using five different names of a same disease while the final semantically enriched query is common to all. Vocabulary Extensions provided all possible vocabularies for the disease while Entity Matching was effectively able to map to the right concept of the disease ontology.

User Query:

What food should be good in presence of Urinary Stones?
What food should be good with kidney stones?
What food should be good for diseases like renal calculus?
What food should be good for diseases nephrolithiasis?
What food should be good for diseases Kidney calculi?

Semantic Query:

User Language: en
Inquiry Domain: Food
Found Domains: Heath

Named Entity: nephrolithiasis
Ontology Reference: http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_585

Relation Words: Word: good Word Type: Relation en
Matched Relation: GoodFor

Figure 34 - Entity Matching for Query Understanding (1)

Figure 35 shows another example where the user provided commonly used terms which actually refers to the disease. Query manipulation should be able to map "weak bones" to the actual disease concept "Osteoporosis" in the ontology by knowing all the possible vocabularies that are relevant to this disease.

We observed from the above two examples that query understanding is improved by providing a rich vocabulary support in the Entity Matching service provided by the Ontology Management Framework.

User Query:

What food is good for people with weak bones?
What food is god for people with disease Osteoporosis?

Semantic Query:

User Language: en
Inquiry Domain: Food
Found Domains: Heath

Named Entity: Osteoporosis
Ontology Reference:
http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_11476

Figure 35 - Entity Matching for Query Understanding (2)

We performed test with around thousand queries and manually identified the concepts in the queries and compared with Entity Matching done by the query manipulation component. We observed the improvements with the Vocabulary Extensions provided to support Entity Matching in query understanding. Table 2 shows the entity matching results in query understanding with and without vocabulary extensions.

	Matched (No Vocabulary)	Matched (WordNet)	Matched Local Vocabulary	Available Concepts
Food-centric questions	95	181	283	315
Nutrition-centric questions	26	93	169	227
Disease-centric questions	15	49	198	222
Body Part-centric questions	33	70	81	86
Body Function-centric questions	20	47	106	116

Table 2 - Entity Matching in Query Understanding

6.3 Ontology Management for Reasoning

We evaluated the reasoning capabilities of the Reasoner Query Manager APIs by performing different experiments to see how cross-domain annotated knowledge can be reasoned. We also evaluated the multilingual support in the reasoning to see if the knowledge extracted based on the English language can be helpful to provide semantic answer to the user with Arabic query. Initially, we will demonstrate how the reasoning process in general then later few examples of queries and results will be discussed. For instance, if the user submitted the query, “Is banana good for any diseases?” As shown in Figure 38, the produced query contains lookup entity (disease), lookup relation (hasPositiveEffectBy), named entity (banana) with the reference concept ID from the ontology (http://www.kfupm.edu.sa/ontology/food/foodItem/I09040)”.

```
INPUT QUERY

hasPositiveEffectBy:
==========================================
    Bananas (http://www.kfupm.edu.sa/ontology/food/foodItem/I09040)

Lookup Entities:   disease[ANY]
URL: http://localhost:8080/openrdf-sesame/repositories/NSTIP6
```

Figure 36 - Semantic Query


```
 <integration:Relation
rdf:about="http://www.kfupm.edu.sa/ontology/integration/relation/Relation_1351351422098">
   <integration:appearsIn>
     <integration:Sentence
rdf:about="http://www.kfupm.edu.sa/ontology/integration/sentence/Sentence_1349945408614_2534">
       <integration:endLocation rdf:datatype="http://www.w3.org/2001/XMLSchema#integer"
       >3316</integration:endLocation>
       <integration:beginLocation rdf:datatype="http://www.w3.org/2001/XMLSchema#integer"
       >3168</integration:beginLocation>
       <integration:content rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
       >For example, the B6 in bananas acts as an anti-in???????ammatory agent that helps ward off
cardiovascular disease,
  type II diabetes, as well as obesity.</integration:content>
     </integration:Sentence>
   </integration:appearsIn>
   <integration:hasPositiveEffectTo>
     <health:Disease rdf:about="http://www.kfupm.edu.sa/ontology/health/disease/D020"/>
   </integration:hasPositiveEffectTo>
   <integration:hasPositiveEffectFrom>
     <food:FoodItem rdf:about="http://www.kfupm.edu.sa/ontology/food/foodItem/I09040"/>
   </integration:hasPositiveEffectFrom>
   <integration:hasText rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
   >hasPositiveEffect</integration:hasText>
 </integration:Relation>
```


Figure 37 - Experiment Relation instance

Figure 37 shows the annotated instance of the relation concept representing the link between given food and the disease along with the type of relation which should be returned if inquired using the SPARQL query. Figure 38 shows the translated query in SPARQL language which is generated based on the semantic query as we discussed earlier and shows the expected columns.

```
Translated SPARQL:
select * where { ?relation rdf:type integration:Relation .?relation integration:appearsIn
?sentence .?document integration:containsSentence ?sentence .?sentence
integration:content ?document_text . ?document integration:hasURL ?document_url
.?relation integration:hasPositiveEffectTo ?disease .?relation
integration:hasPositiveEffectFrom ?foodItem .?disease rdfs:label ?disease_lbl .
?foodItem rdfs:label ?foodItem_lbl . filter(lang(?disease_lbl) = "en") .
filter(lang(?foodItem_lbl) = "en") . bind(str(?disease_lbl) as ?disease_label) .
bind(str(?foodItem_lbl) as ?foodItem_label)
.filter(strstarts(str(?disease),"http://www.kfupm.edu.sa/ontology/health/disease/"))
.filter(strstarts(str(?foodItem),"http://www.kfupm.edu.sa/ontology/food/foodItem/I09040"))
.filter(strstarts(str(?disease),"http://www.kfupm.edu.sa/ontology/health/disease/D020")) }
LIMIT 25

Generated Column Set:
[relation, sentence, document, document_text, document_url, disease, foodItem,
disease_lbl, foodItem_lbl, disease_label, foodItem_label]
```

Figure 38 - Translated SPARQL

6.3.1 Multilingual Support for Reasoning

We performed some experiments to evaluate the reasoning of knowledge for English and Arabic languages. For example, Figure 39 and Figure 40 show the results of the query, "Is Avacado good for skin?" in English and Arabic which is made possible through integration ontology with multilingual support.

FoodItem_label	RelationPropertyTo	BodyPart_label
"Avocados"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"skin"

Figure 39 - English Result Set

FoodItem_label	RelationPropertyTo	BodyPart_label
"الأفوكادو"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"جلد"

Figure 40 - Arabic Result Set

6.3.2 Reasoning for Different Relation

Here we present different queries and results to demonstrate that the reasoning can be done on different food and health domains. Figure 41 shows the results of the query "What are benefits of Apple?" which enables the Reasoner Query Manager to fetch results based on the positive relation of "Apple" to any disease using SPARQL.

Figure 42 shows the results of the query “Which Food is not good for arthritis?” which searches for negative relations of any food item to the disease “arthritis” and found two food items which cause “arthritis” disease.

FoodItem_label	RelationPropertyTo	Disease_label
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"bone"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"familial adenomatous polyposis"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"immune system"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"reactive arthritis"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/treatTo>	"reactive arthritis"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/treatTo>	"SAPHO syndrome"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"Alzheimer's disease"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"small cell carcinoma"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"cancerophobia"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/treatTo>	"neonatal candidiasis"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"osteoporosis"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"small cell carcinoma"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"gestational diabetes"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"sporadic breast cancer"
"Apples"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"allergic asthma"

Figure 41 - Results of Apple to Disease Relations

FoodItem_label	RelationPropertyTo	Disease_label
"Cauliflower"	<http://www.kfupm.edu.sa/ontology/integration/causeTo>	"reactive arthritis"
"Oil"	<http://www.kfupm.edu.sa/ontology/integration/causeTo>	"reactive arthritis"

Figure 42 - Results of Food not good for arthritis

6.3.3 Nutrition to Disease Relation Reasoning

Figure 43 shows the negative relations of the Zinc to disease which enables us to answer questions like “Is zinc harmful for any disease or health condition?” while Figure 44 shows different positive relation of Zinc to various diseases.

FoodItem_label	RelationPropertyTo	Disease_label
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/causeTo>	"hemophilia B"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/causeTo>	"osteoporosis"

Figure 43 - Results of Zinc's negative relations to Diseases

FoodItem_label	NutrientRelationTo	Disease_label
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"SAPHO syndrome"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/treatTo>	"SAPHO syndrome"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/treatTo>	"oral candidiasis"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"osteoporosis"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"SAPHO syndrome"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"postpartum depression"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"portal hypertension"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/hasPositiveEffectTo>	"osteoporosis"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"gestational diabetes"
"Zinc, Zn"	<http://www.kfupm.edu.sa/ontology/integration/preventTo>	"factor X deficiency"

Figure 44 - Results of Zinc's positive Relations

6.3.4 Nutrition Values of Food Item

Figure 45 shows the results of query “What are the nutrition values of Milk?” based on USDA database.

Nutrient_label	NutrientUnit_label	NutrientValue	Food_label
"Fatty acids, total polyunsaturated"	"Gram per 100 grams"@en	0.750	"Milk, filled,
"20:4 undifferentiated"	"Gram per 100 grams"@en	0.000	"Milk, filled,
"Folate, DFE"	"Microprogram Per 100 grams"@en	5	"Milk, filled,
"Vitamin A, IU"	"Integration Unit per 100 grams"@en	7	"Milk, filled,
"Energy"	"Kcal per 100 grams"@en	63	"Milk, filled,
"Zinc, Zn"	"Milligram Per 100 grams"@en	0.36	"Milk, filled,
"Phosphorus, P"	"Milligram Per 100 grams"@en	97	"Milk, filled,
"Alanine"	"Gram per 100 grams"@en	0.115	"Milk, filled,
"Fatty acids, total saturated"	"Gram per 100 grams"@en	0.768	"Milk, filled,

Figure 45 – Nutrition values of Milk

6.3.5 Deducing More Facts Dynamically

Figure 46 demonstrate the reasoning capabilities built in the framework that allow us to find food items that have negative or positive relations to health conditions based on the annotated relations between nutrition and health conditions. With this approach we are able to deduce more fact about food and to health relations.

Disease_label	NutrientRelationFrom	Nutrient_label	FoodItem_label
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Beef, round, top rou
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Beef, round, top rou
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Turkey breast, pre-b
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Beef, tenderloin, stea
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Turkey patties, bread
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Turkey roast, bonele
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Beef, round, top rou
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Turkey thigh, pre-ba
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Turkey roast, bonele
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Beef, round, top rou
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Fish, herring, Atlant
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Beef, round, top rou
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Fish, herring, Atlant
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Fish, herring, Atlant
"SAPHO syndrome"	<http://www.kfupm.edu.sa/ontology/integration/preventFrom>	"Zinc, Zn"	"Fish, grouper, mixed

Figure 46 - Food to Disease Relation based on Nutrition Relations

6.3.6 Distributed Reasoning

We also performed few experiments to evaluate the framework's performance of handling the distributed ontologies. As the framework is built using Sesame, its flexible architecture provides the repository abstraction layer which hides us from dealing with repositories independently. In the experiment, we tested the distributed management of ontologies by utilizing the APIs available for Sesame and exploited the SPARQL querying in Sesame for reasoning in distributed repositories. We used three repositories for the experiment where we put health related ontologies in Repository (A), food and nutrition relation ontologies in Repository (B) and all annotated knowledge is stored in Repository (C). In this experiment, we stored the ontologies in distributed and remote repositories. Figure 47 shows the setup of the repositories and SPARQL execution process.

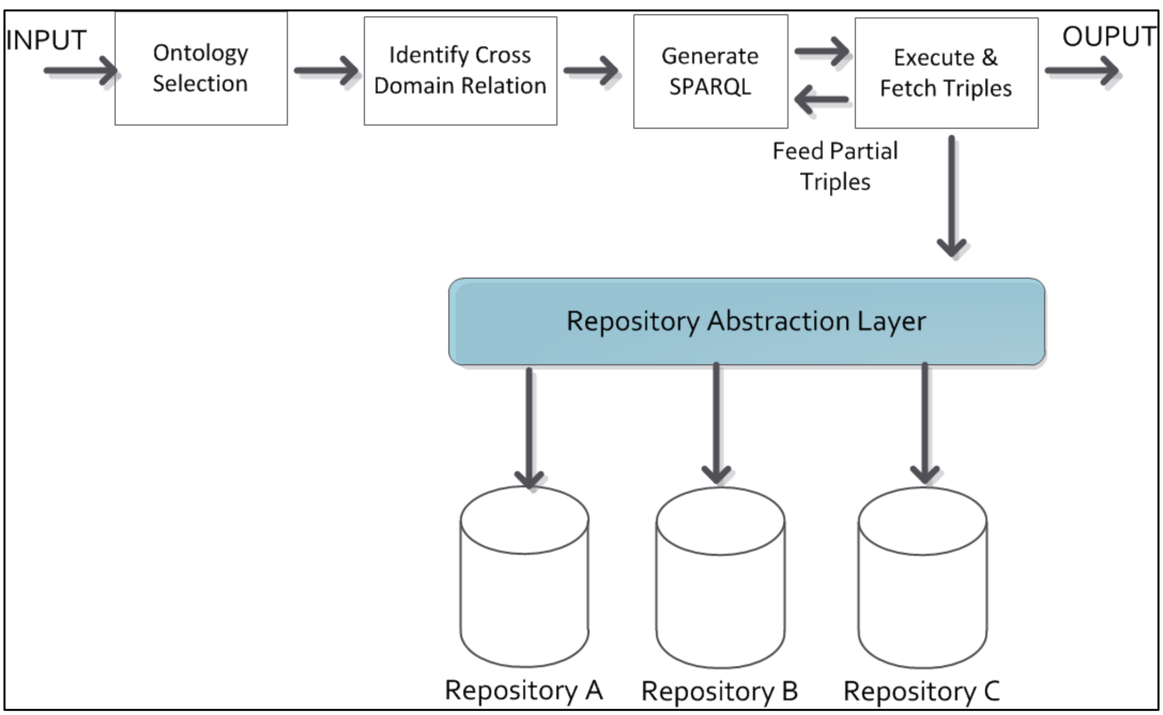


Figure 47 - Experiment of Distributed Ontologies

The queries which are dealing with only food information can be answered based on executing the SPARQL on Repository B which contains all the food and nutrition related ontologies. The queries like “What are the nutrition values of Milk?” as discussed in Section 6.3.4 of this chapter requires executing the SPARQL by Query Manager only on Repository B. In case of any question involving the relation of the ontologies from food to health, the SPARQL will be first executed on Repository C which holds all the annotation linking the food to health condition as shown in Figure 48. Then later, additional SPARQLs are generated to Repository A and Repository B to fetch the triples belonging to respective health and food domains. The union of the results provides the complete results for the users as shown in Figure 49 and Figure 50 respectively.

FoodItem	RelationPropertyTo	Disease
<http://www.kfupm.edu.sa/ontology/food/foodItem/I02048>	<http://www.kfupm.edu.sa/ontology/integration/treatTo>	<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_6196>

Figure 48 - Food and Health Annotation from Repository C

Subject	Predicate	Object
<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_6196>	rdf:type	<http://www.kfupm.edu.sa/ontology/health/Disease>
<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_6196>	rdf:type	<http://www.kfupm.edu.sa/ontology/health/disease/DOID_6196>
<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_6196>	rdfs:label	"reactive arthritis"@en

Figure 49 - Disease related triples from Repository A

In our experiments, we applied distributed querying approach based on the distribution of the ontologies among these repositories. The queries are refined and executed based on the inquired information. If the query involves food only information, then SPARQL is generated to Repository B which holds complete food and nutrition information which is independent of health (stored in Repository A) and any annotations (stored in Repository B). Similarly any health only information can be inquired directly from Repository A directly.

Subject	Predicate	Object
<http://www.kfupm.edu.sa/ontology/food/foodItem/I02048>	rdf:type	food:FoodItem
<http://www.kfupm.edu.sa/ontology/food/foodItem/I02048>	rdfs:label	"خل"@ar
<http://www.kfupm.edu.sa/ontology/food/foodItem/I02048>	rdfs:label	"Vinegar"@en

Figure 50 - Food related triples from Repository B

In case of the questions where the user inquires about the relation among the domains, the SPARQLs initially executed to Repository C to fetch the results and then further SPARQLs are executed to Repository A and Repository B to fetch triples to fetch the information about the annotated food and health terms. This experiment shows that the proposed framework can manage well the ontologies that are independent from each other and stored remotely in the distributed environment.

CHAPTER 7

CONCLUSION & FUTURE WORK

Ontology based knowledge base is being published by different publishers of specific domains and there is no coordination among them in terms of relationships between their domain knowledge. The diversity of knowledge in different fields such as health care, where medical knowledge such as disease and health conditions are published independently while food and nutrition on the other hand, are scenarios which clearly present the need of semantic reconciliation of the knowledge. In this thesis, we investigated the work in the semantic integration between networks of heterogeneous ontologies and present a framework for integrating of cross-domain multilingual ontologies. We proposed an integration ontology model to integrate different domains based on the relation among them. We also developed management framework for the efficient use of those distributed and heterogeneous ontologies to provide services for extraction, annotation and reasoning to make the knowledge interoperable and multilingual. In the case study implementation of food and health, it's evident that the approach of integrating knowledge from these distributed domains is novel and can be very useful for users looking for interoperable knowledge. There are certainly areas of improvements in the framework. We successfully implemented and demonstrated the Arabic knowledge annotation and query answering support by the framework. In the real-world, the ontology structure for each language may not be the same. The culture comes along with the language to dictate how the structure of knowledge could be designed in the ontology. Addition of culture information along with language information, leave more differences in the ontologies structure and not just the translation. Such ontologies which differ in language and concept hierarchy structure as well cannot be directly mapped as in our case. In case of multilingual ontologies where domain ontologies already exist and encapsulates cultural changes in the domain along with language would not be similar in structure and aligning of such ontologies is required to work with this framework. If the framework implementation supports agnostic ontology approach for multilingual ontologies, then this limitation can be handled as well. This could be a future work in order to improve the framework further to handle cultural changes along with languages in the domain ontologies.

References

[1] T. Berners-Lee, “The semantic web,” *Scientific ...*, 2001. [Online]. Available: http://isel2918929391.googlecode.com/svn-history/r347/trunk/RPC/Slides/p01_theSemanticWeb.pdf. [Accessed: 21-Apr-2014].

[2] “Resource Description Framework (RDF).” [Online]. Available: http://www.w3.org/RDF/. [Accessed: 21-Apr-2014].

[3] “Web Ontology Language (OWL).” [Online]. Available: http://www.w3.org/2001/sw/wiki/OWL. [Accessed: 21-Apr-2014].

[4] “The Protégé Ontology Editor and Knowledge Acquisition System.” [Online]. Available: http://protege.stanford.edu. [Accessed: 20-Jan-2013].

[5] D. Lonsdale, D. Embley, and S. Liddle, “Ontologies for Multilingual Extraction.,” *MSW*, 2010.

[6] “Some early ideas for HTML.” [Online]. Available: http://www.w3.org/MarkUp/historical. [Accessed: 03-Dec-2013].

[7] “HTML, version 2.” [Online]. Available: http://www.rfc-editor.org/rfc/rfc1866.txt. [Accessed: 15-Dec-2013].

[8] “HTML, version 3.” [Online]. Available: http://www.w3.org/TR/REC-html32. [Accessed: 15-Dec-2013].

[9] “HTML, version 4.” [Online]. Available: http://www.w3.org/TR/html401/. [Accessed: 15-Dec-2013].

[10] L. O. Berners-Lee T, Hendler J, *The Semantic Web*. 2001.

[11] “eXtensible Markup Language.” [Online]. Available: http://www.w3.org/TR/REC-xml/. [Accessed: 15-Dec-2013].

[12] “Resource Description Framwork Scheme.” [Online]. Available: http://www.w3.org/TR/rdf-schema/. [Accessed: 21-Apr-2014].

[13] “OntoStudio.” [Online]. Available: http://semanticweb.org/wiki/OntoStudio. [Accessed: 12-Feb-2013].

[14] "Jena Framework." [Online]. Available: https://jena.apache.org/. [Accessed: 16-May-2013].

[15] "PowerAqua - multi-ontology question Answering System." [Online]. Available: http://technologies.kmi.open.ac.uk/poweraqua/. [Accessed: 16-Jun-2013].

[16] "TopBraid Composer." [Online]. Available: http://www.topquadrant.com/products/TB_Composer.html. [Accessed: 20-Jan-2013].

[17] "BigData." [Online]. Available: http://semanticweb.org/wiki/Bigdata. [Accessed: 15-Jan-2014].

[18] "Sesame - OpenRDF." [Online]. Available: http://www.openrdf.org/. [Accessed: 15-Jan-2013].

[19] "OWLIM." [Online]. Available: http://www.ontotext.com/owlim. [Accessed: 15-Jan-2013].

[20] "Java - Programming Langauge." [Online]. Available: http://www.java.com/en/. [Accessed: 10-Jan-2013].

[21] B. Bishop, A. Kiryakov, D. Ognyanoff, and I. Peikov, "OWLIM: A family of scalable semantic repositories," *Semant. Web*, 2011.

[22] H. Kong, M. Hwang, and P. Kim, "A New Methodology for Merging the Heterogeneous Domain Ontologies Based on the WordNet," *Int. Conf. Next Gener. Web Serv. Pract.*, pp. 235–240, 2005.

[23] S. Taduri, G. Lau, and K. Law, "An Ontology to Integrate Multiple Information Domains in the Patent System," *2011 IEEE Int. …*, 2011.

[24] T. Shah, F. Rabhi, and P. K. Ray, "OSHCO: A Cross-domain Ontology for semantic interoperability across medical and oral health domains," *2013 IEEE 15th Int. Conf. e-Health Networking, Appl. Serv. (Healthcom 2013)*, no. Healthcom, pp. 460–464, Oct. 2013.

[25] G. Vadivu and S. W. Hopper, "Semantic Linking and Querying of Natural Food, Chemicals and Diseases," *Int. J. Comput. Appl.*, vol. 11, no. 4, pp. 35–38, Dec. 2010.

[26] Z. Cui and P. O'Brien, "Domain ontology management environment," *Syst. Sci. 2000. Proc. 33rd Annu. Hawaii Int. Conf. on. IEEE*, vol. 00, no. c, pp. 1–9, 2000.

[27] A. Maedche and R. Volz, “The Ontology Extraction & Maintenance Framework Text-To-Onto,” *Proc. ICDM’01 Work. Integr. Data Min. Knowl. Manag.*, pp. 1–12, 2001.

[28] A. Aldea, R. Bañares-Alcántara, and J. Bocio, “An Ontology-Based Knowledge Management Platform.,” *IIWeb*, 2003.

[29] E. Montiel-Ponsoda and G. A. De Cea, “Modelling multilinguality in ontologies,” *Coling 2008 Companion Vol. – Posters Demonstr. Manchester, UK*, no. August, pp. 67–70, 2008.

[30] “Semantic Diet Intelligent dieting on the intelligent web.” [Online]. Available: http://www.semanticdiet.com/. [Accessed: 01-Jan-2013].

[31] “United States Department of Agriculture.” [Online]. Available: http://www.usda.gov. [Accessed: 01-Jan-2013].

[32] “FAO. AGROVOC.” [Online]. Available: http://www.fao.org/agrovoc/. [Accessed: 10-Jan-2014].

[33] C. Snae and M. Bruckner, “FOODS: a food-oriented ontology-driven system,” *Digit. Ecosyst. Technol. 2008. DEST 2008. 2nd IEEE Int. Conf. on. IEEE*, pp. 168–176, 2008.

[34] “International Classification of Diseases (ICD).” [Online]. Available: http://www.who.int/classifications/icd/en/. [Accessed: 20-Jan-2013].

[35] “World Health Organization.” [Online]. Available: http://www.who.int/en/. [Accessed: 05-Jan-2013].

[36] “Human Disease Ontology.” [Online]. Available: https://bioportal.bioontology.org/ontologies/DOID. [Accessed: 05-Jan-2014].

[37] Saeed Yousef Al-Bukhitan, “Multilingual Framework for Ontology-Based Semantic Annotation of Health & Nutrition Websites,” King Fahd University of Petroleum & Minerals, 2014.

[38] A. Al-Nazer, “Agent-based Framework for Semantic Query-Manipulation and Personalized Retrieval of Health and Nutrition Information,” King Fahd University of Petroleum & Minerals, 2014.

[39] N. F. Noy, M. A. Musen, S. M. Informatics, and C. Drive, “Ontology Versioning as an Element of an Ontology-Management Framework,” *Intell. Syst. IEEE 19.4*, pp. 1–17, 2004.

[40] G. Solomou, “Semantics-aware querying of web-distributed RDF (S) repositories,” *Semant. Interoperability Eur. Digit. Libr.*, 2008.

[41] P. Mendes, M. Jakob, and C. Bizer, “DBpedia: A Multilingual Cross-domain Knowledge Base.,” *LREC*, pp. 1813–1817, 2012.

[42] “WordNet - A lexical database for English,” *Princeoton Unversity*. [Online]. Available: http://wordnet.princeton.edu/. [Accessed: 01-Jan-2013].

[43] M. Espinoza, “Multilingual and localization support for ontologies,” *Semant. Web Res. Appl. Springer Berlin Heidelb.*, 2009.

Vitae

Name : Ali Mazhar Iqbal

Nationality : Pakistani

Date of Birth : 11/7/1979

Email : mazharai@gmail.com

Address : P.O. Box 18117, Dhahran 31311

Academic Background:

Bachelor of Science in Computer Engineering (2002)

Experience:

Working in Saudi Aramco for above 10 years as IT System Analyst.

Published**:**

The paper "**Management Framework of Heterogeneous Ontologies for Food, Health and Nutrition Information System**" has been published in Journal of Next Generation Information Technology (JNIT) in volume 4, Number 7 on September 2013.