ONTOLOGY DEVELOPMENT IN FERTILIZER SUB-DOMAIN FOR ENHANCING KNOWLEDGE REPRESENTATION IN THE FIELD OF AGRICULTURE

Dissertation submitted to the
Jawaharlal Nehru University, New Delhi
in partial fulfilment of the requirements for the award of the degree of

MASTER OF TECHNOLOGY
In
COMPUTER SCIENCE AND TECHNOLOGY

By
DEENA HIJAM

UNDER THE SUPERVISION OF
Dr. ADITI SHARAN

SCHOOL OF COMPUTER AND SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110067, INDIA
JULY 2015

जवाहरलाल नेहरू विश्वविद्यालय

SCHOOL OF COMPUTER & SYSTEMS SCIENCES

JAWAHARLAL NEHRU UNIVERSITY

NEW DELHI-110067, INDIA

DECLARATION

I hereby declare that the dissertation entitled **"Ontology Development in Fertilizer Sub-domain for Enhancing Knowledge Representation in the Field of Agriculture"**, submitted by me to the School of Computer and Systems Sciences, **Jawaharlal Nehru University**, New Delhi, in partial fulfilment of the requirements for the award of the degree of **Master of Technology** in **Computer Science and Technology**, is a bona fide work carried out by me under the supervision of **Dr. Aditi Sharan**.

The matter embodied in this dissertation has not been submitted to any other university or institution for the award of any other degree or diploma.

Deena Hijam 21/7/15

Deena Hijam

M.TECH-CSE

SC&SS, JNU,

New Delhi-110067

i

जवाहरलाल नेहरू विश्वविद्यालय

SCHOOL OF COMPUTER & SYSTEMS SCIENCES

JAWAHARLAL NEHRU UNIVERSITY

NEW DELHI-110067, INDIA

CERTIFICATE

This is to certify that the dissertation entitled **"Ontology Development in Fertilizer Sub-domain for Enhancing Knowledge Representation in the Field of Agriculture"**, submitted by Deena Hijam to the School of Computer and Systems Sciences, **Jawaharlal Nehru University**, New Delhi, in partial fulfilment of the requirements for the award of the degree of **Master of Technology** in **Computer Science and Technology**, is a bona fide work carried out by her under my supervision.

The matter embodied in this dissertation has not been submitted to any other university or institution for the award of any other degree or diploma.

Dr. Aditi Sharan
(Supervisor)
SC&SS, JNU, New Delhi

Dr. C.P. Katti
(Dean)
SC&SS, JNU, New Delhi

Dean
School of Computer & Systems Sciences
Jawaharlal Nehru University
New Delhi-110067

Acknowledgement

I would like to gratefully acknowledge the enthusiastic supervision of Dr. Aditi Sharan during this work. This work wouldn`t have been possible without her constant support, valuable suggestions and comments during my whole tenure of this dissertation work. I feel privileged to work under her for my master`s dissertation. I would take this opportunity to thank her once again for her esteemed support and I, from the bottom of my heart would like to wish her the best in all her future endeavours.

I would also like to thank Dr. Rajni Jain (Principal Scientist, NCAP), for helping in identifying the research gap in the field of ontology development in agriculture domain and motivating to select fertilizer as a sub-domain for developing ontology. Thanks are also due for Dr. Pravithra S (Scientist, NCAP) for helping in selecting the authentic sources and references for my work. I would also like to convey my gratitude to Dr. Sudeep Marwaha (IASRI) for sharing his work of soil ontology and information related to AGRIdaksh. I would also like to thank my senior Mrs. Nidhi Malik for always making the work interesting and for her suggestions and support till the last day of this work.

I wish to thank my friends Miss Tarundeep Kaur Saini and Mr. Ashish Kumar for their unconditional support, and my seniors Mr. Jagendra Singh, Mr. Mayank Saini, Mr.Rakesh Kumar, Mr. Chandra Shekhar Yadav, Miss Payal Biswas and Mr. Sharad for creating a home like environment in our lab to keep the stress away.

Finally, I would like to thank the whole faculty members of our department for clarifying my doubts throughout this work and last but not least, the JNU administration for creating such a secular and healthy environment amongst the students.

Dedicated to my family and friends

Abstract

The term ontology has its origin in Philosophy. It is defined as the *"study of existence"*. The term was popularized in the field of Computer Science by Gruber when he gave its definition as *"formal specification of a conceptualization"* (Gruber 1993).The main aim of ontology is to structure and provide knowledge about specific domains that are understandable by both the computers and developers. Ontology acts as a representative for knowledge in a particular domain.

Ontology development has been a popular research issue in many domains such as medicine, electronics, agriculture, etc. The domain of agriculture is a very vast domain and consists of a lot of sub-domains. A lot has been done in the ontology development in agriculture domain. However, if we consider the sub-domain of fertilizer, we see that it has always been studied in relation to crop or soil. We have not come across any ontology which is generic in nature with respect to fertilizer sub-domain. Finding this research gap, we have developed an ontology for fertilizer sub-domain on its own, which may further be related to any other entity such as soil or crop. The aim of this thesis entitled *"Ontology Development in Fertilizer Sub-domain for Enhancing Knowledge Representation in the Field of Agriculture"*, is to show how an ontology can be designed and developed to help manage and represent knowledge about fertilizers. This ontology is created from scratch following standard ontological principles.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
OWL	Web Ontology Language
SPARQL	Simple Protocol and RDF Query Language
DL	Description Logic
OE	Ontology Engineering
KR	Knowledge Representation and Reasoning
TOVE	TOronto Virtual Enterprise

Chapter 1

Introduction

The use of ontology to structure information has been there for a long time in the fields of knowledge modelling and Artificial Intelligence. And in the recent times domain specific ontology development has been gaining its importance among the computer scientists as well as domain experts. Ontology development has been a popular research issue in many domains such as medicine, agriculture, human-anatomy, electronics, etc. In the agricultural domain, most of the work done on ontology development has been crop specific (Jaiswal et al. 2005; Takeya et al. 2003; Vincent et al. 2003; Thunkijjanukij et al. 2009) or context specific (Walisadeera et al. 2013; Walisadeera et al. 2014). The sub-domain of fertilizer of the agricultural domain has been poorly explored in terms of ontology development. Finding this research gap, an ontology in fertilizer sub-domain is developed, taking into account the various issues that are faced while constructing an ontology and the enormous amount of data that are available which are not easy to structure and present as ontology.

1.1 Introduction to ontology

The term ontology originated from Philosophy. There have been a number of definitions given for ontology in the past years. Few are listed below.

1.1.1 Definition

Ontology: "*The branch of philosophy which deals with the nature and organization of reality*".

Tom Gruber in 1993, among others, made the term popular in relation to computer science and artificial intelligence. He gave the definition of ontology as a *“formal specification of a conceptualization"*.

"An ontology is a formal conceptualization of a domain that is shared and reused across domains, tasks and group of people" (A. Gomez Perez et al. 1999).

An ontology can be considered as a term which is used to refer to the shared understanding of some domain of interest which may be used as a framework to solve various problems that come up due to the lack of this shared understanding in the domain (Uschold & Gruninger 1996).

1.1.2 Types of ontology

Ontologies can be classified based on the extent of how the conceptualization behind them are generalized, intended purpose, and their coverage (Davies et al. 2006):

1. Application and task ontologies are designed in such a way that they are suitable for certain ranges of application and tasks.
2. Domain ontologies are ontologies representing a conceptualization of a specific domain, such as medicine, electronics, road construction, etc.
3. Upper-level ontologies are the ones that represent the world in the form of a general mode. These ontologies are used for numerous variety of domains, tasks and application areas.

1.1.3 Why develop ontology?

Noy and McGuinness (2001) pointed out some of the reasons why one wants to develop an ontology as follows:

- To share common understanding of the structure of information among people or software agents
- To enable reuse of domain knowledge
- To make domain assumptions explicit
- To separate domain knowledge from the operational knowledge
- To analyze domain knowledge

The reason why ontologies are so popular is due to the fact they provide a shared and common understanding of some domain and that can be communicated across people and computers (Noy & McGuinness 2001). The main motivation behind the use of ontologies is that they allow for sharing and reuse of knowledge in computational form.

1.2 Ontology and Knowledge Representation

Ontologies are very important for representing knowledge. Given a domain, its ontology forms the heart of any system of knowledge representation for that domain. In order to represent knowledge in the form of a vocabulary also, the ontologies, i.e. the conceptualization that underlie knowledge is needed. They may differ in the way they are represented and the way in which the relationships between concepts are constructed, but they all serve as an explicit specification of a conceptualization. For example, if the users of a given information retrieval system can agree upon the conceptualization in the ontologies used, then the retrieval process can benefit from using these in the evaluation and articulation of requests.

Knowledge representation and reasoning (KR) is the field of artificial intelligence (AI) devoted to representing information about the world in a form that a computer system can utilize to solve complex tasks such as diagnosing a medical condition or having a dialog in a natural language. Knowledge representation goes hand in hand with automated reasoning because one of the main purposes of explicitly representing knowledge is to be able to reason about that knowledge, to make inferences, assert new knowledge, etc. ("Knowledge representation and reasoning," 2015).

Ontologies have become the best choice as the medium of knowledge representation in recent years for a range of computer science applications including the Semantic Web and bioinformatics. Processing textual information and retrieving information intelligently in response to user's queries has emerged as one of the great challenges in information retrieval. The information available over the internet are huge and unstructured in nature making the accessibility of relevant information a difficult task for the users. Most of the information retrieval techniques are based on keywords. These techniques make use of keywords for the information retrieval purpose and so they do not take into account the semantic relationships between the keywords, nor do they consider the meaning of word and phrases. It becomes difficult for ordinary users to use information provided by the keyword based searching techniques. Users often have problems in expressing their information needs and translating them into queries. Information retrieval systems do not actually retrieve information but rather documents from which information can be obtained if they are read and understood.

In the words of Gruber:

"A body of formally represented knowledge is based on a conceptualization: the objects, concepts, and other entities that are presumed to exist in some area of interest and the relationships that hold them. A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose. Every knowledge base, knowledge-based system, or knowledge-level agent is committed to some conceptualization, explicitly or implicitly." (Gruber 1993)

Concept can be anything that exists; anything that we can think of or anything that is a thought or a notion. According to Ogden and Richards (1923), the three components for communication are concept, symbol, and thing. It is shown by the "Meaning Triangle" below:

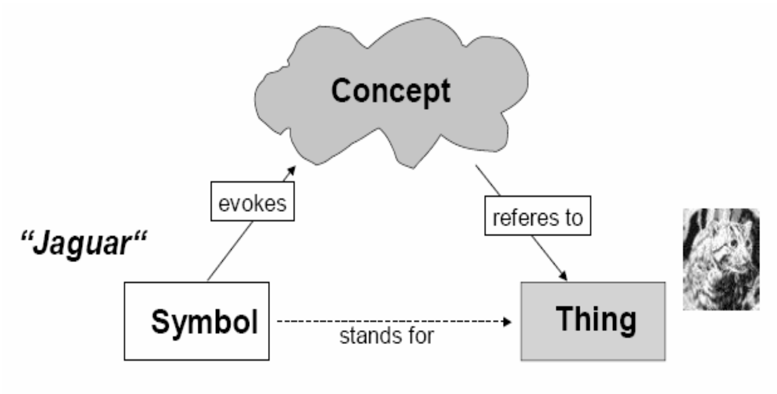


Figure 1.1: Meaning Triangle of Communication
Source: Odgen and Richards (1923)

Ontology is a way of knowledge representation wherein it acts as a medium for human expressions as well as a medium for expressing what we feel to a form that the machines can understand.

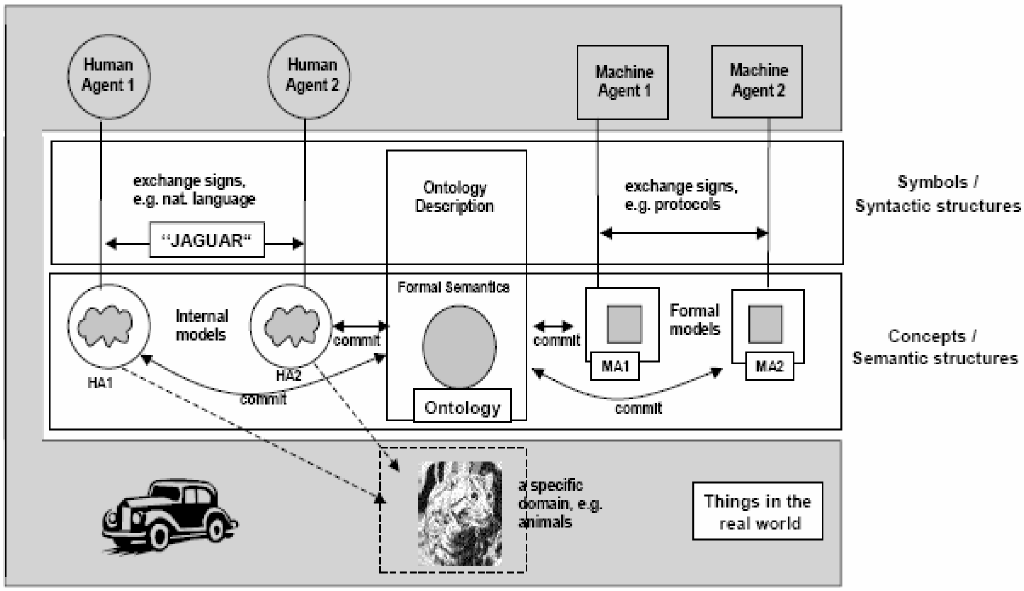


Figure 1.2: Human and Machine Communication

Source: Maedche (2001)

Typically, ontologies are composed of a set of terms representing concepts (hierarchically organized) and some specification of their meaning. The concepts act as a skeleton foundation for the knowledge base.

1.3 Need of ontology in the agriculture domain

India is a developing country and agriculture plays a very important role in its economy. It is among the top two farm producers in the world. Over 70 per cent of the rural households depend on agriculture. Agriculture contributes about 17% to the total GDP and provides employment to over 60% of the Indian population.

A lot of information on agriculture is available on the Web. However, they are scattered and unorganized and cannot be accessed efficiently. In order to have a meaningful extraction of the information from the large corpus of documents present in the Web, there should be a shared understanding of the domain and developing domain ontology for the same addresses the problem.

The conventional search engines that are used today are based on the traditional keyword based search. With Semantic Web, the semantics of what the user is trying to find out comes into play as the software agents are made more "intelligent" while carrying out the required task given by the users rather than just going about the keywords in the user query like the case with conventional search engines. Ontology makes up for a model which plays an important role in the implementation of such type of systems. The structure of information captured by ontologies can be used to share the common understanding of domain among people and software agents (Musen 1992; Gruber 1993). For example, there are various different websites that contain information on fertilizers or provide fertilizer recommendations. Now if there is a common underlying ontology that these websites share, the software agents can extract and aggregate information in a more efficient way. This aggregated information can then be used to answer user queries.

My research work has been driven by the fact that there is a scope of developing ontologies for representing and extracting information about fertilizers. Following are motivations of my research:

- AGROVOC is the most exhaustive and well-established thesaurus available today in the agriculture domain. It covers a lot of terminology related to forestry, fishery, animal husbandry, food, etc. and information related to fertilizer is not properly provided in this thesaurus. Also, the work that has been done in agriculture domain is mostly crop-specific. For example, Agropedia, Thai Rice, etc. Agrovoc has been referred to have a firsthand idea of the hierarchical structure of the concepts.
- It is important to organise the scattered information on fertilizers so that general purpose questions on fertilizers can be answered.
- The fertilizer ontology can always be upgraded by adding more information to it and also it can be made to merge with other generic ontologies such as soil ontology and crop ontology so that questions related to applications of fertilizer based on soil type and crop can be answered.

1.4 Objective

- To develop ontology in fertilizer sub-domain to represent the existing information in fertilizers such as type of fertilizers, nutrient contents, residual effect on the soil, preferred soil type, time of application, method of application, etc.
- There are certain types of information which are not easy to incorporate into ontology or represent using ontology. For example, distributed knowledge is difficult to be entered as an ontological data. Also, we can’t easily transform certain types of

representation into ontology-appropriate formats such as statistical information and diagrammatic knowledge. Our aim is to tackle these difficulties by making use of the different functions present in Protégé and to represent them in the best possible way.

- To reason the knowledge base, make inferences and assert new knowledge by using a reasoner.
- To answer some general questions on fertilizers using SPARQL and DL query on both asserted and inferred information.

The aim of our work is to develop ontology from scratch by consulting domain experts and referring to relevant knowledge base. Our methodology is a mix of knowledge available in Agriculture books, journals, magazines and expert advice.

1.5 Organization of dissertation

The remainder of the work has been organised as follows.

- Chapter 2 discusses the challenges of ontology development in agriculture domain. It also discusses the relevant ontology development methodologies.
- Chapter 3 describes the fertilizer ontology development. This chapter also presents the implementation of the ontology using Protégé as the ontology editing tool.
- Chapter 4 shows reasoning of the ontology using Pellet reasoner and also querying of the knowledge base based on the fertilizer ontology using SAPRQL and DL query languages.
- Chapter 5 provides conclusion and future work.

Chapter 2

Ontology Development Challenges & Methodologies

In this chapter, we talk about the challenges that are commonly faced by an ontology developer during the process of ontology development in agriculture domain. Also, we discuss some of the relevant ontology development methodologies and some work that has been done in the field of agriculture.

2.1 Challenges Related to Ontology Development in Agriculture Domain

The building of ontology from scratch is considered as a time consuming and challenging task. Here are the challenges that are usually faced while developing domain ontology in agriculture domain:

A. Unavailability of authentic resources

The first step in building any domain ontology from scratch is to have an adequate amount of relevant data which is authentic. In a specific domain, finding the relevant authentic data is a big task. Firstly, it requires one to contact the domain experts and need to collect the data upon their suggestions. Secondly, there are a lot of ambiguities when it comes to defining the hierarchy of the terms. For example, in the fertilizer sub-domain, defining a proper hierarchy of the types of fertilizers is a confusing one. Thus, the process of extracting and exploiting the resources becomes harder which leads to problems such as time consuming.

B. Question of which methodology to follow

A number of methodologies have been proposed for ontology development over the years by different researchers. However, due to the relatively immature nature of the field of Ontology Engineering, there is no specific ontology development methodology defined for a particular domain or for a particular task or application for that matter. It is the onus on the ontology developer to find out the methodology that best suits the domain as well as the use of the ontology to be developed.

C. Classification of classes and properties

The task of defining classes for ontology in the agricultural domain is a challenging one. Firstly, there are no proper hierarchy available that we can follow. We have to consider information from all the available sources and see which one or combination of which data best suits our purpose. This task is a very tedious and time consuming task. Again, once the classes are defined, defining its properties is also a big task. We have to read each and every line of the available resources and find out which information can be represented in the form of ontology.

D. Incorporation of statistical data

The domain of agriculture, especially the sub-domain of fertilizer has enormous data in the form of statistical records. There are no specific methods of incorporating such data into ontology. Since the amount of statistical data is too huge, it takes up a lot of time and is an never ending process. The statistical data can always be updated into the ontology.

2.2 Ontology Development and its Relevant Methodologies

2.2.1 Ontology Development Life Cycle

A number of stages are involved in the process of ontology development. The usually accepted stages involved in developing an ontology are specification, conceptualization, formalization, implementation, and maintenance. Pinto and Martins (2004) proposed the following activities to be performed during the various stages of ontology development:

Specification

Identify the purpose and scope of the ontology. The purpose is obtained by answering the question "Why is the ontology being built?" and the scope is obtained by answering the question "What are its intended uses and end users?"

Conceptualization

Using a conceptual model, describe the ontology to be built, so that it meets the specification found in the previous step. Different methodologies propose the use of different conceptual models. The conceptual model of an ontology consists of concepts in the domain and relationships among those concepts. Relationships enhance stronger connections between the concepts.

Formalization

Transform the conceptual description into a formal model, that is, the description of the domain found in the previous step is written in a more formal form, although not yet its final form. Concepts are usually defined through axioms that restrict the possible interpretations for the meaning of those concepts. Concepts are usually hierarchically organized through a structuring relation, such as is-a (class-superclass, instance-class) or part-of.

Implementation

Implement the formalized ontology in a knowledge representation language. For that, one commits to a representation ontology, chooses a representation language and writes the formal model in the representation language using the representation ontology.

Maintenance

Update and correct the implemented ontology.

Some activities that are performed for the whole life cycle are:

Knowledge acquisition

Acquire knowledge about the subject either by using elicitation techniques on domain experts or by referring to relevant bibliography. Several techniques can be used to acquire knowledge, such as brainstorming, interviews, questionnaires, text analysis, etc.

Evaluation

Technically judge the quality of the ontology.

Documentation

Report what was done, how it was done and why it was done.

The activities that are performed during the ontology development life cycle are shown in the figure in the next page:

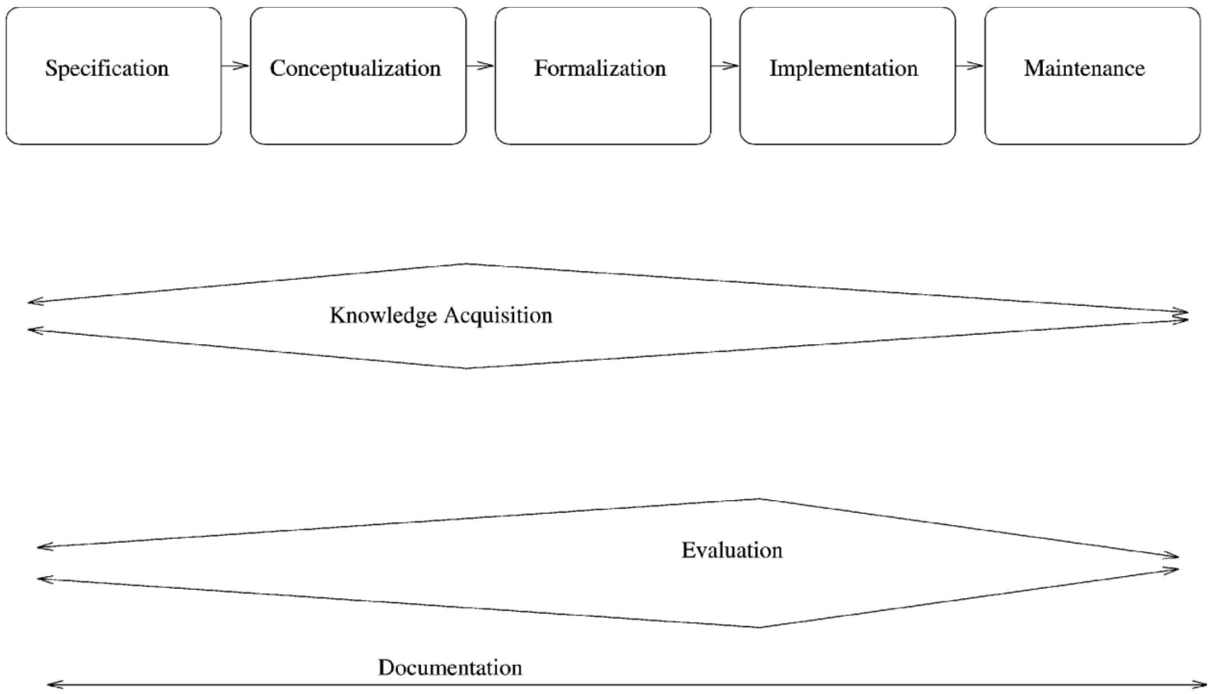


Figure 2.1: Activities in Ontology Development Life Cycle

Source: Pinto et al. (2004)

2.2.2 Relevant Methodologies for Ontology Development

Before we discuss the ontology development methodologies, let us first understand the definition of Ontology Engineering (OE).

*"**Ontology engineering** in computer science and information science is a field which studies the methods and methodologies for building ontologies: formal representations of a set of concepts within a domain and the relationships between those concepts".*

A number of methodologies have been proposed for the development of ontologies by different researchers in the past years. Here, we shall be discussing the most representative methodologies for ontology development from scratch. They are METHONTOLOGY (Fern´andez et al. 1997; Fern´andez et al. 1999), TOVE (Gruninger and Fox 1995; Gruninger 1996), ENTERPRISE (Uschold and King 1995; Uschold 1996), and Ontology Development 101: A Guide to Creating Your First Ontology (Noy and McGuinness 2001).

2.2.2.1 METHONTOLOGY

This methodology was proposed to build ontologies about enterprise modelling processes. It is more recent than the other two methodologies that are discussed below. Since this methodology is highly influenced by the software engineering methodologies, the terminology used for the developmental activities follows the terminology of software engineering. It proposes an evolving prototyping life cycle and its evolving nature lets it to be updated continuously. In this methodology, ontology *reuse* is proposed as one of the activities in the development process. There have been some changes from the earlier versions. For example, the ontology reuse was a stage between formalization and implementation in the earlier versions but it is considered a support activity in the later versions. Later versions also include a *configuration management* activity which maintains a record of different versions of documentation, software and ontology code. It also proposes other activities such as *planning* and *control* for the purpose of project management. The activities in METHONTOLOGY are classified according to the usual terminology in Ontology Engineering as shown in the table below:

Table 2.1: Classification of METHONTOLOGY activities

METHONTOLOGY activity	Corresponds to
Requirement specification	Specification
Conceptualization of domain knowledge	Conceptualization
Formalization of conceptual model	Formalization
Implementation of formal model	Implementation
Maintenance	Maintenance
Knowledge acquisition	Knowledge acquisition
Documentation	Documentation
Evaluation	Evaluation

This methodology believes that ontology building should be done at the knowledge level and not at the symbol level (in formalization) or at the implementation level (when the ontology is codified in a target language), because it is so believed that this will ease (time and effort of) ontology development.

2.2.2.2 TOVE

The TOVE methodology proposes the following stages:

Capture motivating scenarios: This step involves elaborating the motivating scenarios for the ontology. At this stage, possible applications and intended solutions are identified.

Formulate informal competency questions: At this stage, the questions that are supposed to be answered by the knowledge base based on the ontology are formulated. The answer to a question can be used to answer more general questions.

Specify the terminology in a formal language: Since TOVE uses first-order logic (FOL), at this stage, one chooses functions, constants and predicates to be used.

Formulate formal competency questions in FOL: The identified questions are formulated using the terminology defined in the previous stage.

Specify axioms and definitions for the terms in the formal language: Axioms are the restrictions that act as necessary and sufficient conditions to express the competency questions.

Evaluate competency and completeness: The competency and completeness of the ontology is evaluated.

In the table below, the activities of TOVE is classified according to usual terminology in Ontology Engineering.

Table 2.2: Classification of TOVE activities

TOVE activity	Corresponds to
Capture motivating scenarios and formulate informal competency questions	Specification
Specify terminology, formulate formal competency questions and specify axioms and definitions in FOL	Conceptualization, Formalization and Implementation
Evaluate competency and completeness	Evaluation

2.2.2.3 ENTERPRISE

ENTERPRISE proposes the following stages:

Identify the purpose and scope: Identify why the ontology is built, its uses and the intended users.

Build the ontology:

a) Capture Knowledge: Identify the key concepts and relationships in the domains. Give precise text definitions to the concepts and relationships.
b) Code knowledge.
c) Reuse appropriate knowledge from existing ontologies.

Evaluate

Document

Table 5 shows the classification of ENTERPRISE methodology in terms of Ontology Engineering Terminology.

Table 2.3: Classification of ENTERPRISE activities

ENTERPRISE activity	Corresponds to
Identify purpose and scope	Specification
Capturing knowledge	Knowledge Acquisition and Conceptualization
Coding	Formalization and Implementation
Evaluate	Evaluation
Document	Documentation

2.2.2.4 Ontology Development 101

The Ontology Development 101 guide is proposed by Noy and McGuinness. It does not propose an ontology development methodology as such, but the steps by steps procedure that is presented in this guide is recommended and referred to as a good introduction to ontology development for beginners. In this guide, they have taken an example of food and wine ontology and have exemplified each and every tasks involved in the development process. The required tasks such as *Scoping, Reuse, Term enumeration, Class hierarchy construction, Property elicitation, Property definition, and Instance creation* have been discussed to details. There is also an additional section that discusses commonly occurring problems and issues for each step, thereby helping the reader to avoid the common pitfalls that are usually faced during ontology development process. It gives very concrete guidance on some of the practical issues of ontology development, such as the difference between subtyping and subclassing, or the difference between concepts or properties of said concepts thus filling some gaps that the previously discussed methodologies do not cover as all of them aimed at people who are already ontologists.

2.3 Related Work in Ontology Development for Agricultural Domain

During the literature review, it has been found that there are some popular ontologies developed in the agriculture domain.

- **Thai Rice ontology** (Thunkijjanukij et al. 2009): The Thai Rice Ontology is a prototype ontology for plant production using Thai rice as a case study. This ontology covers all the stages of rice production in Thailand starting from cultivation to harvesting. Since there are no well-structured and organized repositories for plant production, researchers face many problems while trying to find relevant information for their research purposes. Hence, Thai Rice ontology has been designed with an aim to facilitate the process of knowledge acquisition and information retrieval for research purposes.
- **User Centered Ontology for Sri Lankan Farmers** (Anusha Walisadeera et al. 2014): The main aim of this work is to provide agricultural information and relevant knowledge that is complete and structured and specific to user context. This ontology has been developed considering the farmers' needs and also taking into account the questions that vary from farmer to farmer such as farm environment, types of farmers, etc.

  There are also many well-established controlled vocabularies in the agricultural domain. Thesaurus can be interpreted as a controlled vocabulary organized in a hierarchy with more information about the concepts including preferred and alternative terms. Fisseha (2002) has identified several limitations and drawbacks with current vocabularies such as semantic ambiguity in definitions and usage of vocabularies; lack of high-level cross-domain concepts; and meaning of their relationships not being precisely defined. One of the most well-established and authoritative controlled vocabulary in agriculture is the AGROVOC.

- **AGROVOC** is a multilingual, structured, controlled vocabulary/thesaurus designed to cover concepts and terminology in agriculture, forestry, fisheries, food, nutrition and related domains developed by the Food and Agriculture Organization (FAO) of the United Nations and the Commission of the European Communities (AGROVOC 2014).
- **Agropedia** is an online knowledge repository for information related to agriculture in India backed by Government of India and sponsored by the World Bank through the *National Agricultural Innovation Project* of the Indian Council of Agricultural Research (ICAR) (Agropedia 2014). It is more useful for those users whose needs are mainly based on crops as the repository is maintained in such a way that it is focused on crops and the related information crop wise. Its main aim is to keep alerts on different crops from the scientists and make the farmers aware of it by keeping them updated through text messages. It is a crop wise based knowledge repository. The results retrieved will only be from its repository. The knowledge models given are from a crop point of view and there is need for ontologies that deal from other concepts point of view to have a better coverage of the ontologies on agriculture domain.

Chapter 3

Development of Fertilizer Ontology: FertOnt

Since the recognition of use of ontology in the field of computer science, the process of ontology development is found to be creative and iterative in nature. No two ontologies designed by two different people will be same. The ontology design choices are dependent on the potential applications of the ontology and the designer's understanding of the specific domain for which the ontology is being developed.

Components of an ontology

Gruber (1993) identified five kinds of ontology components: classes, relations, functions, formal axioms and instances.

1. Classes represent concepts, which can be considered generic entities in the broad sense.
2. Relations represent a type of association between concepts of the domain.
3. Functions are a special case of relations.
4. Formal axioms serve to model sentences that are always true. They are normally used to represent knowledge that cannot be formally defined by other ontology components.
5. Instances are used to represent elements or individuals in an ontology.

Noy and McGuinness (2001) described an ontology as a formal explicit description of concepts in a domain of discourse, properties of each concept describing various features and attributes of the concept , and restrictions on slots.

3.1 Ontology Development

Noy and McGuinness (2001) proposed seven steps to develop ontology:
Step 1. Determine the domain and scope of the ontology
Step 2. Consider reusing existing ontologies
Step 3. Enumerate important terms in the ontology
Step 4. Define the classes and the class hierarchy
Step 5. Define the properties of classes—slots
Step 6. Define the facets of the slots

Step 7. Create instances

Three different approaches have been proposed for the development of ontology (Uschold and Gruninger 1996):

1. Top-down approach: In this approach, development process considers the most general concepts in the domain and define them and then go on specializing into specific concepts.

2. Bottom-up approach: In this approach, development process followed is the reverse of that of the top down approach. It starts with the most specific concepts and defining them, followed by subsequent grouping of these concepts into more general concepts.

3. Combination approach: This approach is a combination of the top-down and bottom up approaches by defining the more salient concepts first and then generalize and specialize them appropriately.

There are three basic approaches to construct an ontology, namely;

1. *Manually-driven by domain-experts*: This approach relies totally on experts of the field. The experts will set the rules and concepts and sort of relationships about words and their relationship based upon experts' knowledge and experience of the knowledge domain.

2. *Automatic approach*: This approach will construct ontology by using a computer program, whereas the program will be produced according to rules and conditions laid out by developer with the help of experts and the computer.

3. *Semi-automatic approach*: This approach also uses computer program but the ontology builder will have product's accuracy and rules verified and confirmed by expert who created the rules. Semi-automatic processing of complex data is becoming possible to extract hidden and useful pieces of knowledge which can be further used for different purpose.

3.2 Development of FertOnt

Keeping in mind the rules proposed by Noy and McGuinness and the various stages involved in the ontology development, the fertilizer ontology is built from scratch. The approach selected is manually-driven. The tool that we are using for the development of FertOnt is Protégé 4.3 (Build 304).

Protégé is a free, open source ontology editor and a knowledge acquisition system and is being developed at Stanford University in collaboration with University of Manchester. It supports lot of plug-ins like Pellet reasoner, SPARQL, DL query, etc. which add extra functionalities. It also exports ontology in many formats (RDFS, OWL, etc). Out of the various phases that have been listed in the previous chapter, the phases that we would be considering for the development of our ontology are as follows:

3.2.1 Ontology Specification

The first step in the ontology construction is to define domain and scope. In order to define the domain and scope of the ontology, two kinds of questions are sketched:

1. Basic questions – this is a list of questions which define the purpose of the ontology and help to limit the scope of the domain. The basic questions that we have answered are:
    a. What is the domain that the ontology will cover?
    The ontology will cover the fertilizer sub-domain in the field of agriculture. It covers only the static data about fertilizers. The dynamic data such as the market prices, information about the places to buy and sell products, consumer behaviour are not the scope to be covered.
    b. For what we are going to use the ontology?
    Agriculture is a very vast domain and as far as the sub-domain of fertilizer is concerned, it has always been studied in relation to crop and soil. But fertilizer in itself is separate entity and it should be studied separately with a more generic approach. The ontology developed can be used as a knowledge base for the information on fertilizers alone unlike the other knowledge bases where it has been related to the other sub-domains of agriculture. Further this ontology can be then merged with crop and soil ontologies so that questions related to application of fertilizer based on crop and soil can be answered.

c. For what type of questions the information in the ontology should provide answers?

   The ontology should be able to answer basic questions related to fertilizer itself, such as the types of fertilizers, its usage and application and the statistical information such as its production, consumption, import, etc.

d. Who will use and maintain the ontology?

   Researchers who are working in the field of domain specific ontologies and academicians and experts who want to develop and maintain knowledge based structure in agricultural domain will use the ontology. The research scholar team at JNU in collaboration with the domain experts will maintain the ontology.

2. Competency questions – these are the questions that the knowledge base based on the ontology should be able to answer. These questions let us know if the ontology has enough information about the domain or if there is a need for a particular level of detail or representation of a particular area. For our research, the list of competency questions was framed by consulting the domain experts at NCAP,ISRI. As and when the ontology gets upgraded and matures with time, the competency questions can be generalized. This would provide a boarder range of questions that can be answered following the same structure. Some of the competency questions along with the corresponding generalized questions are given below:

Table 3.1: List of competency questions with the corresponding generalized questions

Generalized Question	**Competency Question**
What is the concerned nutrient content in the corresponding fertilizers?	What is the specified nutrient content in various specified nutrient fertilizers?
Which type of phosphatic fertilizers are suitable for a specified type of soil?	Which phosphatic fertilizers are suitable for acidic soil?
Which nitrogenous fertilizers use a specified type of application method?	Which nitrogenous fertilizers use Topdressing type of fertilizer application method?

List the phosphatic fertilizers that contain phosphorus in the form of a specified type of phosphate form.	List the phosphatic fertilizers that contain phosphorus in the form of dicalcium phosphate.
What is the specified nutrient production and import for a specified year?	What is the N production and import for the year 2004-2005?
What is the preferred time of application of a specified fertilizer type?	What is the preferred time of application of basic slag?
In what form is phosphorus present in a specified phosphatic fertilizer?	In what form is phosphorus present in Basic Slag?
What is the specified nutrient consumption for a specified state during a specified year?	What is the total N, P and K consumption for the state of Assam during the year 2011-2012?

Looking at the competency questions above, the ontology will contain information on the various types and subtypes of fertilizers available, the application time of different fertilizers and the statistical information of the fertilizers.

3.2.2 Knowledge Acquisition

This step is about extracting the domain specific knowledge from the knowledge resources as much as possible. In order to fulfil this need, we have gone through numerous online sources, books and contacted domain experts. we have extracted the knowledge required for the fertilizer ontology development from the following reliable sources:

- Domain experts at NCAP, ISRI, New Delhi;
- Research journals and papers;
- Text books (Gupta 2003; Chanda et al. 2012);
- Online data sources;
- Mass media (newspaper, television, radio).

3.2.3 Ontology Conceptualization

This step involves building of the conceptual model by following the specification listed in the previous step. Ontology is a data model that represents the concepts related to the domain and the relationships among these concepts. And so, as pointed out earlier also, concepts and relations are the main components of an ontology. Relationships can be of two types: hierarchical relationships and associative relationship. Hierarchical relationships are those between the concepts of the same hierarchy, i.e., between superclass and subclass. Associative relationships are the relationships between concepts which are in different hierarchy.

For conceptualization, much of the work is done following the different steps proposed by Noy and McGuinness (2001). The steps are as follows:

Consider reusing existing ontology

We have developed the ontology from scratch since there are no existing ontologies that fits our requirements.

Enumerate different terms in the ontology

The next step is to write down an exhaustive list of terms that the fertilizer sub-domain covers. For example, the terms that usually come to mind when we think of fertilizer are chemical fertilizer, manure, compost, dosage, biofertilizer, application time, application method, and so on. The handbook (Gupta 2003) was referred to in making the list of terms of our interest.

Define the classes and class hierarchy

Uschold and Gruninger (1996) proposed different approaches for the development of ontology:

- Top –down approach
- Bottom-up approach
- Combination approach

No approach is better than the other two. The approach to take depends on the understanding of the domain to the developer and his convenience. We have followed the combination approach for development of ontology which is a combination of top-down and bottom-up approaches. From the list created in the previous step, we have selected certain concepts we are familiar with and then specialized and generalised them appropriately. For example, we started with a top-level concept 'Fertilizer', and a more specific concept 'AmideFertilizer'. And then we could relate them to a middle-level concept, 'NitrogenousFertilizer'.

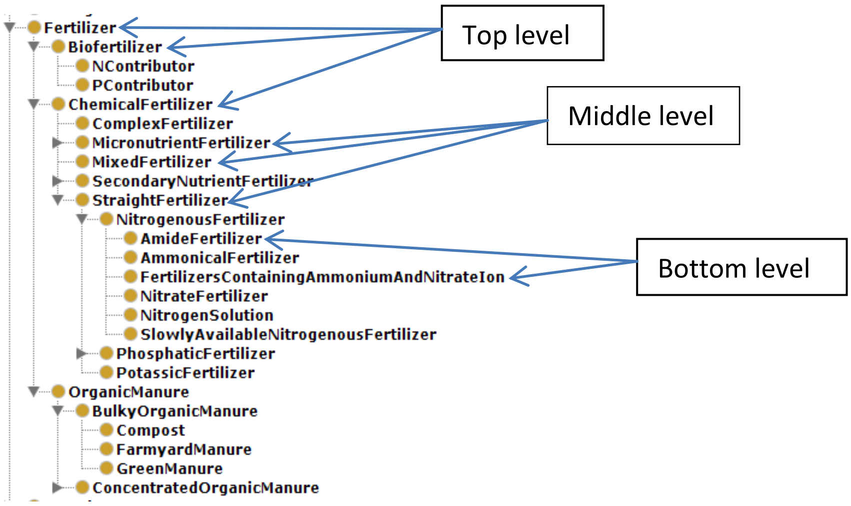


Figure 3.1: Taxonomy of the concept 'Fertilizer'. 'Fertilizer' is the most general concept. 'Biofertilizer', 'ChemicalFertilizer', etc are the general top level concepts. 'AmideFertilizer', 'AmmonicalFertilizer', etc are the bottom level concepts.

Noy and McGuinness had discussed several guidelines to keep in mind while developing a class hierarchy. We have considered these guidelines and have used them to check against the class hierarchy that we have created for our ontology.

They had set various rules of thumb that help us in deciding the class hierarchy.

Subclasses of a class usually (1) have additional properties that the superclass does not have, or (2) restrictions different from those of the superclass, or (3) participate in different relationships than the superclasses.

Keeping these rules in mind, we created our class hierarchy and is as follows:

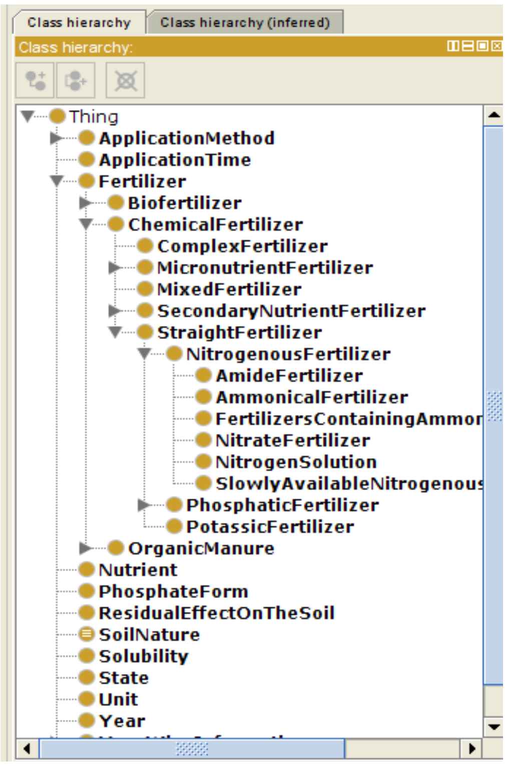


Figure 3.2: Fertilizer class hierarchy

A new class is introduced when there is something that can be said about the new class but the same is not true for its superclass. For example, biofertilizers can have different organisms involved for its production and usage, whereas for fertilizers in general, this property has no significance and is not used to describe them. The nitrogenous fertilizers have a property that describes the nitrogen nutrient content in them which is not a very useful property of the chemical fertilizers in general. Therefore, biofertilizer and nitrogenous fertilizer are subclasses of fertilizer and chemical fertilizer respectively.
It is however, sometimes important to define new classes even if there are no new properties added for them. For example, the class 'ApplicationMethod' has subclasses 'LiquidApplicationMethod' and 'SolidApplicationMethod'. This classification is just a hierarchy and the subclasses have the same set of properties.

Also, deciding whether a specific distinction should be set as a property value or a set of classes depends on the scope of the ontology and its use. For example, deciding whether 'NitrogenousFertilizer' should be a separate class or should it be made a class 'ChemicalFertilizer' and fill the value for the property 'NutrientContained' as Nitrogen, depends on the scope of the domain. Now the distinction of Nitrogenous, Potassic and

Phosphorus fertilizers is very important since we are developing a detailed ontology of fertilizer. Hence this distinction leads to the subclasses 'NitrogenousFertilizer', 'PotassicFertilizer' and 'PhosphoricFertilizer'.

Similarly, deciding whether a particular concept should be a class or an individual instance of a class depends on the possible application of the ontology and scope of the domain. The ontology that we are building is a generic ontology in the domain of agriculture with fertilizer sub-domain as the case study, it should have enough coverage of the information in the sub-domain of fertilizer. In order to decide the level of granularity, i.e. the possible terms which will act as the individuals, there is a need to go back to the competency questions that we have identified earlier. The most specific terms which answer these questions are the possible candidates for individuals.

"Individual instances are the most specific concepts represented in a knowledge base".

For example, the different types of Amide fertilizers are the most specific terms that answer one of the competency questions. Therefore, they are individual instances in our knowledge base. Also, the types of application methods as well as the possible periods of application time have also become individuals for the knowledge base.

Identify relationships

a. There is only one relation for the hierarchical relationships, namely “hasSubclass”. This relation is defined between all of the hierarchical concepts. Some of the hierarchical relationships are given in the table below:

Table 3.2: Concept and Hierarchical relationship

Subject concept	Relation	Object concept
Fertilizer	hasSubclass	Chemical Fertilizer
Chemical Fertilizer	hasSubclass	Straight Fertilizer
Straight Fertilizer	hasSubclass	Nitrogenous Fertilizer
Nitrogenous Fertilizer	hasSubclass	Amide Fertilizer
Amide Fertilizer	hasSubclass	Urea

b. Associative relationships are assigned between concepts belonging to different hierarchies. These relationships are defined by identifying verbs related between concepts and assigning relation name that would form a meaningful statement with the name of the concepts. Some of the examples of associative relationships are given in the table below:

Table 3.3: Concept and Associative relationships

Subject concept	Relation	Object concept
Fertilizer	hasApplicationMethod	Application Method
Phosphatic fertilizer	hasSuitableFertilizer	Soil Nature
Phosphatic Fertilizer	containsPInTheFormOf	Phosphate Form
Urea	hasChemicalFormula	$CO(NH_2)_2$
Urea	hasPercentN	46

Define the properties of classes-slots

In order to answer the competency questions defined previously, we must describe properties of the classes. There are two types of properties:

- *Data Properties:* These are the properties that relate individuals to a user-defined value.
- *Object Properties:* These are the properties that relate the individuals of a class to individuals of another class.

From the terms that we have enumerated earlier, after selecting the classes, most of the terms that are left are the properties of these classes. For example, for the term *Fertilizer*, its properties would be a fertilizer's *nutrient content, application method, application time,* etc.

Figure 3.3: Data properties

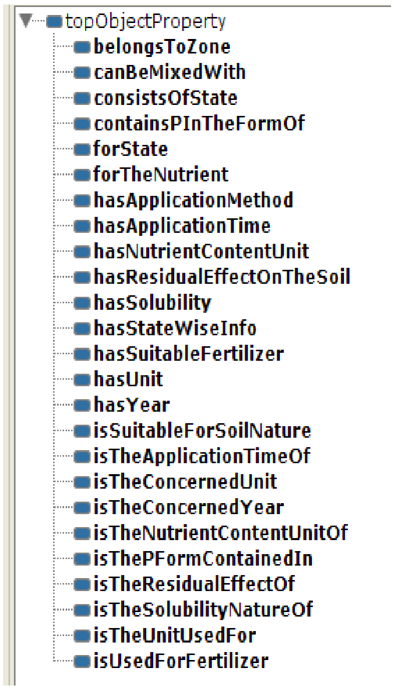


Figure 3.4: Object Properties

Create instances

Creating instances of classes is the last step. We have selected a class and created an individual of that class and filled the values of its properties. For example, we created an individual instance 'BasicSlag' of the class 'PhosphoricFertilizer'. Few properties of this instance are as below:

canBeMixedWith: RockPhosphate

canBeMixedWith: PotassiumSulphate

hasApplicationTime: IllBeforeSowingTheCrop

hasApplicationMethod: BroadcastingAtPlanting

containsPInTheFormOf: DicalciumPhosphateForm

containsPInTheFormOf: CitricAcidSolublePhosphoricAcidForm

hasPercentP: 3-8

Create informal draft models by using the previous summarized knowledge

The tool that we have used to view the knowledge model is the Ontograf which comes as a plug-in with the Protégé tool.

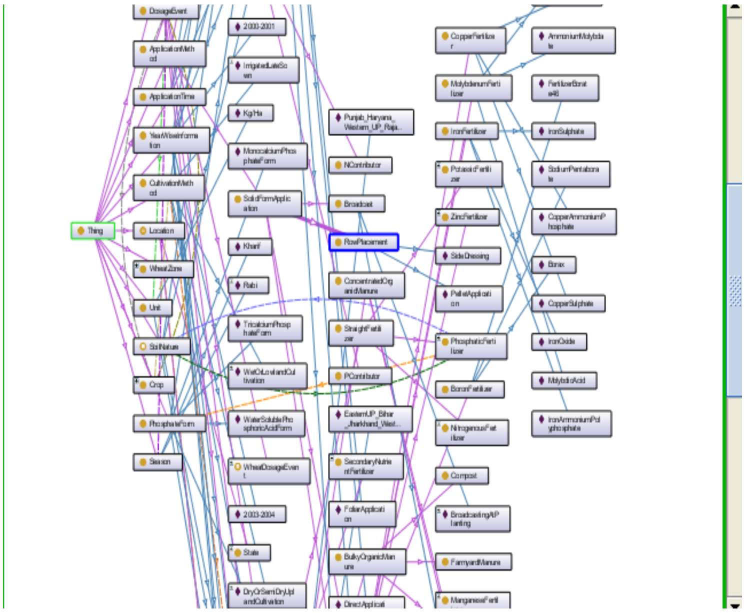


Figure 3.5: View of the ontology using OntoGraf

3.2.4 Ontology Formalization

In this stage, the concepts are defined through axioms. Axioms in OWL ontology provide explicit logical assertions about three types of things - classes, individuals and properties. Many types of axioms can be expressed in OWL ontology. We have used the OWL functional syntax format for easy understanding of the axioms.

Class Declarations

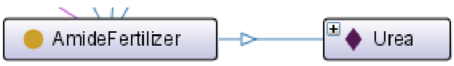
- *Class declaration* defines a class. A class may contain individuals.


```
Declaration(Class(:Fertilizer))
```

- *Individual declaration* defines a named individual.

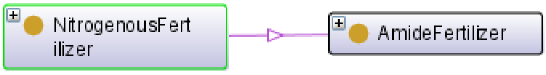

```
Declaration(NamedIndividual(:Urea))
```

- *Class assertion* declares that an individual belongs to a class.




```
ClassAssertion(:AmideFertilizer :Urea)
```

- *Subclass assertion* states that all individuals that belong to a class also belong to its superclass.




```
SubClassOf(:AmideFertilizer :NitrogenousFertilizer)
```

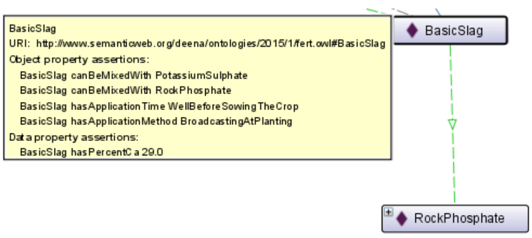
- *Property declaration* states either an object property to link an individual to an individual, or a data property to link an individual to data.


```
Declaration(ObjectProperty(:canBeMixedWith))

Declaration(DataProperty(:hasChemicalFormula))
```

- *Property assertion* defines the relation of an individual to an individual or data.


```
ObjectPropertyAssertion(:canBeMixedWith :BasicSlag :RockPhosphate)
```




```
DataPropertyAssertion(:hasPercentCa :BasicSlag "29.0"^^xsd:double)
```

- *Annotation assertion* lets to annotate anything with some details. For example, we can use SSP as the name of an individual because it is an abbreviation and use annotation to label it with the full form "SingleSuperPhosphate". The images generated by Protégé then show the label instead of the name.

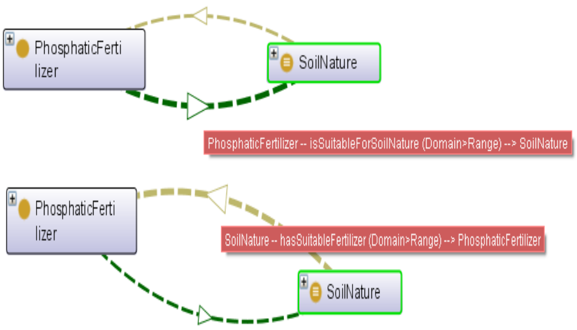

```
Declaration(NamedIndividual(:SSP))
AnnotationAssertion(rdfs:label :SSP :SingleSuperPhosphate)
```


```
Declaration(NamedIndividual(:MOP))
AnnotationAssertion(rdfs:label :MOP :MuriateOfPotash)
```

Property axioms

Many things can be defined about properties. The property characteristics that can be defined are as follows:

- Inverse - If a property x has inverse property y, and if x relates A-B, then it can be inferred that y relates B-A.



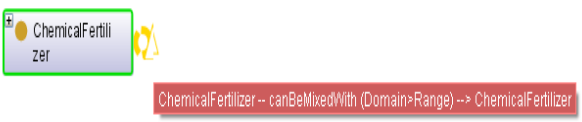

```
InverseObjectProperties(:isSuitableForSoilNature :hasSuitableFertilizer)
```

- Functional - If a property x has functional property then it can have only one value. Inverse Functional - The inverse of the property is functional.


```
FunctionalObjectProperty(:hasYear)
```

- Symmetric - If a property x relates A-B, then it also relates B-A.

  Asymmetric - If a property x relates A-B, then it does not relate B-A.

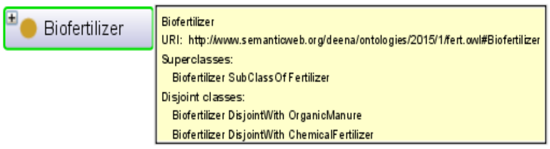



```
SymmetricObjectProperty(:canBeMixedWith)
```

The property 'canBeMixedWith' is defined as symmetric with range and domain as 'ChemicalFertilizer'.

Class expressions

Classes may be defined using class expressions. we can define that classes are disjoined with other class meaning that they do not share any individuals. For example, we can make the class 'Biofertilizer' to be disjoint with 'ChemicalFertilizer' and 'OrganicFertilizer' because 'Biofertilizer' does not share any individuals with either 'ChemicalFertilizer' or 'OrganicFertilizer'.




```
DisjointClasses(:Biofertilizer :ChemicalFertilizer)
DisjointClasses(:Biofertilizer :OrganicManure)
```

There are other properties also that can be defined on the classes. Set operations like union, enumeration, intersection and complement can be defined but since these functions are of less significance in our domain of interest, they have not been used.

3.2.5 Ontology Implementation

There are many ontology implementation languages available. The language that we are using is OWL (Web Ontology Language).

OWL

The OWL Web Ontology Language is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of web content than that supported by XML, RDF, and RDF Schema (RDFS) by providing additional vocabulary along with a formal semantics. There are three different flavours of OWL as given below:

- OWL Lite supports those users primarily needing a classification hierarchy and simple constraints. Owl Lite has a lower formal complexity than other type of OWL.
- OWL DL supports those users who want the maximum expressiveness while retaining computational completeness and decidability. OWL DL is so named due to its correspondence with Description Logics, a field of research that has studied the logics that form the formal foundation of OWL.
- OWL Full is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. OWL Full allows an ontology to augment the meaning of the pre-defined (RDF or OWL) vocabulary.

Components of OWL

- Individuals - Individuals are also known as instances. Individuals can be referred to as being instances of class.
- Properties - Properties are roughly equivalent to slots .They are also known as roles in description logics and relations in UML and other object oriented notions and attributes in other formalisms.
- Classes - The word concept is sometimes used in place of class. Classes are a concrete representation of concepts.

The language adopted is OWL DL as it provides more expressive restriction constructs than OWL Lite and the reasoning support is more predictable than that of OWL Full.

The tool used to create the ontology is Protégé Ontology Editor. Basic steps in creating OWL ontology in Protégé are:

Step 1: Start Protégé.

Step 2: When the editor opens, a default IRI will be assigned for the ontology such as http://www.semanticweb.org/hp/ontologies/2015/5/untitled-ontology-62. You can always change this IRI into your personal web space if you have one and give it a name.

Step 3: You will also want to save the ontology into your personal file on your PC. Click 'save as' from the file menu. You will be asked to select the ontology format in which you want to save your ontology. Select the desired format and save it in your personal PC. You can do this by browsing the hard disk and saving your ontology to a new file by giving it a name like 'fert.owl'. Once a file is chosen press save.

The components of OWL ontology are defined as shown below in Protégé:

- Classes
  - Classes are nothing but the concepts in the domain. Eg. 'Fertilizer', 'ApplicationMethod', 'TimeOfApplication', 'YearWiseInformation', etc.
  - Classes can be created by clicking by Add subclass button after selecting the default root class Thing.

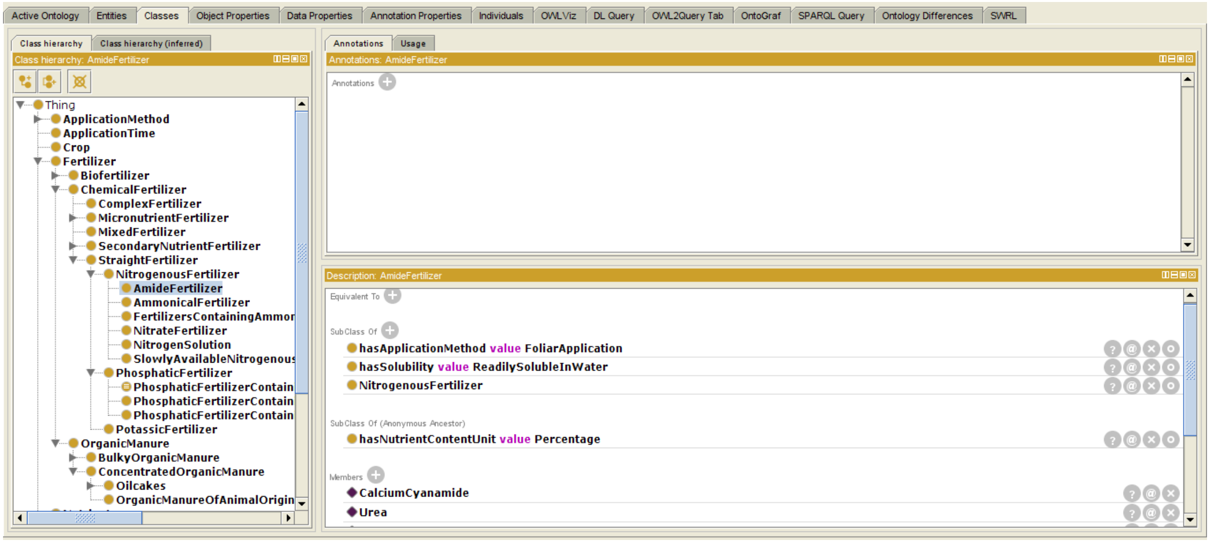


Figure 3.6: Screenshot of creating classes in Protégé

To create sub-classes, the same procedure is followed. First, the class for which sub-classes are to be created is selected and then the Add subclass button on the left is clicked. Subclasses such as 'StraightFertilizer', 'NitrogenousFertilizer', 'AmideFertilizer' are created by selecting the respective classes.

- Individual
  - ➢ Individuals are the instances of the classes. They exhibit all the properties which their class has. E.g. 'BasicSlag', 'BoneMeal', 'DicalciumPhosphate', etc are individuals of the class 'PhosphatiFertilizer'.
  - ➢ They are created by selecting the class for which individuals are to be created. Then clicking on the member 

, then adding the individual to the individual list by clicking on .
  - ➢ 'BasicSlag' is created as an individual of 'PhosphaticFertilizer' and 'Phosphatic Fertilizer' is linked to individuals of other classes through different properties. For example, 'BasicSlag' is an individual that is linked to 'PotassiumSulphate' which is an individual of class 'PotassicFertilizer' by 'canBeMixedWith' property.

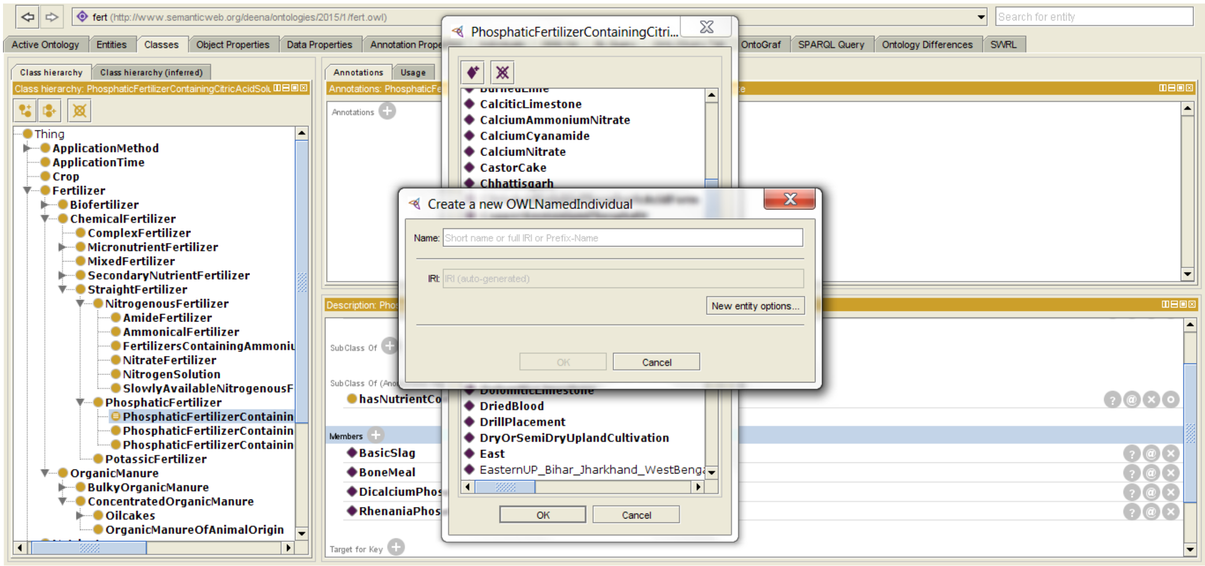


Figure 3.7: Screenshot of adding instances for classes in Protégé

- Object Property
    - ➢ Object property is the relation that exists between two classes. Any object property has a domain class and a range class. The classes to which a property is attached or a class which property is being described, are called the domain of the property. Allowed classes for properties of type Instance are called the range of the property. For example, 'hasApplicationMethod', 'hasApplicationTime' are object property with domain 'Fertilizer' and range 'ApplicationMethod' and 'ApplicationTime' respectively.
    - ➢ Object properties can be added by selecting the Object Properties tab and clicking on the  . The domain and range classes attached with a particular object property are also added.

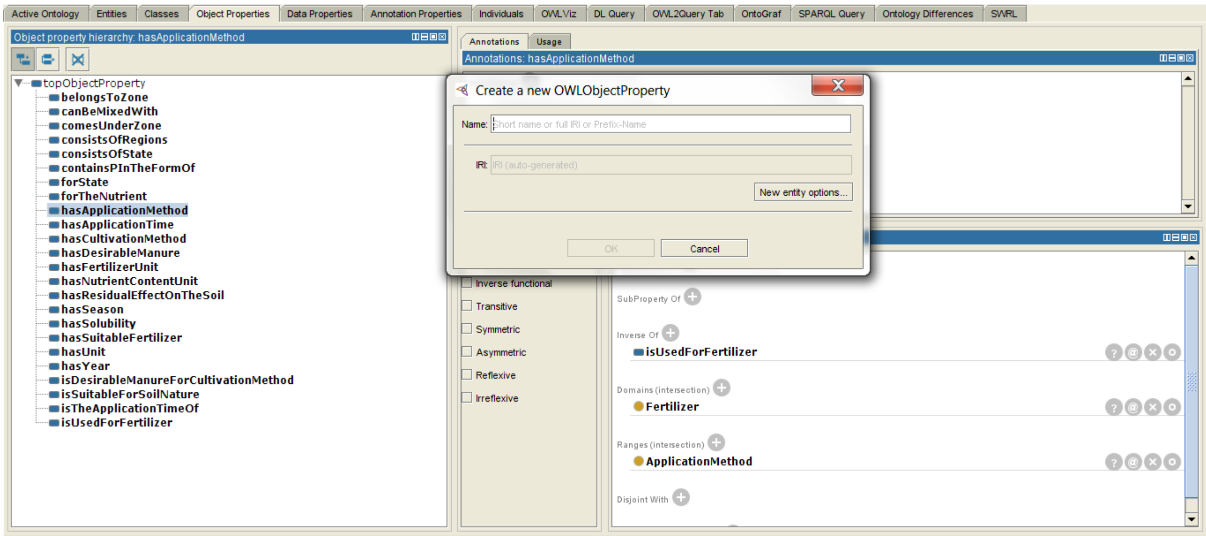


Figure 3.8: Screenshot of adding object properties with their range and domain in Protégé

- Datatype Property
  - ➢ Datatype property of a class is a relation between individuals of that class and datatype values. The values can be Integer, Float, String, Boolean, etc. For example, 'hasChemicalFormula' is a datatype property of the class 'ChemicalFertilizer' with datatype value String.
  - ➢ We can add datatype property by clicking on the Datatype Properties tab in the same way as we did for object properties above.

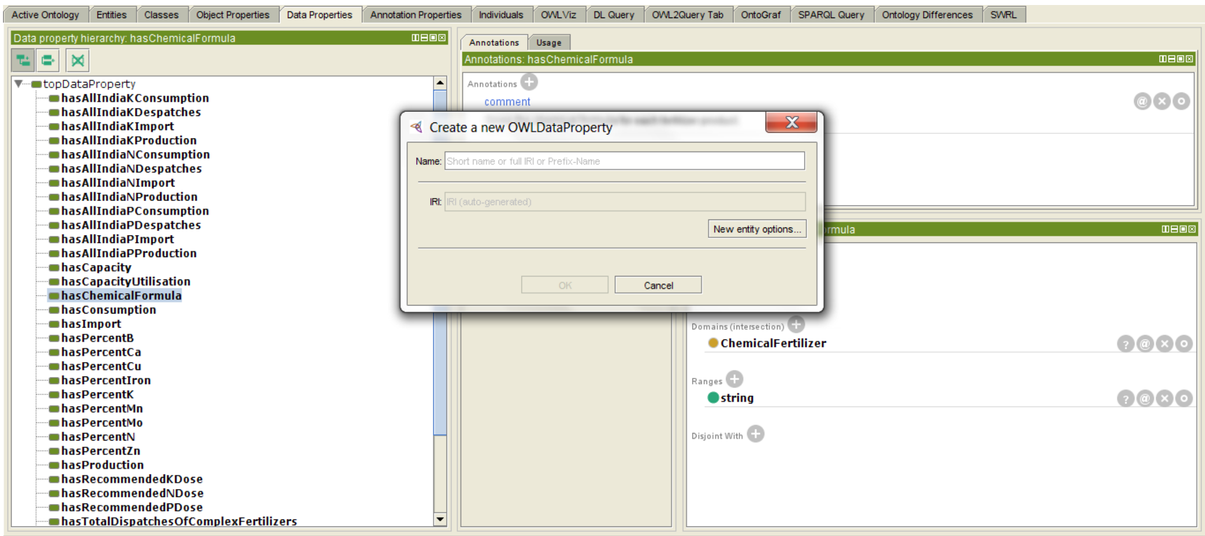


Figure 3.9: Screenshot of adding data properties with domain and range in Protégé

3.2.6 Ontology Validation

For the purpose of validation of FertOnt ontology, a tool named OOPS! was used. It is a web-based tool used to detect anomalies in ontologies and is independent of any implementation language used for the ontology development. The tool tries to find as many anomalies as possible in ontologies. It is able to identify as many as 34 common pitfalls that appear while developing ontologies. The screenshots below show how FertOnt was validated using OOPS! tool.

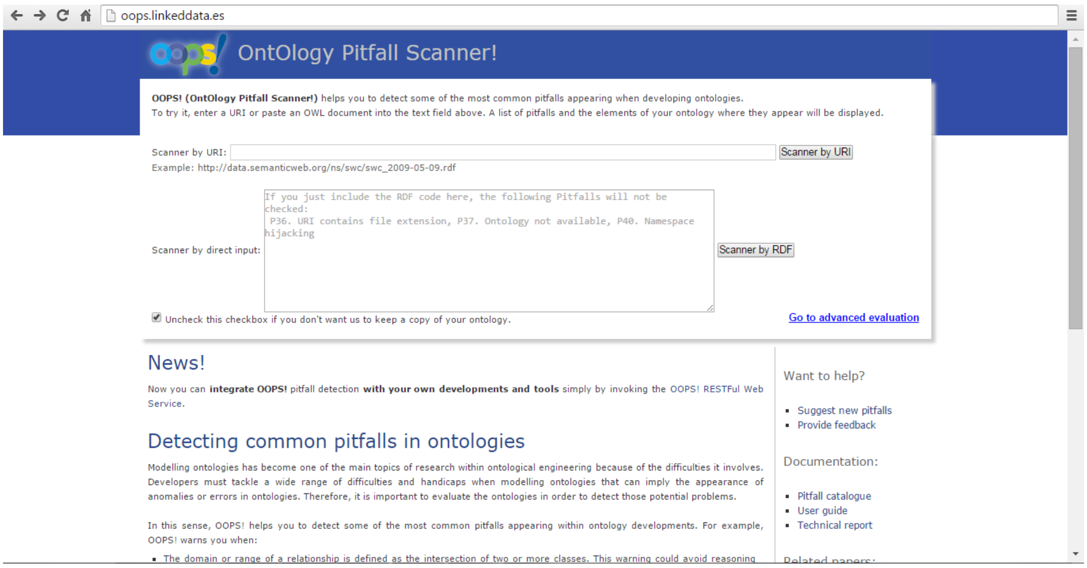


Figure 3.10: Screenshot of the home page of OOPS!

The source code of FertOnt was loaded and was validated. The ontology was analyzed and the results were displayed. The anomalies are highlighted in grey colour. In the first run, few anomalies were detected as shown:

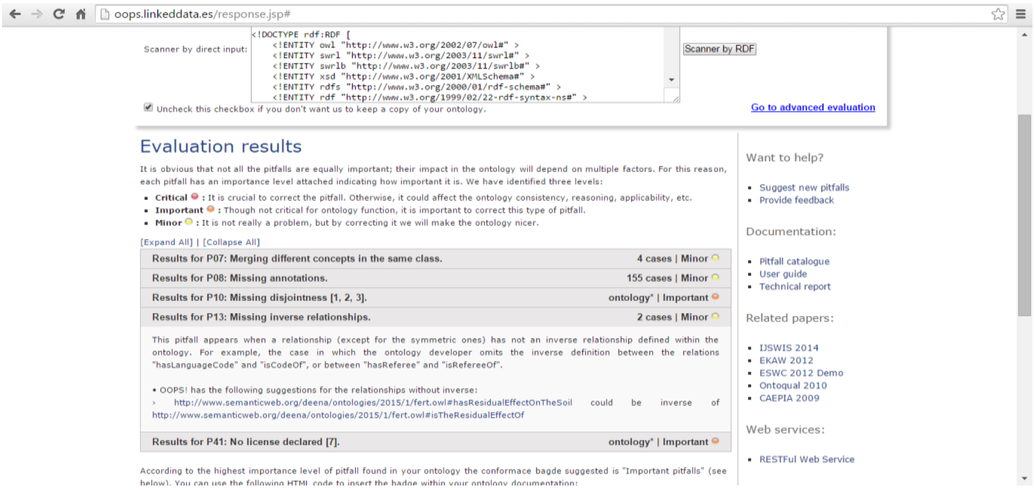


Figure 3.11: Screenshot of the anomalies detected in FertOnt by OOPS!

Once the pitfalls are identified, respective changes are made to make FertOnt conformed to the best ontology modelling practices.

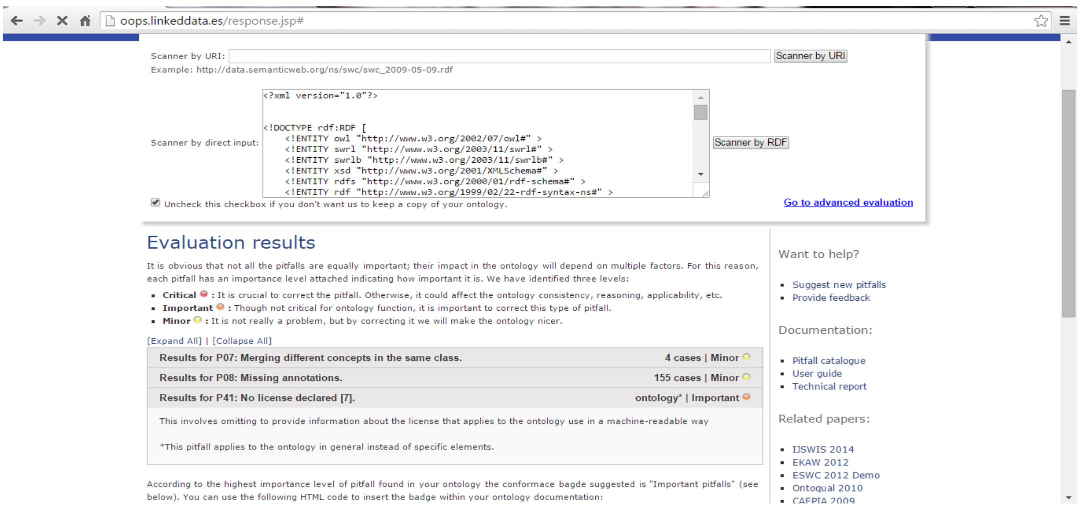


Figure 3.12: Screenshot after anomalies are rectified in FertOnt

Chapter 4

Reasoning and Querying FertOnt

Reasoning in ontology is deriving facts which are not explicitly expressed in the ontology but can be inferred from the asserted data. A specification is made a formal one so as to make reasoning in ontology possible. A reasoner is required to carry out the reasoning and to derive inferred data. We shall also be answering some competency questions by querying the knowledge base based on FertOnt.

4.1 Reasoning FertOnt

The reasoner that we have chosen is called **Pellet**. Pellet comes as a plug-in with the Protégé tool. It is one of the most common reasoning engines used for reasoning with Protégé OWL models. Pellet provides functionality to check consistency of ontologies, compute the classification hierarchy, explain inferences, and answer SPARQL queries.

Examples of certain information that can be inferred from asserted data in FertOnt using Pellet are shown below:

Example 1: For the first type of phosphatic fertilizer, it was asserted that it contains phosphate in the form of dicalcium phosphate form or citric acid soluble phosphoric acid form.

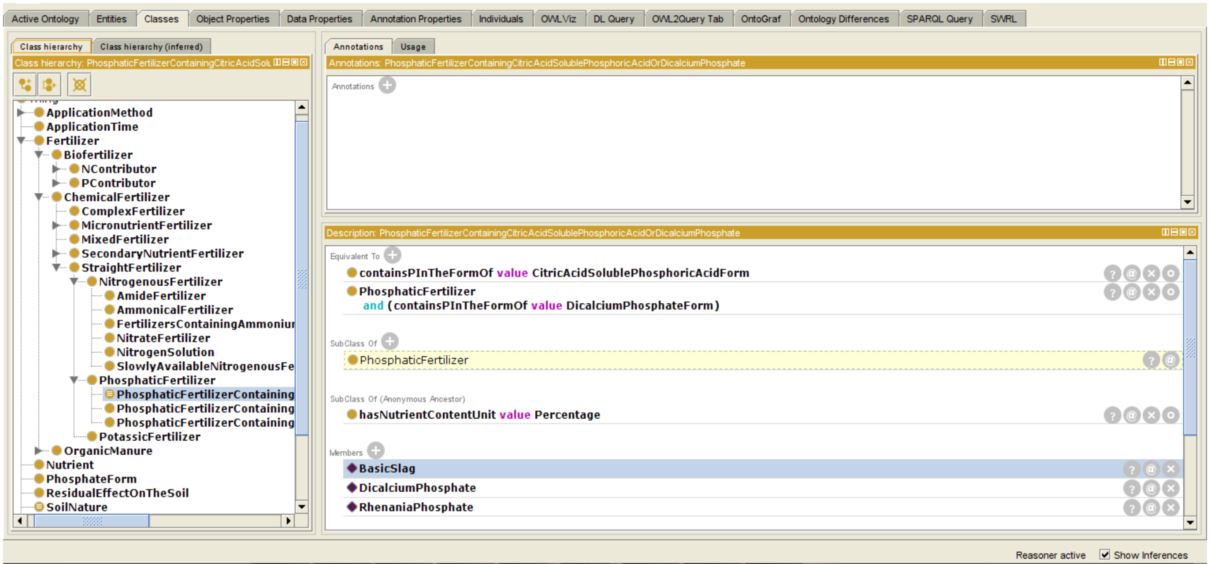


Figure 4.1: A class of phosphatic fertilizer showing the asserted data in Protégé

The reasoner was synchronized to check if all the statements defined in the ontology were logically correct and to derive the inferred data. Now, 'BasicSlag' is a type of phosphatic fertilizer which is an instance of this particular class of phosphatic fertilizer. When we clicked on 'BasicSlag', we could see the inferred data on the right hand side inside the dotted box.

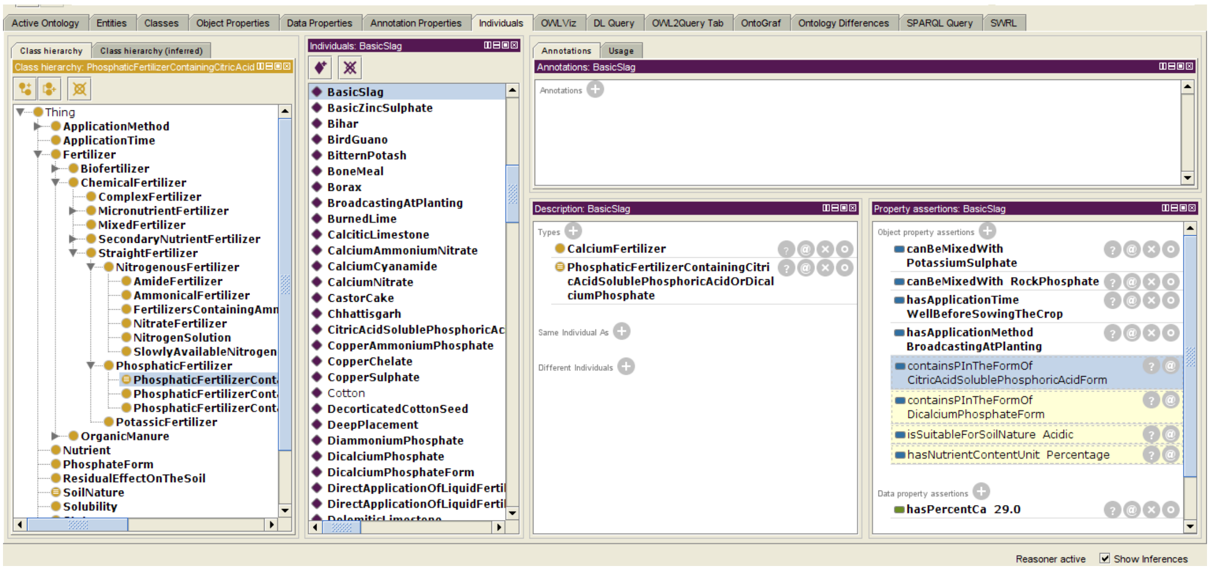


Figure 4.2: Screenshot showing the inferred data for 'BasicSlag' in Protégé

Example 2: We have defined the Amide fertilizers to be a subclass of nitrogenous fertilizer with the specifications defined as shown in the screenshot:

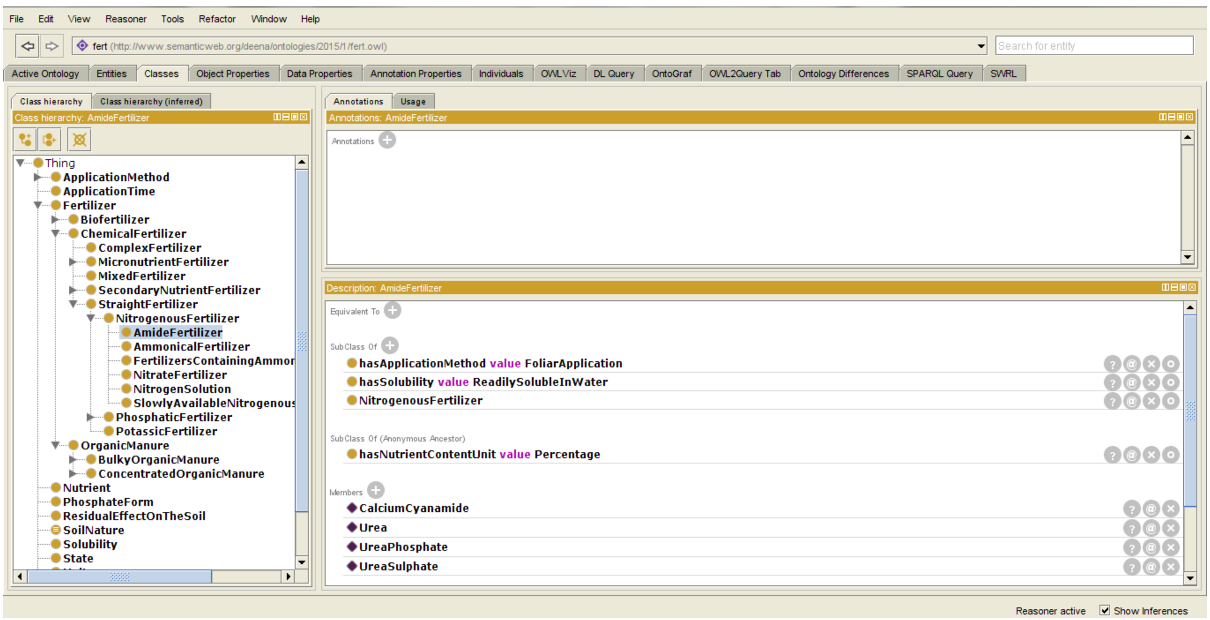


Figure 4.3: Screenshot showing asserted data for the class 'AmideFertilizer'

'Urea' which is an intance of the class 'AmideFertilizer' was then inferred to have the properties which were set for the class 'AmideFertilizer'. These properties can then be queried using a query language.

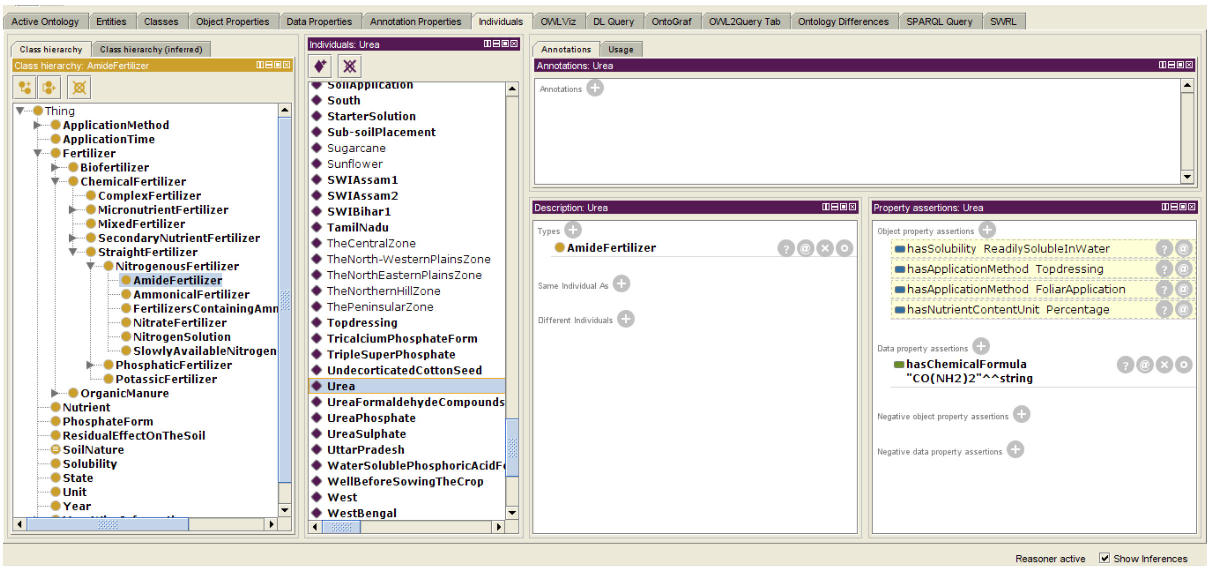


Figure 4.4: Screenshot showing inferred data for the instance 'Urea'

4.2 Querying the FertOnt

The knowledge base based on FertOnt can be queried using query languages. Some of the competency questions which were framed in chapter 3 are queried and answered here. We have used SPARQL to answer the competency questions based on the asserted data and DL Query language to answer the competency questions based on the inferred data.

4.2.1 Querying the asserted data using SPARQL

- CQ1: What is the calcium content in various calcium fertilizers?

  SPARQL Query: SELECT ?Calciumertilizer ?CaContent

  WHERE { ?CalciumFertilizer fer:hasPercentCa ?CaContent }

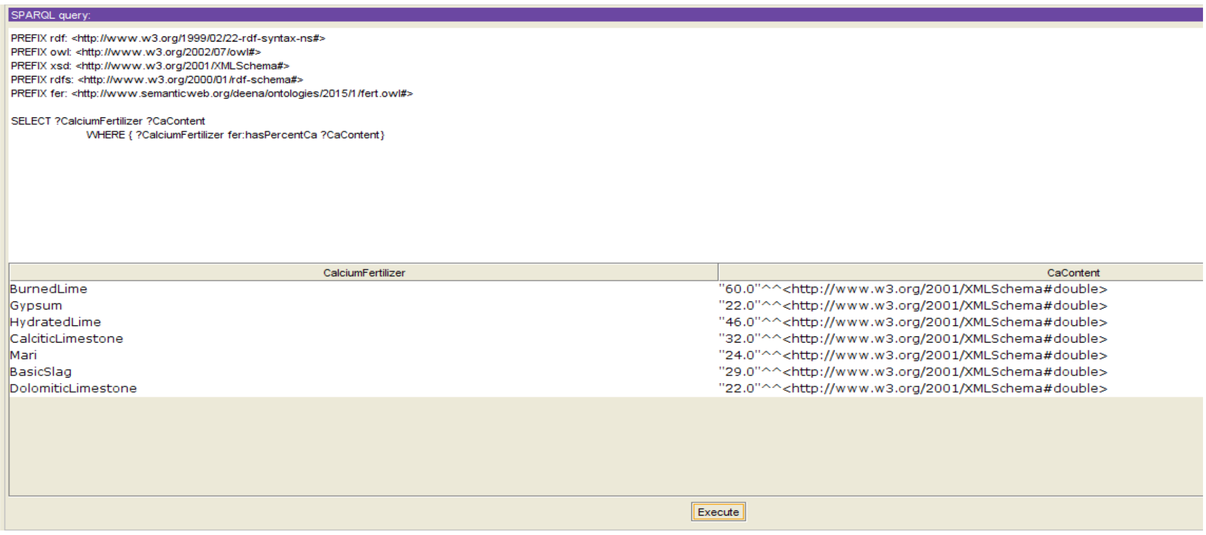


Figure 4.5: Screenshot of result of CQ1

- CQ2: Which chemical fertilizers can be mixed physically and used as mixed fertilizers?

SPARQL Query: SELECT DISTINCT ?ChemicalFertilizer ?canBeMixedWith WHERE { ?ChemicalFertilizer fer:canBeMixedWith ?canBeMixedWith }

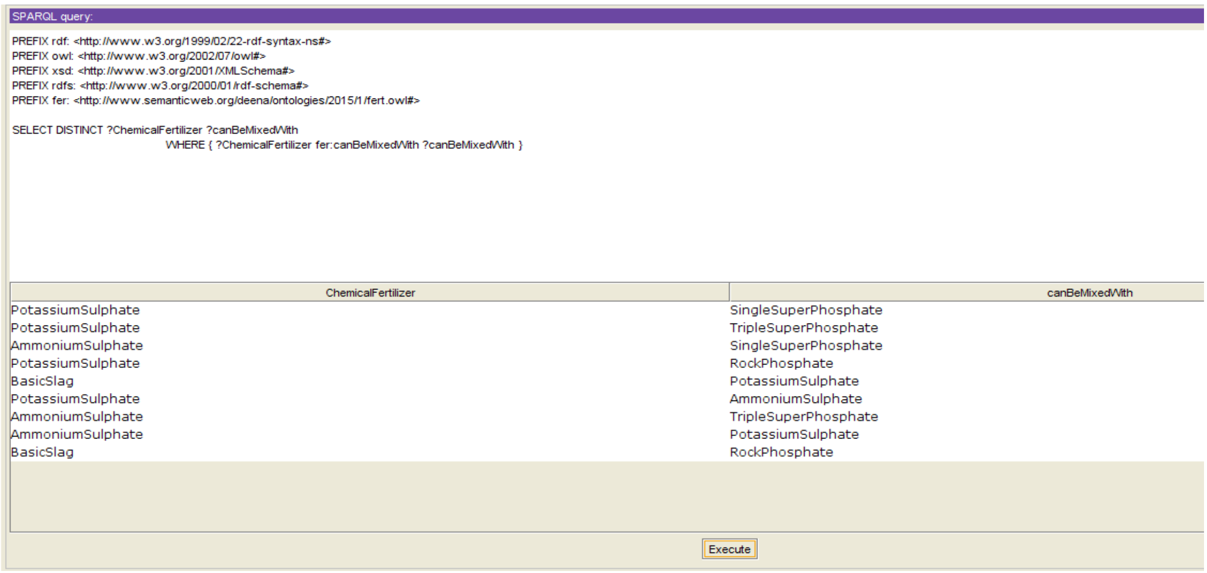


Figure 4.6: Screenshot of result of CQ2

4.2.2 Querying the inferred data using DL query

- CQ3: Which nitrogenous fertilizers use Topdressing type of fertilizer application method?

  DL Query: NitrogenousFertilizer and hasApplicationMethod value Topdressing

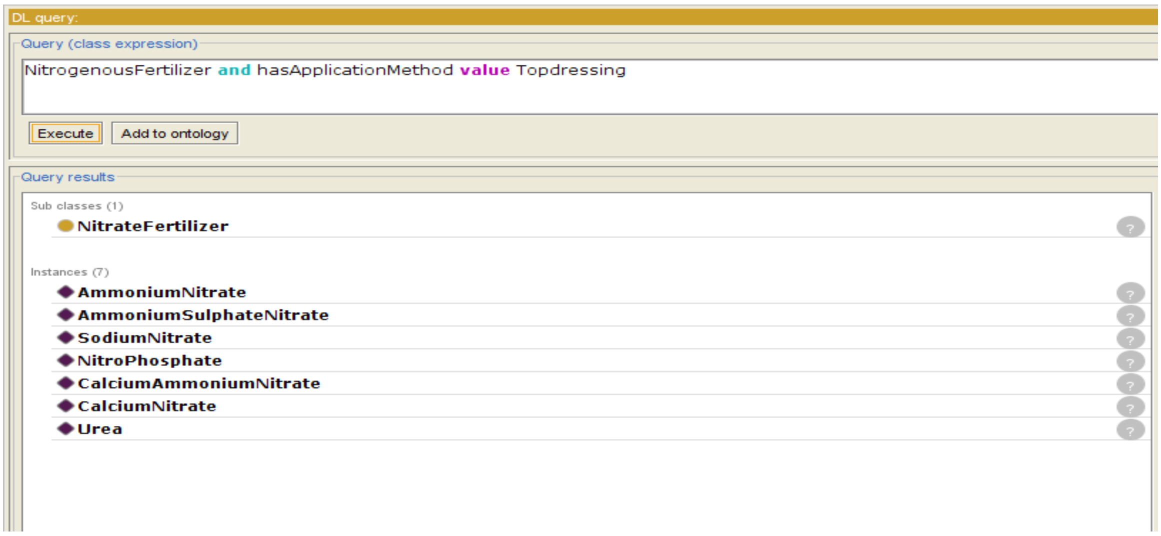


Figure 4.7: Screenshot of result of CQ3

- CQ4: List the phosphatic fertilizers that contain phosphorus in the form of dicalcium phosphate.

  DL Query: PhosphaticFertilizer and containsPInTheFormOf value DicalciumPhosphateForm

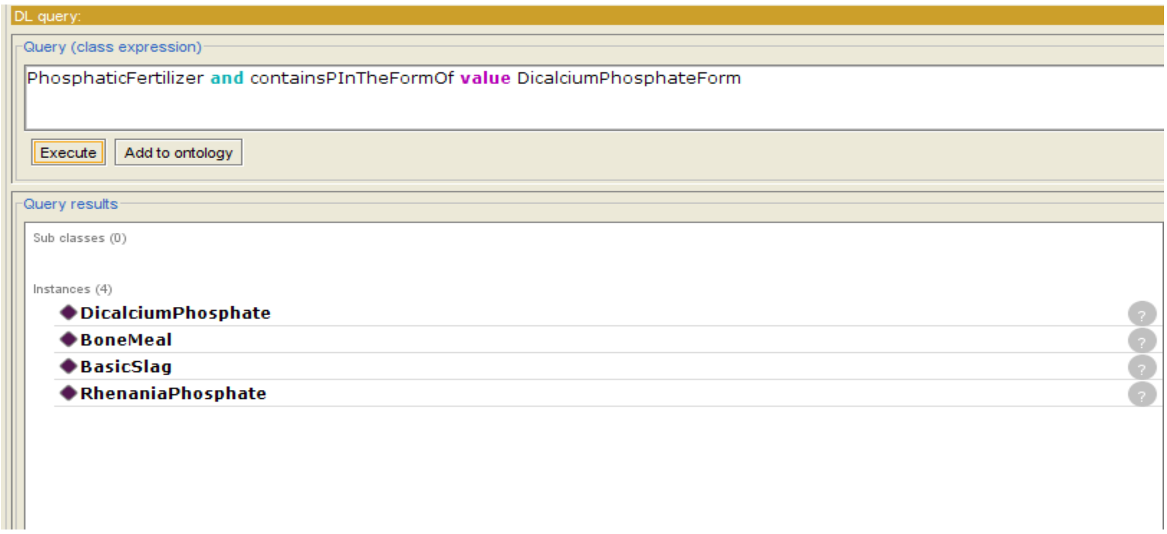


Figure 4.8: Screenshot of result of CQ4

➢ CQ5: Which phosphatic fertilizers are suitable for acidic type of soil?

DL Query: PhosphaticFertilizer and isSuitableForSoilNature value Acidic

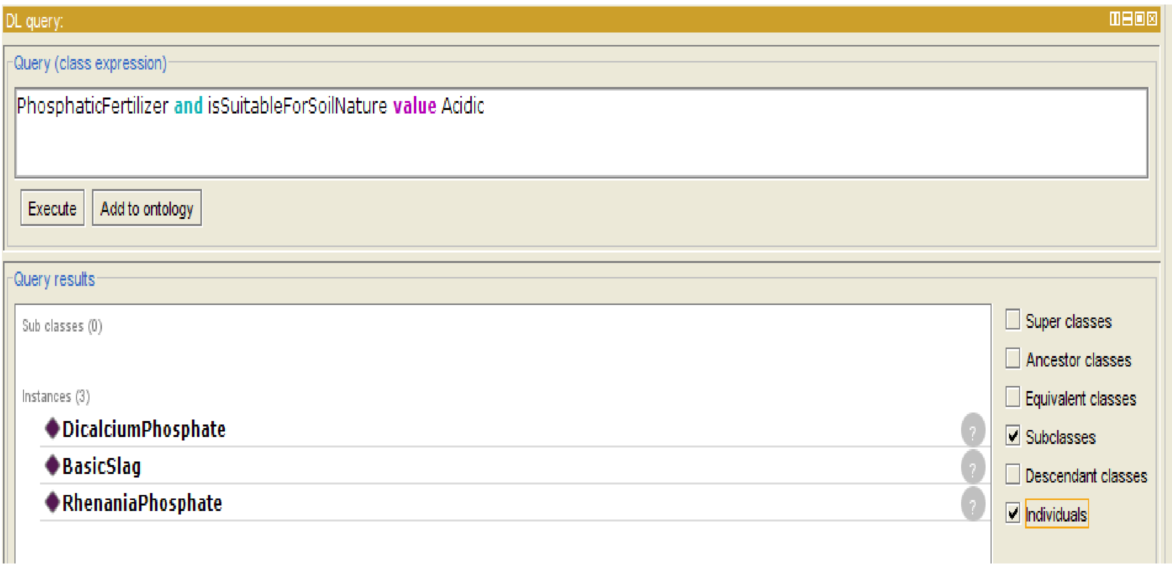


Figure 4.9: Screenshot of result of CQ5

Chapter 5

Conclusion & Future Work

Ontology development in the domain of agriculture has been catching attention of a lot of researchers for quite a long time now. However, the sub-domain of fertilizer has been poorly explored. It has always been studied in relation to other entities such as soil and crop, where as it is significant to study it as a separate entity. We have tried to fill this research gap by developing ontology in the sub-domain of fertilizer. Taking into considerations the common challenges one faces while developing a domain ontology in agriculture domain, we have developed FertOnt.

In this study, we have identified 90 concepts, 25 object properties and 36 data properties. The methodology that we have followed to develop this ontology is a generalized phase by phase procedure of the activities that are required for ontology development. Much help is also taken from the Ontology Development 101 (Noy and McGuiness 2006) for the small details in the ontology conceptualization phase. For the validation of our ontology, we have made use of the OOPS! tool which is a web-based tool and validated our ontology against it. A reasoner is used to reason our ontology to show how knowledge can be inferred from asserted data. We have also answered some of the competency questions by querying on the knowledge base based on our ontology using SPARQL and DL query languages.

The future work would be to add more information into the ontology. Since ontology development is an iterative process, the ontology can be updated continuously. Also, since it is important to add dynamic information, this ontology can be extended by adding dynamic information such as market prices, consumer behaviour, information about places to buy and sell fertilizer products, etc. Also, it is possible that fertilizer ontology can be merged with other ontologies such as soil and crop ontologies. And such ontlogies can be studied to answer more specific questions such as the application of fertilizer with specified soil type or crop type.

References

Agropedia (2014, September 19). Retrieved from Agropedia website: http://agropedia.iitk.ac.in/.

AGROVOC Multilingual Agricultural Thesaurus (2014, September 12). Retrieved from AGROVOC website: http://aims.fao.org/vest-registry/vocabularies/agrovoc-multilingual-agricultural-thesaurus.

Brewster, C., & O'Hara, K. (2007). Knowledge representation with ontologies: Present challenges—Future possibilities. *International Journal of Human-Computer Studies*, *65*(7), 907- 928.

Chanda, T. K., Sati, K., Robertson, C., & Arora, C. (2012). Fertilizer statistics 2011-12. *Fertilizer Association of India, New Delhi*, 440.

Davies, J., Studer, R., & Warren, P. (Eds.). (2006). *Semantic Web technologies: trends and research in ontology-based systems*. John Wiley & Sons.

Deb, C. K., Marwaha, S., Malhotra, P. K., Wahi, S. D., & Pandey, R. N. Strengthening Soil Taxonomy Ontology Software for Description and Classification of USDA Soil Taxonomy up to Soil Series.

Fernández-López, M., Gómez-Pérez, A., & Juristo, N. (1997). Methontology: from ontological art towards ontological engineering.

Fernández-López, M. (1999). Overview of methodologies for building ontologies.

Fertilizer use by crop in India, 2005. Published by Food and Agricultural Organization of United Nations (FAO), Rome.

Fisseha, F. (2002). Towards better Semantic Standards for Information Management AGROVOC and the Agricultural Ontology Service (AOS). *UN FAO, Rome, Italy*.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, *5*(2), 199-220.

Grüninger, M., & Fox, M. S. (1995). The role of competency questions in enterprise engineering. In *Benchmarking—Theory and Practice* (pp. 22-31). Springer US.

Gupta, P. K. (2003). *Handbook of soil, fertilizer and manure*. Agrobios (India).

HE, H. Y., & YUAN, C. F. (2005). Overview of Technology of Building Domain Knowledge Based on Ontology [J]. *Application Research of Computers*, *3*(3).

Horridge, M., Knublauch, H., Rector, A., Stevens, R., & Wroe, C. (2004). A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools Edition 1.0. *University of Manchester*.

Jaiswal, P., Avraham, S., Ilic, K., Kellogg, E. A., McCouch, S., Pujar, A., ... & Zapata, F. (2005). Plant Ontology (PO): a controlled vocabulary of plant structures and growth stages. *Comparative and functional genomics*, *6*(7-8), 388-397.

Knowledge representation and reasoning . (n.d.). In *Wikipedia*. Retrieved January 10, 2015, from https://en.wikipedia.org/wiki/Knowledge_representation_and_reasoning.

Maedche, A. 2001. Development and applications of ontologies (ECML / PKDD‘2001 Tutorial). Research Center for Information Technologies at the University of Karlsruhe, Germany.

Musen, M. A. (1992). Dimensions of knowledge sharing and reuse. *Computers and biomedical research*, *25*(5), 435-467.

Noy, N. F., & McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology.

Ogden, C. K., Richards, I. A., Malinowski, B., & Crookshank, F. G. (1923). *The meaning of meaning* (pp. 9-12). London: Kegan Paul.

Pinto, H. S., & Martins, J. P. (2004). Ontologies: How can they be built?.*Knowledge and Information Systems*, *6*(4), 441-464.

Poveda-Villalón, M., Suárez-Figueroa, M. C., & Gómez-Pérez, A. (2012). Validating ontologies with OOPS!. In *Knowledge Engineering and Knowledge Management* (pp. 267-281). Springer Berlin Heidelberg.

Protege (2000). The Protege Project. http://protege.stanford.edu

Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., & Katz, Y. (2007). Pellet: A practical owl-dl reasoner. *Web Semantics: science, services and agents on the World Wide Web*, *5*(2), 51-53.

Sirin, E., & Parsia, B. (2007, June). SPARQL-DL: SPARQL Query for OWL-DL. In *OWLED* (Vol. 258).

Takeya, M., Numa, H., & Doi, K. (2003). Ontology Using Role Concept Recognized on Biological Relationships and Its Application. *GENOME INFORMATICS SERIES*, 685-686.

Thunkijjanukij, A. (2009). Ontology development for agricultural research knowledge management: a case study for Thai rice.

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *The knowledge engineering review*, *11*(02), 93-136.

Uschold, M., & King, M. (1995). *Towards a methodology for building ontologies* (pp. 15-30). Edinburgh: Artificial Intelligence Applications Institute, University of Edinburgh.

Vincent, P. L. D., Coe, E. H., & Polacco, M. L. (2003). Zea mays ontology–a database of international terms. *Trends in plant science*, *8*(11), 517-520.

Walisadeera, A. I., Wikramanayake, G. N., & Ginige, A. (2013). An ontological approach to meet information needs of farmers in Sri Lanka. In *Computational Science and Its Applications–ICCSA 2013* (pp. 228-240). Springer Berlin Heidelberg.

Walisadeera, A. I., Wikramanayake, G. N., & Ginige, A. (2013). Designing a Farmer Centred Ontology for Social Life Network. In *DATA* (pp. 238-247).

Walisadeera, A. I., Ginige, A., & Wikramanayake, G. N. (2014). Conceptualizing Crop Life Cycle Events to Create a User Centered Ontology for Farmers. In *Computational Science and Its Applications–ICCSA 2014* (pp. 791-806). Springer International Publishing.

Publication

1. Malik, N., Sharan, A., & Hijam, D. (2015) Ontology Development for Agriculture Domain. In *Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on* (pp. 738 - 742). IEEE.