EXTENDING CLASSICAL REASONING FOR CLASSIFICATION QUERIES OVER ONTOLOGIES

Nishal Ashok Morar

Submitted in fulfilment of the academic
requirements for the degree of
Master of Science in the
School of Computer Science,
University of KwaZulu-Natal Durban

November 2016

Supervisor: Prof. Aurona Gerber
Co-Supervisor: Prof. Thomas Meyer

As the candidate's supervisor I have/have not approved this thesis dissertation for submission.

Name:_____ Date:_____ Signed:_____

Abstract

Ontologies are used within Knowledge Representation and Reasoning (KRR) to represent a domain of interest and to assert specific knowledge about the domain. This is done through a class hierarchy and explicit syntactic sentences called axioms, which are made up of concepts, roles and objects. Description Logics (DLs) are a group of knowledge representation languages that can be used to formulate ontologies using similar building blocks. An advantage of using DLs is their ability to support reasoning functionality over the axioms, in order to identify implicit knowledge from the explicitly stated facts. Such reasoning can be performed automatically by software inference engines called reasoners. In a similar way that an ontological concept is defined by declaring facts about the concept, an organism or taxon in taxonomy is defined by specifying all of its unique, defining characteristics. The *conceptual* process of defining a concept in an ontology, and defining a taxon is similar, thus ontologies can be used to model a taxonomy, and classification can be performed through DL queries.

Taxonomy is the scientific classification, description and grouping of certain objects or organisms, and the principles that enforce such classification. One of the goals of taxonomists is the ability to communicate their work, which is normally done through taxonomic keys that are used to identify organisms, and are usually text based. When identifying and grouping objects, certain questions arise such as *'which objects exist that have various identified unique features?'* and the *reverse* of the mentioned question, when dealing with the taxonomic process of taxonomic revisions, *'what features does each (speculated) object possess, and which are the common shared features between them?'* When asking the second question as a query over an ontology, acquiring the needed results proves difficult when using the standard reasoning services. Ways to perform the query through the remodelling of the ontology exist, but are cumbersome and time consuming if dealing with a large ontology. In this dissertation, an alternate way to solve such a query through the use of an existential reasoning algorithm that utilises and extends the standard reasoning services thus avoiding the redundant remodelling, is presented. It is illustrated in a practical way using an ontology and a web ontology based classification application, both which are developed as part of this research study. The ontology and application together function as a computerised taxonomic tool for a specific case study of Afrotropical bees, though they can be applied and used in other domains.

Preface

The experimental work described in this dissertation was carried out in the School of Computer Science, University of KwaZulu-Natal, Durban, from January 2014 to June 2016, under the supervision of Professor Aurona Gerber and Professor Tommie Meyer.

These studies represent original work by the author and have not otherwise been submitted in any form for any degree or diploma to any tertiary institution. Where use has been made of the work of others it is duly acknowledged in the text.

Declaration 1 - Plagiarism

I,_____, declare that:

1. The research reported in this dissertation, except where otherwise indicated, is my original research.
2. This dissertation has not been submitted for any degree or examination at any other university.
3. This dissertation does not contain other person's data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This dissertation does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
   (a) Their words have been re-written but the general information attributed to them has been referenced.
   (b) Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.
5. This dissertation does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the dissertation and in the References section.

Signed: _____

Declaration 2 - Publications

Gerber, A., Eardley, C., and Morar, N. An Ontology-based Taxonomic Key for Afrotropical Bees. *In the International Conference on Formal Ontology in Information Systems,* pages 277-288. 2014.

Contributions of Authors:

A. Gerber contributed to the ontology construction and writing of the document. N. Morar contributed to the application development and implementation work involved in the document. C. Eardley provided the necessary data used in the application and contributed to the theoretical aspects of the document.

Signed: _____

Acknowledgments

Firstly, I wish to express my very great appreciation to my supervisor Prof. Aurona Gerber and co-supervisor Prof. Thomas Meyer for their constant support, guidance and invaluable advice throughout this research. I am very grateful to Dr. Connal Eardley for his drive and assistance with this project, as without him most of this would not have been possible. I would also like to thank the members of the Knowledge Representation and Reasoning (KRR) group at CSIR Meraka for their help and encouragement. A special mention goes to Kody Moodley who provided a lot of help by answering my many questions about description logics. Thank you also to Marlene Jivan and Laila Gurudas for all their help with the various administrative tasks that had to be done with this Masters degree.

A special thank you goes to Dr. Matthew Horridge of Stanford University who provided important advice and access to his reasoning algorithm, which played a big part of the practical component of my research.

Many thanks to the Centre for Artificial Intelligence Research (CAIR) for awarding me the funding to conduct the research and last, but not least, a special appreciation goes to my family and close friends for providing me with their support throughout my studies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Ontologies, Description Logics and Reasoning

Within computing, ontologies are formal representations of a domain of interest [Gruber, 1993], that are able to be interpreted and processed by computers. Ontologies are comprised of concepts, roles and objects, which are used in syntactic sentences termed *axioms*, to describe the particular domain. The axioms consist of assertions made about the concepts, as well as the relations between the concepts via roles.

Ontologies, within computing, are often modelled and represented using a group of knowledge representation languages known as Description logics (DLs) [Krötzsch et al., 2012]. DLs are used to portray information about some domain [Baader et al., 2003] and are made up of atomic concepts, atomic roles and individuals or instances as their basic building blocks. DLs are equipped with a formal logic based semantics, since they are formulated as decidable subsets of first-order logic [Kleene, 2002]. Essentially, this means that the ambiguity within the meanings of terms and sentences that are expressed as DLs, is eliminated. One of the main advantages of DLs is this ability to *reason* over an ontology or a domain, and in turn reduce the ambiguity within the meanings of terms [Krötzsch et al., 2012]. Through this reasoning, additional *implicit* knowledge can be derived and made explicit from the explicitly stated definitions and relationships modelled in the axioms of a DL ontology.

A simple example of the concept of reasoning is provided next. Given two explicitly stated facts, *"Sam is a father"* and *"fathers have children"*, the implicit inference can be drawn that *"Sam has children"*. This process is performed automatically by software inference engines known as DL reasoners, which execute several different reasoning tasks. These reasoning tasks include concept satisfiability, consistency checking, subsumption checking and instance checking. These standard services have been implemented in most of the DL reasoners available, to cater for the many distinct DLs. Subsumption testing for instance, is the procedure of checking whether a concept is a sub-concept or a super-concept of another, and is based on the is-a relation, which is widely used in many different domains [Brachman, 1983]. A simple everyday example is a dog or cat, which can be represented as *a type of* pet.

Within this research study, ontologies and DL reasoning are applied to the field of taxonomy. Ontologies are used to store morphological taxonomic knowledge and DL reasoning is used in the application and classification of the data. Subsumption testing is predominantly

used along with the process of *classifying* the ontology, which formulates and identifies the sub-concepts and super-concepts for all the concept names in the ontology, to ultimately produce a *concept hierarchy*. The standard DL reasoning is unable to solve a specific classification query, and thus a few solutions are investigated, including the use of a non standard existential reasoning algorithm.

1.2 Taxonomy and Afrotropical bees

Taxonomy, within biodiversity, can be seen as the scientific classification, description and grouping of certain objects or organisms, and the principles that enforce such classification [Guerra-García et al., 2008]. The organisms or *taxa* are grouped together according to a number of shared and distinct features, which generally are morphological characteristics. Such characteristics are made up of the organism's body parts and their associated morphological descriptions, for example the *"Metasoma"* (which is part of the abdomen of an insect) could have *"medially dense punctation"* (indentations).

Experts in taxonomy, or taxonomists, play a vital role in the field of biodiversity and within the world. They have two main roles, the first being the identification and description of taxa, and secondly the identification and establishment of new taxa to science. Once an organism or a certain group is identified the next undertaking is to establish how it can be distinguished from other taxa or other groups, and what are its unique characteristics. This thorough process forms part of the practice termed taxonomic revision, which involves the description, identification and/or revision of new taxa [Maxted, 1992]. The taxonomic revision procedure is quite a substantial and tedious process that involves many long uninterrupted hours of work on the taxonomist's part.

Worldwide there are relatively few taxonomists with very slow growth in the area and thus a shortage of such taxonomic skills exist [Stork, 1993]. Taxonomists play an essential role in biology for many reasons, amongst others, they are capable of making predictions about described taxa and they can identify exotic pests and disease organisms. Taxonomists are also able to offer expertise to other areas within biodiversity such as determining behavioural properties and patterns of species and their interactions with ecology, which could prove crucial for the continued use of the earth's natural resources [Hoagland, 1996]. The technological state of taxonomy is concerning since a large part of the activities in the field are still paper based and the data is still in the process of being digitally stored [Godfray, 2002]. Most of the data that is 'computerised' is often stored in simple text based formats that cannot be easily used and accessed. Similar to moving to more digital formats for data storage, the use of technological applications in the field could lead to greater efficiency and effectiveness, especially for a taxonomist.

A specific practice that taxonomists use for dissemination are taxonomic keys. Taxonomic keys are used to help identify organisms or taxa using certain unique key diagnostic features. Such keys are created and used by taxonomists amongst other users, although many of them are text based keys that allow no computer aided support [Eardley et al., 2010].

Afrotropical bees, as the name suggests, are found in Sub-Saharan Africa, which is the region south of the Sahara desert. Within these ecosystems they are one of the largest pollinator groups that enable the successful growth of many different crop variations via pollination. The bees play a vital role within agriculture and in maintaining biological diversity, and without them the world's food security could become compromised [Eardley, 2002]. Due to the key function of Afrotropical bees in biodiversity, it is regarded as crucial that knowledge about the behavioural patterns and characteristics of Afrotropical bees is maintained and increased to support ecological research, in order to continually understand their role in natural ecosystems [Eardley, 2002; Conte and Navajas, 2008].

Taxonomy includes the classification of various taxa that are placed into defined groups [Guerra-García et al., 2008]. A taxon is defined by specifying all of its unique, defining characteristics. Similarly, a concept in an ontology is defined and grouped by declaring facts or characteristics about the concept, so that every element that belongs to the concept inherits those characteristics [Gruber, 1993]. Thus the *conceptual* process of defining a concept in an ontology, and defining a taxon is similar [Franz and Thau, 2010; Schulz et al., 2008]. Ontologies can be used to model a taxonomy, and classification can be performed through queries using the ontology and DL reasoning.

1.3 Classification and Ontologies

Classification is the general method of grouping certain objects together, according to one or more distinct characteristics that the objects possess [Merriam-Webster Dictionary; Daniel, 2016]. Classification and taxonomy are associated, since the classification procedure is commonly done to develop the taxonomy of organisms. Organisms can then be identified using the unique characteristics.

One of the most difficult aspects of taxonomic classification is recognising the various features of an organism that distinguish the taxa, which can take years of practice as a taxonomist [Culverhouse et al., 2003; Gaston and O'Neill, 2004]. When attempting to identify an organism (or any object) certain questions or queries come to mind. The queries are used over specific pieces of data to eventually retrieve the desired result or organism. Essentially one of the questions could be *'which species* (or objects, in a general situation) *exist that have these identified unique features?'* (taxonomic keys are typically used for such questions). Another possible question could be the *reverse* of the first one, *'what key features does this (speculated) species possess?'* or, with many specimens being examined, *'what specific features does each species possess, and which are the common shared features between them?'* Such classification questions can typically be asked over an ontology as DL queries. The queries are carried out by DL reasoners that are capable of performing the standard reasoning tasks over an ontology.

Ontologies consist of concepts that are defined and associated with other concepts via roles, resulting in syntactic sentences or *'axioms'*. Given an ontology of only concepts and roles (and no individuals), a query to solve the first question (*'which species or objects exist that have*

these identified unique features?') is possible utilising the standard reasoning services, such as subsumption testing, which are available to draw inferences from the standardised ontology languages, such as OWL. On the other hand, the second question (*'what key features does this (speculated) species possess?'*) is not so simple and cannot be solved using the standard reasoning services. A few solutions that involve modifying the ontology to use subsumption testing and instance checking are investigated, but when dealing with large ontologies these solutions are not the most favourable.

1.4 Research questions

1.4.1 Motivation

In this dissertation a case study of Afrotropical bees is presented as well as research on how to use morphological data about these bees to model an 'Afrotropical bee ontology'. An investigation into how particular classification queries can be answered over an ontology forms the basis of the study. Questions such as *'what key features does a particular (speculated) species possess?'* or *'what are the individual as well as common shared features of multiple species?'* are taken into account and posed as queries over the Afrotropical bee ontology. The queries though, cannot be solved through standard reasoning alone, thus a solution to enable such queries is investigated. Also part of the investigation is how an ontology-based multi-access taxonomic key application can be developed for Afrotropical bees that assists taxonomists with a taxonomic procedure taxonomic revisions.

1.4.2 Research Questions

The main research question that is addressed by this study is:

- How can classical reasoning be extended for classification queries over ontologies?

The main question is then divided into three sub-questions, which are:

1. How can Afrotropical bee morphological data be modelled into an ontology for classification purposes?
2. How can reverse classification queries over concepts be executed over an ontology given the case of an ontology for Afrotropical bee morphological data?
3. How can a multi-access taxonomic key application be developed for Afrotropical bees that assist taxonomists with taxonomic revisions?

By addressing all three sub-questions, the main research question is answered.

1.4.3 Research Design

The research questions are answered using an experiment [Collins et al., 2004]. The first phase will consist of the collection of the requirements for features that support taxonomy, from the domain experts. Next, the modelling of taxonomic knowledge, particularly Afrotropical bee taxonomic knowledge, in an ontology will be investigated to enable and suit classification queries. Certain classification questions are usually asked when classifying organisms and

when performing the taxonomic procedure of taxonomic revisions, which can be translated into DL queries. When the questions dealing with taxonomic revisions are applied to a typical key, the *reverse* functionality of a key is the needed outcome i.e. to identify the key diagnostic features of a given Afrotropical bee species (or genera), or of multiple species, and enable access to more information such as the shared features of the species. The queries will be formulated and executed over the ontology to determine if the relevant knowledge can be retrieved from the ontology. Where the queries are unable to be performed using the standard reasoning services, suitable solutions involving the remodelling of the ontology as well as the extension of the standard reasoning will be investigated, tested and presented. The final phase of the experiment will be the development of an ontology-based web application that supports taxonomy and its processes of classification and taxonomic revisions. Such an application or tool could thus assist with taxonomic revisions because the common diagnostic features of a given taxa could be analysed. Taxonomic revisions are discussed in more detail in Section 2.1.3. The application will aim to function as an interface for the ontology and incorporate the classification queries, to allow for user friendly interactive information retrieval. The application will then be evaluated using the following mechanisms:

1. An investigation into whether the extension of the standard reasoning rendered the required results through the queries over the ontology.

2. A detailed usage evaluation by a taxonomist and feedback from other taxonomists and potential users of the application.

1.5 Organisation of the Dissertation

The dissertation is structured as follows:

- Chapter 2 introduces the application domain, which scopes the study. The case study discusses biodiversity, taxonomy and Afrotropical bees, mentioning the various reasons why these fields need attention. The types of taxonomic keys are also introduced and discussed.

- Chapter 3 provides background on ontologies, DLs and the *reasoning* that DLs provide for. The syntax and semantics of the basic DL $\mathcal{ALC}$ is covered to give insight into the DL terminology that is used in many instances throughout the dissertation. The various reasoning services that are implemented in many standard reasoners are also presented.

- Chapter 4 provides the main contribution of the research study. It covers classification and the types of *classification questions* and queries that are typically asked in classification tools such as taxonomic keys. The standard reasoning services proved insufficient in answering some of the queries, and thus a few solutions (involving the remodelling of the ontology) are given. Though they provided the correct results, the methods were time consuming, hence a more convenient solution of using an additional reasoning algorithm is explained.

- Chapter 5 discusses two sub-contributions of the research. Firstly the Afrotropical bee ontology is described and the ontology's modelling of the Afrotropical bee's morphological

data is presented. Then the web ontology classifier application is discussed, which was developed to function as a multi-access key to aid bee taxonomists with identifying and classifying species. Through a few examples the application is shown to incorporate an existential reasoning algorithm to cater for the *reverse* classification queries over concepts, with the goal of aiding taxonomic revisions.

- Chapter 6 presents an evaluation of the ontology classifier application by expert taxonomists through a questionnaire, in order to deduce how effective the tool could be to taxonomists, and to the field of taxonomy as a whole. A bee taxonomist, Dr. Eardley, also evaluates the tool and the whole approach of the project.

- Finally, Chapter 7 summarises the various findings and contributions of the research and identifies future work following from this research.

Chapter 2

Scoping of the Study

Introduction

In this chapter, the case study that is applicable to this specific research study is presented in order to provide the scope. The project was initiated through a collaboration with an acknowledged bee taxonomist, Dr. Eardley from the Agricultural Research Council. Through the collaboration, the requirements for an ontology-based expert system was elicited. Further requirements highlighted the need for expert tools to assist with taxonomic revisions. This chapter provides some background on taxonomy and biodiversity, the impact that these fields have as well as the challenges that are faced within the fields.

The first section covers the background of taxonomy in general and all the different aspects of it. The role of the taxonomist and the challenges taxonomists face are also discussed. In addition, the different types of taxonomic keys are highlighted and a brief analysis of an example of an existing key application is presented, as well as a discussion on what taxonomic revision entails.

The second section provides an overview of Afrotropical bees, which this case study is based upon.

This section concludes with a discussion about the impact that biodiversity has on the world's ecosystems as well as some of the world-wide concerns that could arise due to the insufficient investment in taxonomy.

2.1 Taxonomy

In this section an introduction and background to the field of taxonomy and species taxonomy is given. The section looks at the particular standards and rules that are in place for the naming of species as well as the roles of the taxonomy experts who are responsible for naming and identifying species. Taxonomic keys are subsequently discussed. Lastly the process termed *taxonomic revisions*, which is a significant part of a taxonomist's job, is explored.

Taxonomy, broadly speaking, is the scientific classification, description and grouping of objects or concepts, as well as the principles that motivate such classification [Ereshefsky, 2000]. In

biology it is the science of the description and classification of organisms into taxa [Guerra-García et al., 2008; Ereshefsky, 2000]. These taxa are grouped and the groups are named according to shared characteristics or features. The key features will have been formulated to specifically describe a set, or a specific taxon, and these features are morphological characteristics that are made up of the organism's body parts and their associated descriptions. For example, the male's basitarsus (which is a specific part of an insects' leg) could have a particular colour such as red, black or orange. Each taxon group is assigned a taxonomic rank in order to organise them and thus form a taxonomic hierarchy [International Code of Zoological Nomenclature]. Kingdom, Phylum, Class, Order, Family, Genus and Species are all taxonomic ranks, and this hierarchical system, which is still used today, is based upon the original system of Linnaeus [de Queiroz, 1997; Ereshefsky, 2000].

Carl Linnaeus is considered to be the father of modern biological taxonomy [de Queiroz, 1997]. He developed a basic universal classification system to uniquely classify all living things within a hierarchy [CBD: Linnaeus Lecture Series]. Some of his main contributions to society include being the first person to use binomial nomenclature consistently, and through his books he provided taxonomic keys to assist people in identifying plants and animals. Linnaeus, through his book, the 10^{th} edition of Systema Naturae [Linnaeus, 1758], gave rise to the modern biological naming scheme of binomial nomenclature [Griffiths, 1976]. Nomenclature encompasses the system of scientific naming of taxa (e.g. species or genera). Nomenclature establishes standards, rules and guidelines that all naming of biological organisms should adhere to, to enable standard conventions to be followed when assigning names to specific taxa [International Code of Zoological Nomenclature]. Through this naming process certain problematic situations such as duplicate names can be avoided. Each field of organisms is governed by a book or code of nomenclature for the naming of the respective group of organisms, such as the International Code of Nomenclature for algae, fungi, and plants, (ICN) or the International Code of Zoological Nomenclature (ICZN) [International Code of Zoological Nomenclature].

There are three typical phases for activities in taxonomy [Kapoor, 1998; Disney, 2000]. Firstly alpha taxonomy, is the initial phase and comprises of the species, subspecies and other taxa being identified and described. Beta taxonomy has to do with the grouping of the species into a natural system of lower and higher categories. Thirdly, gamma taxonomy deals with the analysis of intra specific variations, ecotypes, polymorphisms and evolutionary rates and trends.

Taxonomists have two core roles within taxonomy, to *identify* and *describe* species or taxa, and the identification and naming of new species [Culverhouse et al., 2003]. Taxonomists are also responsible for naming and grouping certain species into their respective taxa, which represent groups of organisms. This grouping enables better organisation and classification. Once a taxon is identified the next undertaking is to establish how it can be distinguished from the other groups, and what are its unique characteristics. This thorough process forms part of the practice termed taxonomic revision, which is discussed later in section 2.1.3.

Worldwide there are relatively few taxonomists and a shortage of taxonomic skills exist [Stork, 1993]. The field of taxonomy is under pressure as many experienced taxonomists are retiring and are not being replaced. A reason for the taxonomic skills not being replaced could be a lack of interest and funding in the field [Guerra-García et al., 2008; Wheeler et al., 2004]. Becoming an expert taxonomist does take time and years of experience studying the particular discipline is needed to refine taxonomic skills.

Taxonomists' roles extend beyond taxonomy into ecology and other areas impacting the world. Their responsibilities can extend, to making predictions about the described taxa, identifying exotic pests and disease organisms, as well as to offer insight to other areas within biodiversity, such as determining behavioural properties and patterns of species and their interactions with ecology [Hoagland, 1996]. This taxonomic information is relevant and can be helpful to ecologists and management authorities in understanding species distributions, species interactions and ecosystem structure and rank, the justification of conservation areas, as well as planning restoration efforts [Hoagland, 1996]. Experienced taxonomists who have an in depth knowledge or expertise in a group of organisms are required to accomplish these tasks.

The tasks of a taxonomist are often quite complex and may require a great deal of effort, patience, long continuous hours and having to utilise poor data sources such as museum guides, which can be up to 200 years old [Godfray, 2002]. Older specimens need to be found to be compared to current samples, which can be time consuming in itself. The process of finding and establishing a new group of taxa, to finalising the printed name can take a few years and may require a number of revisions. Taxonomists are often subjected to work with outdated and unstructured research notes, since much of taxonomy is still paper based [Godfray, 2002]. The data is in the process of being stored on computers and most of the existing digital data are stored in formats that cannot be easily used and ported to useful applications. Since there is a shortage of taxonomists it is not uncommon for the few taxonomists to be burdened with additional work as well as tasks that involve computerising large amounts of specimen records going back almost 200 years [Hoagland, 1996]. When performing classification and taxonomic revisions, taxonomists rely mostly on their memories, informal research notes and documented keys.

With the advent growth of technology and computer based applications, many aspects of the world have embraced this opportunity to aid the particular field. Taxonomy though, is still dependant on manual labour and computers essentially only replaced typewriters in taxonomy, and thus computing applications or technologies are rarely used for computing or classifying taxonomic data and knowledge [Godfray, 2002]. Although, through the Internet a lot of taxonomic data is now digitally stored and is somewhat available online, but there is still paper based taxonomic data in museums of unstudied specimens and undescribed new species that need to be attended to [Hoagland, 1996]. Even though there has been numerous discussions and ideas to *'computerise'* aspects of taxonomy, little has actually been applied [Gaston and O'Neill, 2004; Patterson et al., 2010; Pennisi, 2012; Godfray et al., 2007], and thus there are very few advanced computer based applications or tools to assist with taxonomic procedures. Many taxonomists still manually write pages of notes, especially when

out in the field collecting samples for example. They may then at a later stage translate some of those notes into a more formalised document on a computer, having to schedule time specifically for such a task. Taxonomists still make use of desktop applications such as Microsoft (MS) Word documents and MS Excel spreadsheets as a form of storing their data. To input and manipulate large amounts of data in such applications, can become quite taxing and challenging.

Taxonomic information needs to be maintained and made more readily available and can be done through the use of computer based taxonomic applications or over the Web [Godfray, 2002]. Applications could also provide a way to document and preserve the large amounts of data that have been collected over several years. As taxonomists retire, few are replaced, and most of their unpublished information and work can disappear and may never be used [Boero, 2001]. Thus, through taxonomic applications the data can then be accessed and used by future taxonomists, scientists of other disciplines like ecologists, and even the general public. Continuation and addition to the data will be made easier when it is readily available and accessible in a standardised format.

As part of this research study an ontology was developed (to represent specifically Afrotropical bee taxonomic data), as well as an application that makes use of the ontology to classify the bee species, functioning as an *Ontology Based Taxonomic Key* for Afrotropical bees. Discussed next is taxonomic keys and the different types that exist to disseminate taxonomic data.

2.1.1 Taxonomic Keys

Taxonomic keys are tools that biologists use to disseminate the taxonomic data and they can be used to identify organisms or taxa using unique key diagnostic features [Walter and Winterton, 2007]. These features, determined by taxonomists, represent the set of characteristics that can uniquely describe a taxa or taxon. The keys display the diagnostic features as choices to the user, and through the choices, the user navigates the set of diagnostic features until a specific taxon is identified. Two types of keys exist, namely multi-access keys and single access keys. Single access keys are divided into a further two types, dichotomous, if it has 2 choices and polytomous, if it has more than 2 choices [Guerra-García et al., 2008; Walter and Winterton, 2007]. Essentially a single access key is an identification key, where the user is presented with two or more options that describe a key feature of the taxon. The user then, based on the sight of the specimen, chooses the appropriate feature and is then directed to the next set of features. The process is repeated until eventually the organism is identified. These keys are somewhat fixed by their creator who defines the succession of the identification steps and creates the drawback of the user becoming stuck during the decision process, if one of the features is not identifiable (due to the specimen being damaged for example). A simple example of a single access key[1] is shown in Figure 2.1.

Conversely, multi-access keys are identification keys where the user can choose where to enter with whichever feature(s) he/she has available. The user can select one, or multiple key

[1]Figure used from http://biology-igcse.weebly.com/dichotomous-keys.html

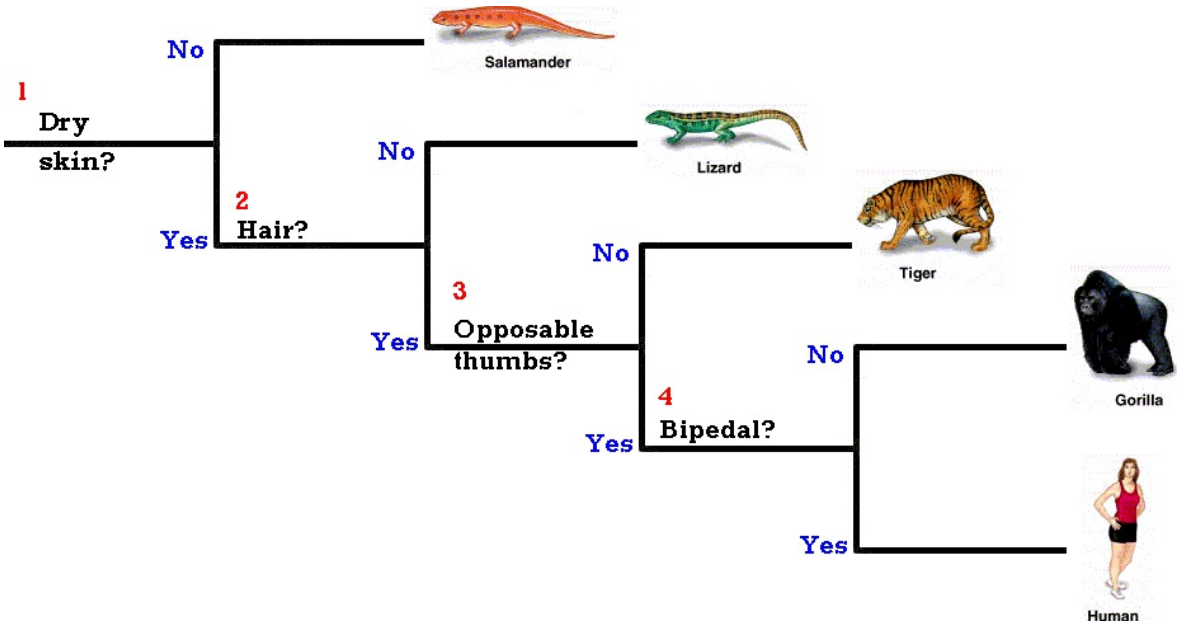


Figure 2.1: A simple example of a single access (dichotomous) key.

features from a list, with the goal of returning the set of specimens that satisfy such selection. Through refining the chosen list, any specimens not common to the set of features become eliminated until the specimen on hand is identified. This allows much more freedom for the user to choose the features that he/she can see directly and refine their search as they go on. A simple example of a multi-access key[2] is shown in Figure 2.2 and in Figure 2.4.

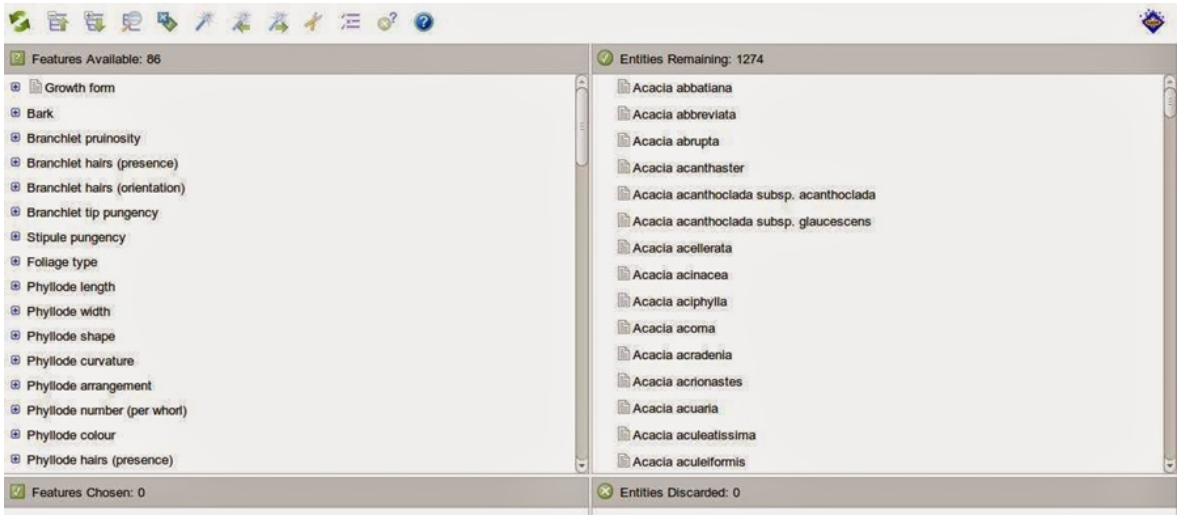


Figure 2.2: A simple example of a multi-access key.

Taxonomists are concerned with creating such taxonomic keys, which they themselves will often use. Figure 2.3 shows an example of a single access key for a Southern African bee species as a document or in 'text' format taken from Eardley [2012]. Such a document is the typical structure and format in which these keys are developed. As a text based key there is

[2]Figure used from http://phylobotanist.blogspot.com/2014/09/multi-access-keys-especially-of-lucid.html

no interactivity and thus the user has to do some work in order to traverse through the key to eventually identify a specimen.

Male. Length 9.3–11.1 mm; integument usually black, T6–T7 orange in *M. seewaldi*; clypeus gently convex, short, clypeo-ocellus:clypeus 1:0.7; antennal flagellum convex ventrally; vertex short, orbit–occiput:orbit–ocellus 1:10; orbit-ocellus:gena length 1:1.0; mandible with three similar teeth, a large tubercle medially on ventral edge; forecoxa spinose; tarsi unmodified, without large fringes; T3–T5 with distinct preapical ridges.

Identification key

1.	Female clypeus distinctly modified, ventral edge concave mediolaterally, pointed or tuberculate medially; male unknown	2
-	Female clypeus unmodified, ventral edge entire and horizontal	4
2.	Ventrolateral region of clypeus flat; vestiture of T1white	*Megachile trichroma* Friese.
-	Ventrolateral region of clypeus tuberculate; vestiture on T1 black or white	3
3.	Ventrolateral region of clypeus stongly tuberculate; vestiture on T1 black	*Megachile cornigera* Friese
-	Ventrolateral region of clypeus slightly convex; vestiture on T1 white	*Megachile braunsiana* Friese
4.	Metasoma clothed mostly with bright orange vestiture	5
-	Metasomal vestiture partly white	6
5.	Head and mesosoma brownish–orange	*Megachile dorsata* Smith
-	Head and mesosoma mostly black	*Megachile cognata* Smith
6.	Metasomal dorsum with incomplete white distal fringes, black vestiture mesally, white hair semi–erect	*Megachile ianthoptera* Smith
-	Metasomal with complete pallid, appressed distal fringes	7
7.	Mesosomal dorsum mostly with reddish–orange vestiture; male T6 acutely pointed posteromedially	*Megachile discolor* Smith
-	Mesososomal dorsum with whitish vestiture, if yellowish–brown metasomal distal fringes concolorous with mesosoma; male T6 concave posteromedially	8
8.	Metasoma entirely clothed with white vestiture; large species, 25 mm long	*Megachile rufoscopacea* Friese
-	Metasomal terga black anteriorly, pallid posteriorly; small to medium sized bees, less than 19 mm long	9
9.	Female T6 black; occurring in the South–western Cape Province	*Megachile serrula* **sp. n.**
-	Vestiture on female T6 at least partly white; occurring north of the Vaal River (South Africa)	10
10.	Female T6 black, male T6 white	*Megachile angulata* Smith
-	Female and male T6 yellowish to orange	*Megachile seewaldi* Strand

Figure 2.3: An example of a taxonomic single access (dichotomous) key document.

One of the core functions of taxonomists and scientists is the ability to communicate their research through identification or diagnostic techniques such as these published keys [Walter and Winterton, 2007]. Computerised keys are gaining some prominence due to their ease of use and accessibility [Walter and Winterton, 2007]. There are many software solutions and computerised taxonomic tools that amongst other features, support and encourage the publication of computerised keys[3][4], serving many different roles [Walter and Winterton, 2007]. One of the most popular packages, Lucid, is discussed next along with a brief overview of various computerised taxonomic tool initiatives.

2.1.2 Computerised Taxonomic Tools

Throughout the world there have been numerous computerised taxonomic initiatives to enable the storage of taxonomic data online. The aim of these databases are to document the millions of taxa in the world, and to give users the ability to search the databases, and thus have access to the taxonomic information. By enabling online taxonomic databases, other avenues can be pursued more efficiently such as checking inconsistency in taxonomy [Chavan et al., 2005].

[3] http://lis-upmc.snv.jussieu.fr/lis/?q=en/resources/software/xper3
[4] http://freedelta.sourceforge.net/

With various taxonomic databases existing, there will inevitably be some discrepancies in terms of spelling errors and in some cases naming of the data. Some species names can be identical, for example a species of wasp and a species of conifer both share the name *Agathis montana* [Page, 2005]. Through the Internet and the use of modern information and communication tools, the identification of taxonomic discrepancies can be made quicker, and they can be resolved in a collaborative manner, leading to globally acceptable standardised inventories [Chavan et al., 2005].

The Catalogue of Life (CoL) is the largest and most comprehensive catalogue of all known species of organisms on Earth [Roskov et al., 2016]. It contains upwards of 1,6 million living and 5,719 extinct species, which is still somewhat deficient for many groups, but those figures continue to rise as more information is acquired and added to it. The Col is compiled with checklists, which contain taxonomic data from extensive networks of specialists, provided by 158 taxonomic databases from around the world [Roskov et al., 2016]. Through the Col, species can be searched by name or by browsing through the hierarchical classification. Species 2000 is an autonomous federation of taxonomic database custodians, involving taxonomists throughout the world [Species 2000]. Species 2000 and the Integrated Taxonomic Information System (ITIS), though both independent, started to work together in 2001 to create the Col and also provide the taxonomic backbone to the Encyclopedia of Life (EOL)[EOL]. The ITIS database consists of more than 833,500 scientific and common names of all seven kingdoms of life (Archaea, Bacteria, Protozoa, Chromista, Fungi, Plantae, Animalia), and contains global treatments for most groups [ITIS]. All groups that are incorporated into ITIS are subjected to diverse proofing and validation criteria, as well as being assessed for both taxonomic credibility and completeness. Also integrated in the ITIS is a complete checklist of the bee species of the world.

With multiple online taxonomic databases that each have their own scope and limitations, a central search engine that uses a federated approach to query the databases would prove to be useful. One such application is the Taxonomic Search Engine (TSE) [Page, 2005]. The TSE[5] is a web application that queries multiple taxonomic databases such as ITIS, and summarises the results in a consistent format.

Lucid

Lucid is a commercial software suite of tools that is marketed as powerful and highly flexible knowledge management software applications designed to help users with specimen identification or diagnostic tasks [Lucid; Norton et al., 2012]. Lucid's design allows taxonomy experts to capture their taxonomic knowledge in a format suited for distribution via several types of media such as the Internet or CD. Lucid offers both dichotomous and multi-access keys, and can include multi-media files that make the diagnostic keys user friendly and accessible by non-expert users [Norton, 2005; Norton et al., 2012].

An example of a Lucid key for Afrotropical bee genera is shown in Figure 2.4, which is

[5] http://taxonomicsearch.sourceforge.net/index.html

available on the web[6]. In this key the user can select certain features in the top left panel, and as they are selected other features are made visible and accessible in the same panel. Simultaneously, the top right panel is updated with the matched bee genera. The data set used here was the same data set that formed the basis of our ontology, which is showcased in Section 5.1.1.

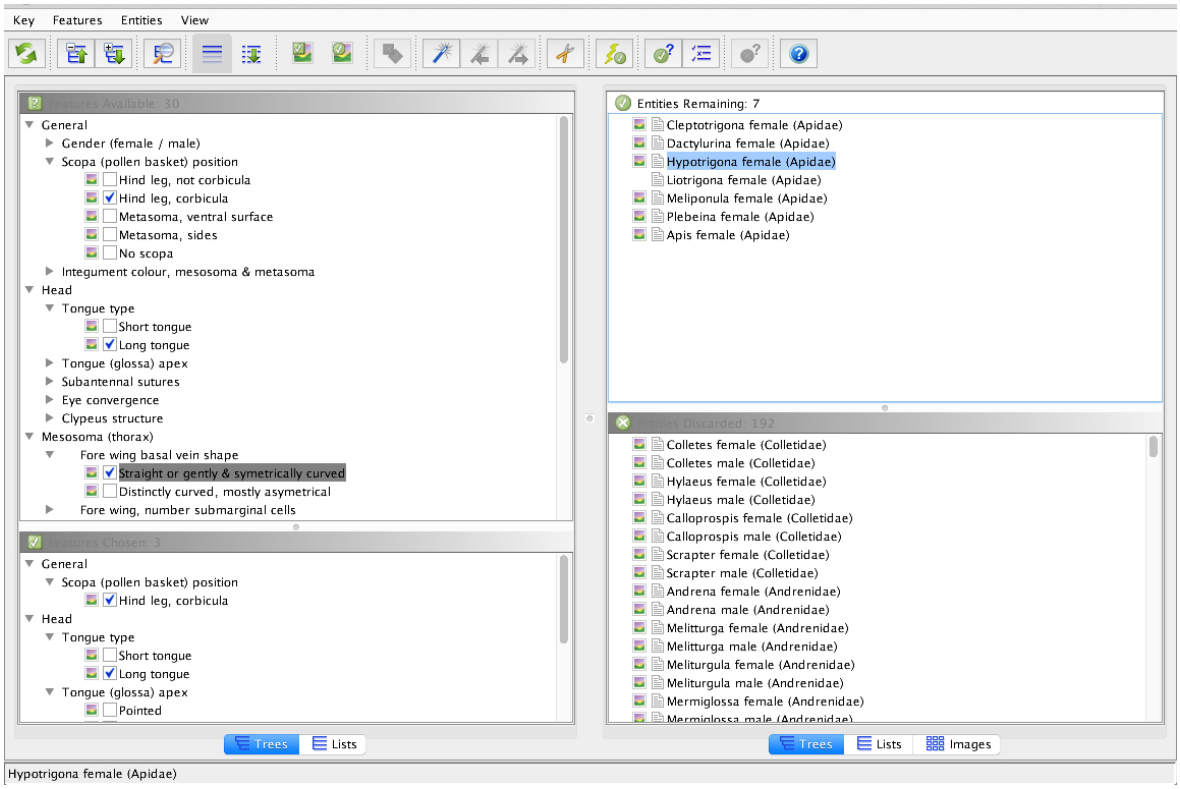


Figure 2.4: A Lucid Key available on the web.

Lucid keys are detailed keys that contain large amounts of information allowing taxonomists and users to accurately classify and identify certain taxa. The Lucid keys and databases are hard coded from taxonomists' specialised data and thus users of the keys cannot edit the keys directly. Single access Lucid keys have the drawback of the user having to follow the identification steps set out by the key creator. If at some point the user does not have access to a certain feature that is required to progress to the next step, it can be problematic and cause the user to become stuck at that point in the key. This can be due to features being damaged on a specimen for instance. Another limitation of some of the Lucid keys is that they only allow one way classification, i.e. from the key features to the species, and not the reverse direction (from the species to the key features). The ability to classify and identify the features of a given species or multiple species can be useful throughout the taxonomic revision procedure.

The process of taxonomic revisions is discussed next, including some of the challenges that are faced throughout the process.

[6] http://africanpollination.org/Africanbeegenera/Key_to_african_bee_genera.html

2.1.3 Taxonomic Revisions

Taxonomic revision is the description, identification and/or revision of new species based on a combination of morphological characteristics [Davis et al., 1963]. Taxonomic revision is a result of the discovery of new aspects and findings, and it is normally done geographically where all the species in at least a region are all studied together [Maxted, 1992; Davis et al., 1963]. This scientific field is the domain of taxonomic experts who often have many years of experience on the subject matter. Taxonomists use a variety of different morphological characteristics to describe certain species depending on the particular field [Maxted, 1992]. More specifically, bee taxonomists make use of morphological characteristics of bees such as *'integument colour'* to describe bee species. Through continued research in the field there arises a need to evaluate and revise the bee taxonomy as there is always new data and new specimens being discovered. An example of a taxonomic revision can be seen in the document by Timmermann and Kuhlmann [2009], which shows the taxonomic revision of the African bee subgenera Patellapis, Chaetalictus and Lomatalictus, and provides keys for species identification.

As discussed, a taxonomic revision is a lengthy process that usually involves many uninterrupted hours of work on the taxonomist's part due to a number of species and morphological information about those species being dealt with all at once and in memory [Maxted, 1992]. Going through documents of keys every time something is required can be time consuming and rigorous. Interruptions result in the taxonomist having to backtrack and in extreme cases start again, which is ineffective.

Knowledge of new species and the characteristics of these new species play an important role within the world [Guerra-García et al., 2008]. For example, many crops depend on the pollination of Afrotropical bees in order to be produced and a majority of crop production sustains life within many ecosystems [Sodhi et al., 2009]. Thus, if the essential need to monitor and continually keep track of these pollinators and their behavioural characteristics are underestimated, the world's food resources could be severely threatened.

Presently users of taxonomic knowledge do make use of *keys*, but most are paper based or are simply Word documents similar to the one shown in Figure 2.3. While this works it does add time to the process, which could be better utilised. These keys are also mostly *one way* keys where the user cannot use them to identify the key diagnostic features of a given Afrotropical bee species (or genera), or of multiple species. An application or tool where the reverse of a key is implemented, could thus assist with taxonomic revisions because the common diagnostic features could be analysed. A simple example would be where a taxonomist has a specimen and believes it to be a specific species (or belongs to a specific genus). An instantaneous check of what features that species has and/or what features it shares with a similar species would need to be done.

This research study investigated taxonomic applications that can help with identifying and classifying species, as well as providing support for the taxonomic revision procedure. This research includes a taxonomic case study dealing exclusively with Afrotropical bees due to a

collaboration with an acknowledged bee taxonomist. In the next section, more information on Afrotropical bees is provided.

2.2 Afrotropical Bees

The taxonomy of Afrotropical bees form the basis of the case study and this specific case provides the scope of this research study.

Bees belong to the order Hymenoptera (bees, wasps and ants): superfamily Apoidea: division Apiformes. Afrotropical bees belong to six families: Colletidae, Andrenidae, Melittidae, Halictidae, Megachilidae and Apidae, and comprise 99 genera and 2755 valid species excluding the honey bee [Eardley and Urban, 2010]. Afrotropical bees as the name suggests, are found in Sub-Saharan Africa which is the region south of the Sahara desert. Within the ecosystems in the southern hemisphere, Afrotropical bees are one of the biggest pollinator groups that provide pollination to many diverse high value crops to enable their successful growth [Eardley, 2002; Eardley et al., 2010]. Without these bees our food security could become compromised and thus have major effects on the world as we know it. Afrotropical bees play a vital role within agriculture and in maintaining biological diversity [Eardley, 2002]. Due to the bees' key function in biodiversity, it is crucial that knowledge about the behavioural patterns and characteristics of Afrotropical bees is maintained and increased to support ecological research, in order to continually understand their role in natural ecosystems [Eardley, 2002].

In Section 5.1.1 more detail is given of the specifics of the Afrotropical bee data which was used to create an ontology that captures the taxonomic information. *The Catalogue of Afrotropical Bees (Hymenoptera: Apoidea: Apiformes)* [Eardley and Urban, 2010] and the booklet, *The Bee Genera and Subgenera of Sub-Saharan Africa* [Eardley et al., 2010] were the main sources of data used.

2.2.1 Bee Taxonomist

Dr. Connal Eardley is a specialist scientist at the Agricultural Research Council[7] and holds a Doctorate of Philosophy in Entomology. Dr. Eardley is, amongst other things, a world renowned bee taxonomist. He carries out bee identification for a number of pollination projects in several different African countries [Eardley, C.]. With regards to Afrotropical bees, he is one of the only few taxonomists in Africa dealing with such taxa and has performed the revision of 269 valid species, 71 of which were new to science [Eardley, C.].

Dr. Eardley provided the domain expertise for this project, including the data needed to create the ontology for the application. Dr. Eardley's requirements included an all in one key classifier application, which could provide taxonomists a way to increase their productivity and support in some way the taxonomic revision practice. One of Dr. Eardley's requirements included the ability to identify the key features of a given set of Afrotropical bee species (or genera) or multiple species. Additional to identifying the key features was the capability to include the display of some extra information, such as the shared features common to multiple

[7] http://www.arc.agric.za/Pages/Home.aspx

species as well as the *uncommon* features (features that are possessed by at least one species, but not by *all* of them).

In the last section of this chapter some of the world wide environmental concerns that can become apparent due to a lack of taxonomy and biodiversity research are considered.

2.3 World Wide Environmental Concerns

Biodiversity plays a large role in maintaining the proper functioning of most of the earth's ecosystems [Biodiversity Brief]. There are many different components that make up these ecosystems, and knowledge about behavioural characteristics can most certainly assist in understanding and predicting changes within such biospheres [May, 1992]. Several factors such as the continuous rise of the human population and the resulting habitat loss are causes for concern, since human existence relies on natural systems for resources such as clean water, air and food [Stork, 1993]. In this section a brief overview is given of some of the worldwide environmental concerns that arise due to insufficient research within taxonomy and the naming and identifying of species [Mora et al., 2011]. Software applications for taxonomy, such as the one developed as part of this study, could assist with the resolution of these concerns.

Even though many species have been identified and named, there is still an exceptionally large number of them that have not yet been described on earth. Taxonomic classification has been practised for around 250 years and in that time approximately 86% of the existing species on earth and 91% of species in the ocean still await description, out of over 1.2 million species that have already been catalogued in a central database [Mora et al., 2011]. It is estimated that fewer than two million of an estimated 10-15 million species have been scientifically described [Biodiversity Brief]. Thus, there is still much to do in taxonomy with regards to the identification and classification of taxa in order to close quite a big knowledge gap [Mora et al., 2011].

One of the biggest factors holding back species identification is a shortage of taxonomists [Stork, 1993], and not enough funding and interest are provided for it [Guerra-García et al., 2008; Wheeler et al., 2004]. In addition, the role that biodiversity plays in the world is relatively unknown and is not as clear as it should be. Not enough emphasis is placed on biodiversity, and without decent funding and interest it is very hard to advance or to progress. Another factor contributing to the shortage of taxonomists, is that these experts are expected to execute additional work [Godfray, 2002]. The technological assistance to taxonomy could also hinder support for taxonomy. Most major museums have actual data and records dating back many years that have to be used by taxonomists, and the process to obtain such data can be time consuming [Hoagland, 1996]. Additionally, the museums also have several unstudied specimens and undescribed new species, yet to be digitised and analysed [Hoagland, 1996]. What could end up happening is that by the time something is done and more taxa start to be identified and described, there could have been other species that have recently become extinct, which may never be known along with the effect their behaviour had on the ecosystem [Mora et al., 2011].

Knowing and understanding the different species that exist within the world and within particular habitats and ecosystems can be useful. The various living organisms all over the world offer something different to the many biological processes that occur [Kapoor, 1998]. These processes usually contribute to the production of several natural resources that humans depend upon such as clean air, water, food and medicine [Conte and Navajas, 2008]. Biodiversity plays a huge role in the effective functioning of ecosystems and more so within rural areas where it is severely depended upon for survival. Crops are grown and are usually the main source of food and income for the local people, as well as clean water and herbal medicines, which are based on plants and animals [Biodiversity Brief; Sodhi et al., 2009]. If these natural resources become compromised, many people, especially the poorer population will be affected and their livelihood threatened. Not only will food supplies be under jeopardy, but health can become a major issue with the onset of diseases from polluted water and air, and from loss of raw materials that are used for medicinal purposes [Biodiversity Brief]. Such effects can occur when natural habitats are torn down and modified for commercial reasons. The destruction of these ecosystems cause many valuable species of animals and plants to be destroyed [Sodhi et al., 2009]. What would happen if an unidentified species that plays part to a crucial natural process gets destroyed or even extinct in the process? There are many ramifications that it could result in. When humans interfere and take over these habitats they depend on what little resources are left, and when their populations see substantial growth those resources can become exhausted due to their decreased availability.

As has been shown, biodiversity has numerous environmental and health effects on us and our surroundings. By knowing more about the various species and their behaviours, trends can be analysed and certain more important species can then have emphasis placed upon them for their preservation in order to prevent the extinction of many organisms [Hoagland, 1996]. It can also become easier to identify what processes certain species are involved in, and how these processes affect our continued existence and health. Through species description, more knowledge of the species is gained and the ability to aid in their conservation is increased [Hoagland, 1996].

Returning to a more specific case of Afrotropical bees. Bees are one of the largest pollinator groups in Southern Africa, which makes them a very important group of organisms, especially to the human race [Eardley, 2002]. They use pollen and nectar that have been collected from flowers to feed their progeny. When bees visit the flowers and crops to collect such products they end up *pollinating* them, which enables new seed production of many different species of flowering plants [Eardley, 2002]. Therefore, many agricultural crops are dependant on such bees for this pollination process in order to produce fruit and/or seeds to enable growth [Conte and Navajas, 2008]. Without these incredible organisms and their abilities many plants will fail to yield any produce resulting in a decline of the earth's natural food resources. This could affect many ecosystems. Thus the maintenance and monitoring of these pollinators and their behavioural characteristics is essential for sustainable agriculture and to protect our food resources from becoming severely threatened [Conte and Navajas, 2008]. In order to do this, an accurate understanding and knowledge of the systematics of Afrotropical bees is

critical. Through continued research trends in the species behaviours can be identified and the organisms can be monitored to preserve them.

This chapter discussed several different aspects of the case study that provided the scope of this research project. Details of taxonomy, Afrotropical bees and their associated roles were explained to show their importance in the world. The next chapter provides background to ontologies and DLs, which are a group of knowledge representation languages that are most often used to represent knowledge about a domain. They are thus used in ontological modelling for a specific domain of interest.

Chapter 3

Background

Introduction

This chapter commences with a discussion on ontologies as well as some examples to illustrate their application. A brief explanation of an ontology editor, Protégé, which is used to construct and manipulate ontologies (and was used to model the Afrotropical bee ontology that is looked at in Section 5.1.2) is also given. An overview of various ontology methodologies that exist is presented before moving on to DLs. An introduction and description of DLs, which are used to portray information about some domain and can be used to represent an ontology, is given in the second section. Reasoning is also introduced along with a few of the different types of DLs. Following from this, one of the most basic DLs, $\mathcal{ALC}$ and its syntax and semantics is presented in order to discuss the DL's expressivity. The correlation between DL expressivity and reasoning complexity is then discussed with descriptions of the various letters that are used to name the numerous features that make up DLs.

In section three, a summary of the details pertaining to the semantic web and the Web Ontology Language (OWL) is given, including some background information on what the Semantic Web is and its history. OWL, with its standards is discussed as well.

The final section presents some of the different standard reasoning services that are used to deduce certain logical facts about a domain.

3.1 Ontologies

Any domain or field of knowledge can be represented as an ontology. Gruber [1993] stated that anything that merely exists can be represented, supporting the notion of ontologies. Ontologies, thus depict a representation of some domain. An ontology uses axioms or sets of syntactic sentences to describe the specific domain and an ontology can be defined as a shared, formal explicit specification of a conceptualisation [Gruber, 1993]. An ontology is composed of a set of concepts, roles and objects. Assertions can be made about such concepts, and roles can be used to assert a relationship between those concepts.

Guarino [1998] provided a definition of an ontology stating that "it is an engineering artefact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary."

Using a domain of a small business or organisation for example, concept names could be Manager, Employer, Employee and Task. Role names such as isEmployedby, employs, creates, and isAssigned could be used. In order to then represent this business domain a combination of these constructs would need to be formed into sentences to do so, for instance, a Manager creates certain Tasks or an Employee isAssigned to a certain Task. Together such sentences or axioms describe and define the ontology. Figure 3.1 shows a diagram depicting some of the axioms in an example ontology. For simplicity, an ontology can be thought of as a controlled

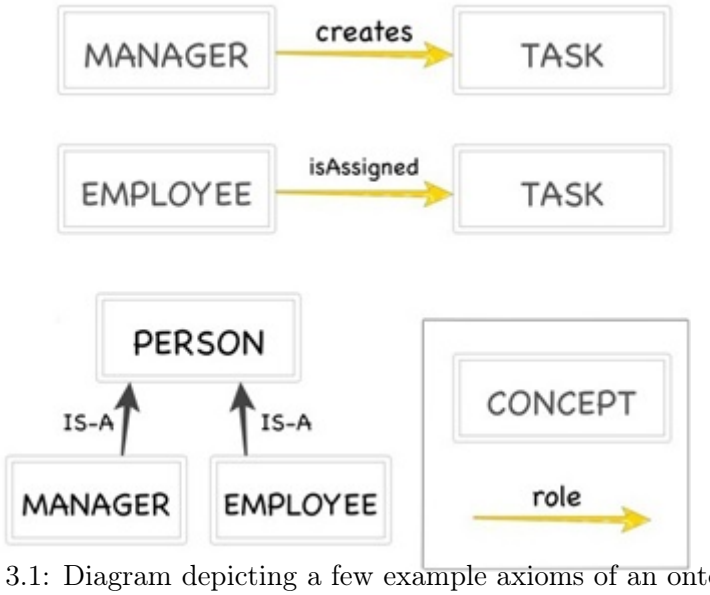


Figure 3.1: Diagram depicting a few example axioms of an ontology

vocabulary i.e. a finite list of terms [McGuinness, 2005]. These terms and sets of sentences constitue the essence and meaning of the ontology and give it structure. The hierarchical nature of an ontology allows concepts to be more general than others. The sentence, "Every Manager is a Person", or "Every Employee is a Person", allows for a more general concept such as Person, which is a super class of Manager and Employee, as well as a more precise concept, Manager or Employee, which are subclasses of Person. It can be seen that the Manager and Employee concepts are more specific than the Person concept in this case.

Ontologies can also be described as machine interpretable definitions of basic concepts as well as the relations between them within a domain, which also allow for information in a domain to be shared between researchers [Noy et al., 2001]. A few reasons to construct an ontology or where an ontology is beneficial as highlighted by Noy et al. [2001] may include:

- To be able to share knowledge and the structure of information. Through ontologies, similar domains can share and publish their information, which can be accessed and used by people or software agents over the World Wide Web.
- The reuse of domain knowledge and semantics. Other researchers can use already developed ontologies for their domain depending on what they require. Researchers can also add to or integrate existing ontologies to create a larger ontology within the domain.

- To make domain assumptions explicit. If knowledge about the domain alters, then changing such assumptions is easier and better than having them hard coded which would make them difficult to find and understand.

There are different *ontology languages* that formalised ontologies can be represented by, such as Unified Modelling Language (UML) [Rumbaugh et al., 2004], DLs [Baader et al., 2003] and Semantic Networks [Sowa, 2014]. A logic-based knowledge representation language known as Description Logics exists, equipped with formal semantics and supported by powerful reasoning tools, which in turn makes it a popular language to represent ontologies and information about a domain [Baader et al., 2003]. A more detailed look at DLs is given in section two of the chapter, after a discussion on various ontology engineering methodologies, and on an OWL ontology editor, Protégé, which is presented next.

3.1.1 Protégé

There are various ontology editors available that can be used to construct and edit ontologies. The editors act as a graphical interface for the development of ontologies and provide for an easy way to manipulate them without having to use detailed OWL syntax, which is an ontology language used when constructing ontologies. Protégé is one of many ontology editors, a few others are available such as SWOOP [Kalyanpur et al., 2006], NeOn Toolkit [Neon Project] and Apollo [Apollo].

Protégé is a free and open source ontology editor that was originally developed by the Stanford Center for Biomedical Informatics Research at the Stanford University School of Medicine [Protégé; Gennari et al., 2003]. The latest version is Protégé 5 for desktop use but an online version called WebProtégé also exists [WebProtégé; Tudorache et al., 2008]. WebProtégé enables the creation, modification and sharing of ontologies over the web. Other previous major versions are Protégé 3.x and Protégé 4.x. Protégé 4 was developed by the University of Manchester in collaboration with Stanford Medical Informatics as part of the CO-ODE project [CO-ODE Project] and was tailored for use with the latest OWL 2 standard of ontology languages [Krötzsch, 2012].

Protégé is customizable and has a number of plugins that can be installed and incorporated into it, most notably a few different reasoners such as Hermit [Shearer et al., 2008], Pellet [Sirin et al., 2007] or FaCT++ [Tsarkov and Horrocks, 2007, 2006].

When discussing the Afrotropical bee ontology that was developed as part of this research paper, in Section 5.1.2, illustrations of the ontology are provided, which show it within the Protégé environment.

3.1.2 Ontology Engineering

Many ontology engineering methodologies exist within the realm of ontology engineering. Some though, are more technical and rely on specialised knowledge engineers [Braun et al., 2007]. Ontology methodologies have been established using Artificial Intelligence (AI) research

methods and techniques, as well as through actual ontology construction and modelling, thus many may be ontology or domain specific. There are also methodologies that have been formulated through non AI techniques, as Gomez-Perez et al. [2006] discuss a few different ways of ontology modelling using software engineering techniques, as these are sometimes easier to understand for non AI communities. Similarly, Spyns et al. [2008] presents a methodology for ontology engineering from scratch. The methodology is inspired by various scientific disciplines such as database semantics and natural language processing. Uschold and Gruninger [1996] established some of the initial methodologies in the domain of enterprise modelling and business enterprises.

Within most methodologies there are certain principles that are used in line with the methodologies, for example Gruber [1993] identified five principles for designing ontologies to be used in knowledge sharing, which are clarity, coherence, extendibility, minimal encoding bias and minimal ontological commitment. It should be kept in mind when constructing an ontology, that it should really be seen as a process of continuous evolution rather than a one-time activity [Braun et al., 2007]. The METHONTOLOGY methodology presents an approach to build ontologies from scratch, and provides extensive details of the various phases and techniques involved [Fernández-López et al., 1997]. It follows a life cycle based on evolving prototypes and also supports the notion of reusability. METHONTOLOGY is based on the experience acquired in developing an ontology in the domain of chemicals as well as environmental pollutants [Fernández-López et al., 1997]. The NeOn Methodology provides a few different scenarios that cover some of the most common occuring situations. This gives a lot more options to the ontology creator and allows for a variety of ways to build and model an ontology. The NeOn Methodology is a scenario-based methodology that supports different aspects of the ontology development process, as well as the reuse and dynamic evolution of networked ontologies in distributed environments [Suárez-Figueroa et al., 2012]. It also allows for the inclusion and analyis of knowledge, by different people at different stages of the ontology development process. Noy et al. [2001] establishes certain basic ways and methods of constructing an ontology, taking into account the domain data and what exactly is trying to be represented and achieved. Noy et al. [2001] also determines that there is no one *correct* way or methodology for modelling an ontology. How detailed or general the ontology is, will determine much of the modelling decisions. Essentially though, it comes down to the application of the ontology and its intended use, thus not many *mature* methodologies exist [Iqbal et al., 2013]. The ontology that was developed as part of this research study was modelled and constructed using the basic approach by Horridge and McGuinness [Horridge et al., 2009; Noy et al., 2001]. A few basic principles were borrowed from an established taxonomic ontology, the Hymenoptera Anatomy Ontology (HAO) [HAO; Yoder et al., 2010b; Deans et al., 2012], which is discussed in more detail in Chapter 4. An essential part of modelling the ontology that was kept in mind, was to keep a pragmatic approach in order to allow the domain expert to understand and participate, and to keep it as close to the existing domain data's structure as possible.

The next section discusses the logic-based knowledge representation language known as Description Logics.

3.2 Description Logics

This section is focused on DLs, which are logic-based knowledge representation languages, used for instance, for constructing ontologies [Krötzsch et al., 2012]. Initially a broad overview of DLs, their constituents, and a few simple real world examples are given. Reasoning is also discussed. A few different types of DLs are then introduced and what they are suited for, which leads to the next section where one of the basic DLs, $\mathcal{ALC}$ is presented. $\mathcal{ALC}$'s associated syntax and semantics are given with some supporting examples. Finally, the association between DL expressivity and reasoning complexity is discussed, along with descriptions of the various features of DLs.

DLs are a group of knowledge representation languages that are most often used to portray information about some domain [Baader et al., 2003]. DLs are also consequently used in ontological modelling and are widely used in representing an ontology or a specific domain of interest [Krötzsch et al., 2012]. DLs specify the definitions of the appropriate entities within the domain and then provide a means to model their properties and the relationships between entities. An advantage in using DLs is that they are equipped with formal model-theoretic semantics, due to them being formulated as decidable subsets of first-order logic [Kleene, 2002]. This semantics means that the ambiguity within the meanings of terms and sentences that are expressed as DLs, is eliminated. A related advantage of DLs is their ability to support reasoning functionality over a domain [Krötzsch et al., 2012]. This reasoning capability allows additional information and implicit represented knowledge to be derived from the explicitly stated axioms using the DL.

DLs consist of atomic concepts, atomic roles and individuals, which comprise their basic building blocks [Krötzsch et al., 2012]. Atomic concepts represent sets of entities from the domain and can be seen as unary predicates looked at from a first order logic perspective [Kleene, 2002]. An example of an atomic concept could be a concept name, Employee, which represents all the employees in a certain organisation. Atomic roles represent binary relations between the entities and can be seen as binary predicates looked at from a first order logic sense [Kleene, 2002]. An example of an *a*tomic role, managedBy, would represent the (binary) relationship between employees and their associated managers, which in turn would be the set of all employees who have managers. Thus the sentence *James* managedBy *Susan* formally states that the entities within the domain that are represented respectively by the names, James and Susan, are related to each other via the atomic role that is represented by the role name managedBy. Individuals or constants are the actual objects in the domain, which can be instances of any of the atomic concepts [Krötzsch et al., 2012]. An example could be the previously used entities, such as James or Susan that could represent individual employees or managers. In order to define more complex concepts and roles, concept and role constructors are used, which are used in conjunction with the basic building blocks of DLs [Krötzsch et al., 2012].

A primary advantage of DLs is that they can be applied in the formal specification of ontologies [Krötzsch et al., 2012]. DLs are equipped with a precise model-theoretic semantics and have the capability of computing inferences, or in other words, providing reasoning over ontologies

[Baader et al., 2003; Krötzsch et al., 2012]. Ontologies can be represented by DLs in terms of a set of statements or logical sentences, which are known as axioms and in this case more so as DL axioms. By having such formal semantics, it means that each of those axioms has a precise meaning and little ambiguity exists about what they represent in the domain. Eliminating ambiguities is why DLs are so advantageous for representing ontologies over other modelling languages such as UML [Krötzsch et al., 2012; Rumbaugh et al., 2004].

Reasoning is a means of deriving implicit knowledge from a set of explicitly stated facts in a domain or ontology [Baader et al., 2003]. The stated facts are represented as axioms in a DL ontology. For instance, if we state, as axioms, that *"Jenny is a mother"* and *"mothers have children"* we can then infer that, *"Jenny has children"*. Using the stated axioms to derive a conclusion and compute the inference *"Jenny has children"*, is the essence of reasoning, which can be executed over DL ontologies automatically by software inference engines known as DL reasoners[1] [Shearer et al., 2008; Sirin et al., 2007; Tsarkov and Horrocks, 2006]. The reasoners are software applications that are responsible for deriving logical implicit consequences within an ontology, based on inference rules and on an asserted model of the domain [Baader et al., 2003; Gardiner et al., 2006].

Within DLs there are different families or types of DLs and these various DLs have varying levels of expressivity [Baader et al., 2003]. The level of expressivity is determined by adding or removing constructors that are used within the DL, which come with a trade off in computational complexity [Levesque and Brachman, 1984; Brachman and Levesque, 1987; Baader et al., 2003]. Some DLs are more complex or more expressive than others, which means that they can handle more features. The more expressive the DL the more complex the reasoning for that DL is. Thus by adding more expressive features to a DL the more complicated the reasoning becomes. Particular DLs are therefore generally better suited for specific reasoners, which can handle their level of complexity more efficiently [Krötzsch et al., 2012]. As such there are a number of varying reasoners and a number of different DLs for different purposes but the best balance between expressivity of the language and complexity of reasoning depends on the intended application [Krötzsch et al., 2012].

There are many different types of DLs, but for this discussion we focus on the basic DL $\mathcal{ALC}$ [Baader et al., 2003; Krötzsch, 2012]. Other DLs can be built onto or even reduced from $\mathcal{ALC}$ by adding or removing certain constructs, which will be presented in a later section. The syntax and semantics of the DL $\mathcal{ALC}$ is given next.

3.2.1 The Description Logic $\mathcal{ALC}$

$\mathcal{ALC}$ (Attributive concept description Language with Complements) [Schmidt-Schauß and Smolka, 1991] is one of the more important and fundamental DLs that can be extended or restricted in order to make up other DLs, depending on the complexity and expressivity required for specific applications [Levesque and Brachman, 1984; Brachman and Levesque, 1987; Baader et al., 2003]. $\mathcal{ALC}$ is often used as the basic example when introducing DLs in

[1] A list of reasoners can be found at `http://www.cs.man.ac.uk/~sattler/reasoners.html` and `http://owlapi.sourceforge.net/reasoners.html`

the literature as it includes all the basic constructors used in most applications. A look at the syntax and semantics of the DL $\mathcal{ALC}$ is presented next. The definitions and theorems presented in the remainder of this chapter are adapted from the DL Handbook [Baader et al., 2003].

$\mathcal{ALC}$ Syntax and Semantics

$\mathcal{ALC}$ includes the following DL constructors [Baader et al., 2003]:

- The $\top$ (Top) and $\bot$ (Bottom) special concepts
- $\sqcap$ (concept conjunction), $\sqcup$ (concept disjunction) and $\neg$ (concept negation)
- In addition, the complex concept constructors $\exists$ (existential restriction) and $\forall$ (value restriction) are also available.

The $\mathcal{ALC}$ concept syntax is defined below, followed by the syntax for $\mathcal{ALC}$ sentences (axioms).

Let N_C denote a set of atomic concept names. VegetarianPizza and Dog are examples of such concept names. Let N_R be a set of role names. hasPizzatopping and hasPet are examples of such role names. The set of $\mathcal{ALC}$ *concept descriptions* is the least set such that:

- $\top$, $\bot$ and every atomic concept name A $\in N_C$ is an $\mathcal{ALC}$ concept.
- If C and D are concepts and R $\in N_R$, then the following are concepts:
  - C $\sqcap$ D (the intersection of two concepts is a concept)
  - C $\sqcup$ D (the union of two concepts is a concept)
  - $\neg$C (the complement of a concept is a concept)
  - $\forall$R.C (the universal restriction of a concept by a role is a concept)
  - $\exists$R.C (the existential restriction of a concept by a role is a concept)

Examples of $\mathcal{ALC}$ complex concepts are: $\neg$(Dog $\sqcup$ Cat) (which, in natural language, represents the set of entities in the domain that are neither a dog nor a cat), $\exists$hasPizzatopping.Mushroom (which, in natural language, represents the set of entities in the domain that have at least one pizza topping consisting of mushrooms) and $\exists$hasPizzatopping.$\top$ (which, in natural language, represents the set of entities in the domain that have at least one pizza topping). Any concept that follows the role name in a concept description is called a *filler* concept for that role name. Consider the concept description $\exists$hasPizzatopping.Mushroom, the filler concept for the role hasPizzatopping is Mushroom. In a case where the filler concept is $\top$, the concept description can be abbreviated to omit $\top$, the reason being that the meaning of the description is the same without it. Thus, $\exists$hasPizzatopping has the same meaning as $\exists$hasPizzatopping.$\top$.

Description Logic sentences are usually classified as axioms and assertions, presented next, where the syntax of $\mathcal{ALC}$ sentences is defined [Baader et al., 2003].

Let C and D be $\mathcal{ALC}$ concepts, R be a role name and a and b, individual names.

- C $\sqsubseteq$ D and C $\equiv$ D are axioms, where $\sqsubseteq$ is the *subsumption* symbol and $\equiv$ is the *equivalence* symbol.
- C(a) is an $\mathcal{ALC}$ concept assertion and R(a,b) is an $\mathcal{ALC}$ role assertion, both are $\mathcal{ALC}$ assertions.

Examples of $\mathcal{ALC}$ axioms are: Lecturer $\sqsubseteq$ Person, Parent $\equiv$ Mother $\sqcup$ Father and Father $\sqsubseteq$ $\exists$hasChild.$\top$. An example of an $\mathcal{ALC}$ concept assertion is Employee($Jonathon$) and an $\mathcal{ALC}$ role assertion is ismanagedBy($Jenny$, $James$).

The meaning or semantics of $\mathcal{ALC}$ concepts and sentences are given next [Baader et al., 2003].

Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation with the non-empty set $\Delta^{\mathcal{I}}$ representing the domain of $\mathcal{I}$ (a set of objects or individuals), and function $\cdot^{\mathcal{I}}$ which maps:

- Every $\mathcal{ALC}$ concept name to a subset of $\Delta^{\mathcal{I}}$
- Every role name to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and
- Individual names a and b to elements $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ and $b^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ respectively.

The interpretation function $\cdot^{\mathcal{I}}$ is extended to complex concepts in the following way. For every $\mathcal{ALC}$ concept C and D and every role name R:

Definition 3.1 ($\mathcal{ALC}$ complex concept semantics)
Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation. Then:

- $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$, $\bot^{\mathcal{I}} = \emptyset$
- $(\mathrm{C} \sqcap \mathrm{D})^{\mathcal{I}} = \mathrm{C}^{\mathcal{I}} \cap \mathrm{D}^{\mathcal{I}}$ (union means disjunction)
- $(\mathrm{C} \sqcup \mathrm{D})^{\mathcal{I}} = \mathrm{C}^{\mathcal{I}} \cup \mathrm{D}^{\mathcal{I}}$ (intersection means conjunction)
- $(\neg \mathrm{C})^{\mathcal{I}} = \Delta^{\mathcal{I}} \backslash \mathrm{C}^{\mathcal{I}}$ (complement means negation)
- $(\forall \mathrm{R}.\mathrm{C})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \text{For all } b \in \Delta^{\mathcal{I}}, (a, b) \in \mathrm{R}^{\mathcal{I}} \text{ implies } b \in \mathrm{C}^{\mathcal{I}}\}$
- $(\exists \mathrm{R}.\mathrm{C})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \text{There is a } b \in \Delta^{\mathcal{I}} \text{ s.t. } (a, b) \in \mathrm{R}^{\mathcal{I}} \text{ and } b \in \mathrm{C}^{\mathcal{I}}\}$

Consider the complex concept, $\forall$hasPet.Dog, which represents the set of all the individuals in the domain such that, if they have one or more pets, then these pets can *only* be dogs. Within this concept something to be aware of, is that any individual that is not related via the role hasPet to any individual residing in the domain, also appears (by vacuity) in the concept given in this example, $\forall$hasPet.Dog. This is shown in the fifth point in the semantics of $\forall$R.C given above.

In another instance consider the complex concept $\exists$hasPet.$\top$, which represents the set of all the individuals in the domain such that, each of these individuals is related to at least one individual in the domain via the role hasPet. In other words, it represents the set of all individuals that have at least one pet.

The meaning for $\mathcal{ALC}$ axioms and assertions is formally defined next [Baader et al., 2003].

Definition 3.2 (Satisfaction of $\mathcal{ALC}$ sentences in an interpretation)
Let C and D be $\mathcal{ALC}$ concepts, R be a role name and a and b, individual names. Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation. Then:

- $\mathcal{I}$ *satisfies* $C \sqsubseteq D$ if and only if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
- $\mathcal{I}$ *satisfies* $C \equiv D$ if and only if $C^{\mathcal{I}} = D^{\mathcal{I}}$
- $\mathcal{I}$ *satisfies* $\mathrm{C}(a)$ if and only if $a^{\mathcal{I}} \in C^{\mathcal{I}}$
- $\mathcal{I}$ *satisfies* $\mathrm{R}(a, b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$

A few examples are now given to demonstrate each type of axiom and assertion. A *subsumption axiom* such as $\mathsf{PetOwner} \sqsubseteq \exists \mathsf{hasPet}.\top$, in natural language means that, "if some individual in the domain is a Pet Owner, then this individual has at least one pet".

An *equivalence axiom* such as $\mathsf{Pet} \equiv \mathsf{Dog} \sqcup \mathsf{Cat}$ means that (assuming that it has been specified that a dog is not a cat), "an individual of the domain is a Pet if and only if it is either a Dog or a Cat".

A *concept assertion*, $\mathrm{C}(a)$ means that "the individual referred to by a belongs to the set referred to be C". For example consider the concept assertion $\mathsf{Dog}(Jock)$, where Dog is a concept name and $Jock$ is an individual name, it means that Jock is a dog (or "Jock is an instance of Dog").

A *role assertion*, $\mathrm{R}(a, b)$ means that the individual referred to by a is related to the individual referred to by b via the role represented by R. For example consider the role assertion $\mathsf{hasPet}(Jane, Jock)$, where hasPet refers to a role and $Jane$ and $Jock$ are individual names, it means that Jane has a pet named Jock.

Table 3.1, shows a summary of the $\mathcal{ALC}$ syntax and semantics [Schmidt-Schauß and Smolka, 1991; Baader et al., 2003].

Name	**Syntax**	**Semantics**
Top concept	$\top$	$\Delta^{\mathcal{I}}$ (all individuals)
Bottom concept	$\bot$	$\emptyset$ (no individuals)
Conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
Concept negation	$\neg C$	$\Delta^{\mathcal{I}} \backslash \mathrm{C}^{\mathcal{I}}$
Value restriction	$\forall R.C$	$\{a \in \Delta^{\mathcal{I}} \mid$ For all $b \in \Delta^{\mathcal{I}}$, $(a, b) \in \mathrm{R}^{\mathcal{I}}$ implies $b \in \mathrm{C}^{\mathcal{I}}\}$
Full existential restriction	$\exists R.C$	$\{a \in \Delta^{\mathcal{I}} \mid$ There is a $b \in \Delta^{\mathcal{I}}$ s.t. $(a, b) \in \mathrm{R}^{\mathcal{I}}$ and $b \in \mathrm{C}^{\mathcal{I}}\}$
Subsumption	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
Equivalence	$C \equiv D$	$C^{\mathcal{I}} = D^{\mathcal{I}}$
Concept instance	$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
Role instance	$R(a, b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$

Table 3.1: Summary of $\mathcal{ALC}$ syntax and semantics.

The sentences that a DL ontology consists of can be divided into two separate sets. The first being the set of sentences that represent information about the relationships between concepts, termed axioms (as has been introduced earlier). The second being the set of sentences that represent assertions about individuals in the domain, called assertions. The sets serve a specific purpose within an ontology, and are termed Terminological Axioms (TBox) and Assertional Axioms (ABox) respectively [Krötzsch et al., 2012; Baader et al., 2003].

Definition 3.3 (Ontology, TBox and ABox)
A TBox $\mathcal{T}$ is a finite and possibly empty set of axioms. An ABox $\mathcal{A}$ is a finite and possibly empty set of assertions. If $\mathcal{T}$ is a TBox and $\mathcal{A}$ is an ABox, then $\mathcal{O} = \mathcal{T} \cup \mathcal{A}$ is an ontology.

$\mathcal{ALC}$ is not an overly expressive DL and some applications may require more expressivity or less expressivity than what $\mathcal{ALC}$ offers, depending on their intended purpose [Krötzsch et al., 2012]. The next section discusses this DL expressivity and the complexity of reasoning associated with it, as well as the correlation between expressivity and reasoning complexity.

3.2.2 DL Expressivity and Reasoning Complexity

There are many different types of DLs such as $\mathcal{ALC}$, which has been described. $\mathcal{ALC}$ is considered to be a basic DL and thus not very expressive [Krötzsch, 2012]. For example, it does not include *cardinality restrictions* (allows for the expression of restrictions on the number of elements a concept may be related to via a role) [Baader et al., 1996]. By adding more DL constructors or features to a DL, the expressivity increases [Levesque and Brachman, 1984; Brachman and Levesque, 1987; Baader et al., 2003]. Depending on the specific application, additional attributes to the DL $\mathcal{ALC}$ for instance, may be required in order to more accurately represent the domain.

DLs are named according to the features that they include [Krötzsch et al., 2012; Krötzsch, 2012]. Normally specific letters (or symbols) that are used in the name of the DL represent specific features. For instance the $\mathcal{AL}$ (Attributive Language) in $\mathcal{ALC}$ is the base language that allows atomic negation (negation of concept names that do not appear on the left hand side of axioms), concept conjunction, concept disjunction, value restrictions and limited existential quantification [Krötzsch, 2012]. The $\mathcal{C}$ represents complex concept negation. Other symbols such as the symbol $\mathcal{S}$ is an abbreviation for $\mathcal{ALC}$ with *transitive roles* [Horrocks and Sattler, 1999]. Transitivity of roles is defined by the following:

A role R is transitive if, for every interpretation $\mathcal{I}$, $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$ and $(b^{\mathcal{I}}, c^{\mathcal{I}})$ *in* $R^{\mathcal{I}}$ implies that $(a^{\mathcal{I}}, c^{\mathcal{I}}) \in R^{\mathcal{I}}$. Where a, b and c are individual names.

$\mathcal{H}$ represents the feature for *role hierarchies* [Horrocks and Sattler, 1999], where roles can be included in other roles. For example $\mathsf{hasDog} \sqsubseteq \mathsf{hasPet}$. $\mathcal{I}$ depicts *inverse properties/roles* [Horrocks and Sattler, 1999] and the letter $\mathcal{O}$ represents the use of nominals [Schaerf, 1993], which allow for additional information about the actual elements in the domain to be represented. $\mathcal{Q}$ caters for *qualified cardinality (number) restrictions* (number restrictions that have fillers

other than $\top$) [Tobies, 2000; Baader et al., 2003]. These are just a few of the DL feature symbols available, which can then be formed to make other more complex DLs such as $\mathcal{SHOIN}$ [Krötzsch et al., 2012] or $\mathcal{SROIQ}$ [Horrocks et al., 2006]. The ontology that has been developed as part of this study (discussed in section 5.1.2), has an expressivity of $\mathcal{ALCHQ}$. Thus the ontology makes use of the basic $\mathcal{ALC}$ DL, as well as allows for role hierarchies and qualified number restrictions.

$\mathcal{ALC}$ is an example of a DL that has a relatively good balance between expressivity and complexity of reasoning [Baader et al., 2003]. There are some DLs that are designed for applications that do not require high levels of expressivity and are more focused on tractability (polynomial decidability). These DLs include fewer constructs than $\mathcal{ALC}$ and in turn, the lower expressivity translates to lower reasoning complexity and faster classification times [Krötzsch, 2012]. Essentially by having fewer constructs there are limits on what information can be represented, but the gain is ensuring the complexity of reasoning is lessened.

There are some applications and ontologies that are more suited to utilising fewer constructs than $\mathcal{ALC}$, common cases are with extremely large ontologies that have thousands of concepts and axioms. Reasoning over such large scale ontologies can become cumbersome and complex due to the loads of information that they contain. With fewer constructs the reasoning becomes less expressive [Brachman and Levesque, 1987; Baader et al., 2003]. A domain where large ontologies can be seen is within the biomedical field where copious amounts of medical terms and biomedical information is represented for use in medical information system applications. An example of such a medical ontology is SNOMED CT (Systematized Nomenclature of Medicine, Clinical Terms) [SNOMED CT; Spackman et al., 1997], which is a massive ontology having a few hundred thousand concepts and axioms in it. Reasoning over so much information can be a tremendous task in itself. However, SNOMED is represented using a DL from the inexpressive $\mathcal{EL}$ family of DLs, which is well suited for such an ontology as it is quite inexpressive and also provides efficient reasoning properties [Krötzsch, 2012; Baader et al., 2003]. Another example of a large biomedical ontology is the Gene Ontology [GO: The Gene Ontology; du Plessis et al., 2011], where major portions of it can be represented using the DL $\mathcal{EL}$ [Krötzsch, 2012].

One of the main aspects of Knowledge Representation Systems (KRS) is the ability to use reasoning to derive implicit information. DLs are used within KRS and allow for these reasoning tasks, made up of a few different types within the TBox and the ABox [Krötzsch et al., 2012; Baader et al., 2003]. As mentioned, the reasoning complexity increases as the DL expressivity increases, thus the computational complexity of the following reasoning services varies with the expressivity of the particular DL [Krötzsch, 2012; Brachman and Levesque, 1987].

TBox reasoning services include:

- Concept satisfiability testing is where checking is done to see if a concept description contains any individuals.
- Subsumption testing includes checks to ascertain whether a concept is a sub-concept or

a super-concept of another.

ABox reasoning services include:

- Consistency checking. ABox statements are checked to see that they are consistent with TBox statements to show whether an ontology has a model or not.
- Instance checking is done to determine if a specific individual belongs to a certain concept.

The semantic web is discussed next as well as the Web Ontology Language (OWL), which is based on DLs and used to construct ontologies. The basic reasoning tasks that DLs support, including the services highlighted above, are presented as well.

3.3 The Semantic Web and OWL

The World Wide Web (WWW) [Berners-Lee, 1992] as we know it consists of a massive information space where large amounts of information and data can be found. The problem that lies herein is not with finding information, but more so obtaining quality information that is relevant to what we are looking for. Tim Berners-Lee[2], the inventor of the WWW had a two part vision for it [Berners-Lee et al., 2001; Shadbolt et al., 2006]. The first was to use it to connect people throughout the world to information and enable easier information sharing as well as communication between people. The second was to be able to give computers the capability of understanding and analysing the information on the web in the hopes of then providing more contextually relevant information [Berners-Lee et al., 2001]. In order to do this the idea of a *semantic web* was conceived, where the basic concept was to enrich the information on the web with meaning or semantics to allow websites to be processed by computers and ultimately be machine readable [Berners-Lee et al., 2001]. The semantic web would lead to more relevant information becoming accessible to people as well as to other computers and ultimately enable computers to speak to other computers.

Computers on their own cannot directly understand semantics unless they are encoded in a language that the computer can understand. For example sentences such as *"Jonathon is employed by Apple"* or *"Jonathon is the senior software developer"* cannot be understood by computers, but humans can attach meaning to such assertions and comprehend them within context of course. Following from this, web pages are enriched with semantic information to provide meaning and 'understanding' to a computer. The semantic web can be seen as an extension of the WWW [Berners-Lee, 1992] and is based upon the Resource Description Framework (RDF) [Lassila et al., 1998]. The RDF is what encodes semantics into the web pages. Ontologies are used by the semantic web to encode and represent such information in the websites that would lead to computers being able to process and analyse their content and attach relevant meaning to the website's information [Berners-Lee et al., 2001; Shadbolt et al., 2006]. Once the computer can understand the websites' content, the querying power and search ability can be improved resulting in better quality results [Shadbolt et al., 2006].

[2] https://www.w3.org/People/Berners-Lee/

The World Wide Web Consortium (W3C) [W3C] adopted the RDF and Resource Description Framework Schema (RDFS) [Lassila et al., 1998] as its Semantic Web language together with OWL [Horrocks et al., 2003; Grau et al., 2008]. When developing semantic web technologies something more appropriate than RDFS was required due to its limited expressive power, which led to the development of OWL and subsequently to the OWL 1 set of languages [Horrocks et al., 2003]. The OWL family was developed as a series of languages with formally defined meaning made up of different profiles or fragments, in order to represent semantic web ontologies [Horrocks et al., 2003; Krötzsch, 2012]. The different fragments were established to cater for the different complexity and expressivity that can exist within ontologies [Brachman and Levesque, 1987; Baader et al., 2003]. OWL languages are based on DLs but borrow syntactically from RDF/XML formats. Though OWL languages are based on DLs, they differ from them in that certain fragments of their languages allow for more expressive constructs [Horrocks et al., 2003]. OWL has been standardised by the W3C as a powerful knowledge representation language for the Web [Krötzsch, 2012].

The W3C originally set out to create and develop a language, which became known as the OWL 1 set of languages to be used to represent semantic web ontologies [Horrocks et al., 2003]. The set of languages are made up of three 'Species', namely the less expressive OWL Lite, the more so OWL DL and the most expressive OWL Full [Horrocks et al., 2003]. Unfortunately a few limitations in the syntax, semantics and expressivity of OWL 1 were identified that led to a new standard becoming developed in order to resolve the issues. The W3C then developed a new standard, OWL 2, which is at the time of writing the latest version of OWL according to the W3C's recommendation for the representation of semantic web ontologies [Grau et al., 2008]. OWL 2 consists of two basic 'flavours', OWL 2 DL and OWL 2 Full. Beneath these are three OWL 2 profile languages namely OWL 2 EL, OWL 2 QL and OWL 2 RL [Krötzsch, 2012; Krötzsch et al., 2012]. These light sublanguages are restricted in ways that significantly simplify ontological reasoning by restricting certain modelling features, and thus they are suited to specific ontological applications. In other words, the sublanguages are trimmed down versions of OWL 2 that trade some expressive power for the efficiency of reasoning [Krötzsch, 2012].

OWL 2 EL, is suited for large ontologies where performance and efficient reasoning is preferred [Krötzsch, 2012; Krötzsch et al., 2012]. OWL 2 EL is based on the DL $\mathcal{EL}$++ [Baader et al., 2003] family of DLs that is an extension of the $\mathcal{EL}$ language. The large biomedical ontology called SNOMED [SNOMED CT; Spackman et al., 1997] is one such example of an ontology based upon OWL 2 EL. OWL 2 EL suits SNOMED well because, while the ontology consists of thousands of concepts and roles, due to its nature it does not need highly expressive features that are available in other DLs. This in turn means that the reasoning is kept less complex than would be the case with having such expressive features involved [Brachman and Levesque, 1987; Baader et al., 2003].

OWL 2 RL (Rule Language) is commonly used to reason with web data where instance retrieval is an important inference task, and it does this through the use of implemented reasoning tasks as a set of ontology rules [Krötzsch, 2012; Krötzsch et al., 2012].

OWL 2 QL (Query Language) is based on the DL-Lite family of DLs [Krötzsch, 2012; Krötzsch et al., 2012] and provides database applications with an ontological data access layer, or ontology-based data access (OBDA) [Calvanese et al., 2007]. Through OWL 2 QL it is possible in OBDA to translate ontological queries into standard relational query languages such as sql [Krötzsch et al., 2012]. OWL 2 QL is designed to cater for applications that need to query large amounts of instance data also known as individuals, and it allows for efficient reasoning over such data.

The various reasoning services that were highlighted previously are presented next.

3.4 Standard Reasoning Services

Throughout the previous sections reasoning has been mentioned and how DLs and ontologies can be *reasoned* over to deduce a set of logically calculated facts about the domain and ultimately make implicit statements or knowledge, explicit [Baader et al., 2003; Krötzsch et al., 2012]. The domain models or ontologies are represented as axioms and assertions in a formal language such as OWL, which allows for such reasoning and logical inferences to be drawn from the model. Through the standardisation of OWL by the W3C a variety of tools and reasoners have been developed that support the development and querying of ontologies. An example of such a tool is the ontology editor Protégé [Protégé; Gennari et al., 2003], which makes use of some of its packaged reasoners such as Hermit [Shearer et al., 2008], Pellet [Sirin et al., 2007] or FaCT++ [Tsarkov and Horrocks, 2007, 2006] to query and reason with ontologies.

Next, some of the main reasoning services are discussed. Some of these tasks include: concept satisfiability, consistency checking, subsumption checking and instance checking. Many of these kinds of services have been implemented in the various DL reasoners available to cater for differing DLs, since some of the reasoning tasks are utilised only in certain DLs. Something to note is that it is possible for some of the reasoning services to be represented as special cases of one of the other tasks [Krötzsch et al., 2012].

3.4.1 Concept Satisfiability

Satisfiability testing is the process of checking and determining the possibility of a concept containing any individuals. A concept is said to be unsatisfiable if it cannot have any individuals or instances in any model of the ontology [Baader et al., 2003].

Given an ontology $\mathcal{O}$ with a concept, C, and an individual name a, C is satisfiable *if and only if* $C(a)$ is consistent in a model $\mathcal{I}$ of $\mathcal{O}$.

DL versions of Tableaux algorithms [Baader and Sattler, 2001] are used as the basis for most reasoners and in this instance they would try to prove the satisfiability of a concept C by constructing a model, an interpretation $\mathcal{I}$ in which $C^{\mathcal{I}} \neq \emptyset$ (is not empty).

3.4.2 Subsumption Checking

Subsumption testing is when checks are done to ascertain whether a concept is a sub-concept or a super-concept of another, or in other words to check whether some concept is more general than another one [Baader et al., 2003; Krötzsch, 2012]. Subsumption testing is based on the *'is-a'* role [Brachman, 1983], or is-a relation that is widely used in many different domains and thus is used within DLs to represent certain relationships. Examples could be, within an animal domain a dog can be represented as a type of pet or in the context of an organisation a manager can be a type of employee. These is-a relationships are so common in many application domains that they are encoded into the syntax and semantics of DLs as *subsumption* [Baader et al., 2003].

Given an ontology $\mathcal{O}$, C is subsumed by D if in every model of $\mathcal{O}$, C is subsumed by D (i.e. $C \sqsubseteq D$).

Most reasoners are equipped with the capability to perform subsumption testing since it is needed and used in many different applications. Another reasoning task that is of primary concern to reasoners and that can be seen as a somewhat special case of subsumption testing, is the process of *classifying* an ontology. The goal of classifying an ontology is to formulate and identify the sub-concepts and super-concepts for all the concept names in the ontology, to ultimately produce a *concept hierarchy* also known as the *taxonomy* of the ontology. Essentially the concept hierarchy provides an overview of the main concepts in the ontology and as with most reasoning tasks, the complexity of the classification varies with the complexity of the particular DL [Baader et al., 2003].

3.4.3 Consistency Checking

Consistency checking is a reasoning task whereby ABox statements are checked to see that they are consistent with TBox statements to show whether an ontology is logically consistent and has a model or not [Krötzsch, 2012; Baader et al., 2003]. If there is a model of the ontology, it means that there is an interpretation of it that satisfies all the sentences in the ontology. On the other hand, if no model can be made then the ontology is said to be *inconsistent*, which means everything in it is false or it contains contradicting statements, and thus $\top \sqsubseteq \bot$.

Tableau algorithms [Baader and Sattler, 2001] incorporated into reasoners attempt to construct a model and an interpretation $\mathcal{I}$ that satisfies every sentence in the ontology in order to determine if the ontology is consistent. If the interpretation $\mathcal{I}$ is found then the ontology is consistent.

3.4.4 Instance Checking

Instance checking involves determining if a specific individual within an ontology belongs to a certain concept or complex concept [Baader et al., 2003; Krötzsch, 2012]. Formally defined below:

Definition 3.4 (Instance checking)
Given an individual a and a concept C in an ontology $\mathcal{O}$, a is an instance of C with respect to $\mathcal{O}$ if and only if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ for all interpretations $\mathcal{I}$ of $\mathcal{O}$ if and only if $\mathcal{O} \models C(a)$.

It will be seen that within classification and more specifically taxonomy classification, there are certain classification questions that are usually asked when classifying objects. With regards to ontologies there is one specific question that cannot be answered using the standard reasoning tasks that have been described above. For such a question an additional reasoning algorithm needs to be incorporated on top of these standard reasoning services. The questions are used within an ontology based taxonomic key application to assist taxonomists with some of their tasks. The next chapter discusses classification and ontologies, and the specific classification questions or queries that need to be carried out. Also explained is the additional reasoning algorithm and how it manages to solve the queries.

Chapter 4

Classification and Existential Reasoning

Introduction

This chapter discusses classification in terms of biodiversity and the types of questions or queries that would usually be asked in taxonomic key applications. The answers to these questions aid the users in a number of crucial tasks. The questions can be translated into DL queries that can be asked in a suitably modelled ontology. With a particular question pertaining to taxonomic revisions, it is found that a standard DL query over the ontology using the standard reasoners does not yield the required results. Lengthy alternatives exist that entail the remodelling of the ontology, which would then enable the standard reasoners to answer the queries. In addition to the alternatives, another solution is presented and discussed, utilising a reasoning algorithm developed by Matthew Horridge of Stanford University [Matthew Horridge].

The chapter is composed of two sections. The first discusses classification and how it fits in the field of taxonomy and biodiversity in terms of taxonomic keys. Also presented is how ontologies fit in these domains and a few initiatives from around the world are highlighted to show the different types of *classification* based ontologies that are used within biology and taxonomy. Certain classification questions that are normally asked in order to classify objects are given and translated into DL syntax to see their application in an ontology.

In the second section it is established which of the queries the standard reasoners are able and unable to carry out. One of the queries that is useful for the taxonomic revision procedure cannot be executed and thus a few alternatives that involve the remodelling of aspects of the ontology are put forward to give options of what could be done. The solution of extending the standard reasoning capabilities using an existential reasoning algorithm is analysed and presented.

4.1 Classification and Ontology Classification Queries

The section begins by defining classification and its impact in taxonomy. The various ontologies that are in use within biology are highlighted, and some insight into the way these ontologies are modelled, is given.

The next part of the section presents the different classification questions that are typically

asked in order to classify objects. These questions are translated into DL syntax as DL queries that can be executed over an ontology.

4.1.1 Taxonomy, Classification and Ontologies

Classification is a general term but is used extensively in many different disciplines, predominantly within biodiversity and taxonomy. According to the Merriam-Webster dictionary [Merriam-Webster Dictionary] the definition of classification is "the act or process of putting people or things into groups based on ways that they are alike". In essence, classification is the procedure of grouping together certain things according to one or more distinct features that they possess [Guerra-García et al., 2008]. These unique features can then be used to identify certain items depending on the chosen characteristics. What seems like a relatively simple process can become quite intensive and complex in some situations. In section 2.1 classification in terms of taxonomy was introduced and discussed, where organisms, or groups of organisms are classified according to certain unique morphological features. Taxonomists are responsible for naming, identifying and grouping the organisms together according to such features [Culverhouse et al., 2003; Gaston and O'Neill, 2004]. This grouping allows for the differentiation between the various taxa and gives more structure and organisation to taxonomy.

Also discussed in Section 2.1 was taxonomic keys and the types that are in use today. The keys are useful in identifying species and assist taxonomists with their work [Walter and Winterton, 2007]. Single access keys are much more fixed and direct you on a path to follow when classifying a particular taxa by providing two or more choices to choose from [Guerra-García et al., 2008; Walter and Winterton, 2007]. Multi-access keys on the other hand offer a much more flexible approach allowing the user to choose whatever feature(s) they want to start with, and the ability to refine those characteristics until the desired species is found. Most of these keys are 'paper' based or in document formats [Eardley, 2012] where no interactivity is available as with a computerised application equipped with an interactive interface.

Ontologies are used in many domains today including the field of biology [Bard and Rhee, 2004]. There are certain sub domains that have utilised ontologies on a relatively large scale such as the Gene Ontology (GO) [GO: The Gene Ontology; du Plessis et al., 2011], SNOMED CT [SNOMED CT; Spackman et al., 1997], BIOTOP [Beisswanger et al., 2008] and The Environment Ontology (EnVO) [Bennett, 2010]. The Gene Ontology essentially describes gene products in terms of their associated biological processes, cellular components and molecular functions in a species-independent manner. With new discoveries the ontologies are constantly being reviewed, and consist of in the region of 40,000 biological concepts. SNOMED CT is a large scale ontology comprising of clinical health information and it is the most comprehensive clinical vocabulary available [Spackman et al., 1997]. SNOMED CT is concept-oriented and uses an advanced structure that meets most accepted criteria for a well-formed, machine-readable terminology. BIOTOP is an upper domain ontology for molecular biology [Beisswanger et al., 2008], and EnvO is an ontology that focuses on environmental concepts including terms for biomes, environmental features, and environmental material [Bennett, 2010]. EnvO is primarily intended for annotating data and samples collected by researchers in the biological, medical and environmental sciences [Bennett, 2010].

Taxonomy encompasses the classification of various taxa and in order for classification to be effective, the various taxa are defined and placed into groups [Guerra-García et al., 2008; Daniel, 2016]. A taxon is defined by specifying all of its unique, defining characteristics. Each *group* of taxa can also be defined in the same way. Similarly, a concept in an ontology is defined by declaring facts or characteristics about the concept, so that every element that belongs to the concept inherits those characteristics [Gruber, 1993]. Concepts can also be grouped together and defined. Thus the *conceptual* process of defining a concept in an ontology, and defining a taxon is similar [Franz and Thau, 2010; Schulz et al., 2008].

Within biodiversity and taxonomy there are a select few domains that have also created and made use of ontologies, such as the NCBI Taxon Ontology [NCBI], the Vertebrate Taxonomy Ontology [VTO; Midford et al., 2013] and a few more, which are listed at the OBO Foundry [The OBO Foundry]. The focus of these ontologies are mostly on interoperability, shared terminologies and meta-data annotations through the modelling of the various aspects of different species and taxa. Closely related to our case study is the Hymenoptera Anatomy Ontology (HAO) [HAO; Yoder et al., 2010b; Deans et al., 2012], which consists of the morphological modelling of insects, such as sawflies, wasps, bees and ants. The HAO was developed to create a common and shared terminology for Hymenoptera Anatomy, and to provide a platform to integrate the corpus of information about hymenopteran phenotypes that is inaccessible due to language discrepancies [Yoder et al., 2010a]. For instance, the understanding of certain widely known structures of taxa such as head, wing or leg, are generally shared by most people, but some communities of specialised hymenopterists may utilise different terms to the commonly shared expressions [Yoder et al., 2010a].

When it comes to the modelling of such ontologies as listed at the OBO Foundry or given above, Schulz et al. [2008] presents a detailed discussion on some of the ways biological taxa can be modelled and how the realm of biological systematics can be embedded into an ontological framework. On the other hand, Franz and Thau [2010] investigated to what extent a full-blown representation of biological taxonomy in the ontological domain is possible, while also analysing and commenting on Schulz et al. [2008] approach. A major role in this possibility, which they distinguish, will be the ability of the taxonomic expert community to present the data and products in a way that is more compatible with ontological principles. With regards to the modelling of the ontology of Afrotropical bees that forms part of this research study, a few concepts and structural ideas were borrowed from the HAO [HAO; Yoder et al., 2010b]. It was identified that the ontology would need to be kept as pragmatic as possible and as close to the structure of the taxonomic expert's data in order for the taxonomists to understand and use it. Some of the concept hierarchy principles from the HAO were borrowed and used in the ontology of Afrotropical bees. Various body part concepts are similar, thus could also be applied and had to be applied to the Afrotropical bees. The principles of the roles part_of and attached_to in the HAO were also adapted and used in our ontology. The roles are expanded and represented as roles such as hasPart, hasBodypart, hasAntenna and hasFeature. The HAO no doubt contains much more information and includes many terms with detailed annotations and figure annotations, as well as Universal Resource

Identifiers (URIs). The ontology of Afrotropical bees deals with a small number of genera (at this stage), but one specific taxa being the Afrotropical bees, unlike the multiple taxa of sawflies, wasps, bees and ants in the HAO. Our ontology's structure, where it could be, is modelled similarly to the taxonomist's data structures to keep it simple and pragmatic for the taxonomist, which will allow them in the future, the ability to also manipulate the ontology.

4.1.2 Ontology Classification Queries

Presented in this section are some of the questions or queries that are used when classifying an object or a species. The queries are given in natural language and are translated into DL syntax as would be used in an ontology.

First, a few aspects of the ontology are presented in order to give context to the kind of queries that will be used. The two main concepts in the ontology are the *'unique features'* or *'diagnostic features'*, and the actual species. These two concepts are related via an object property or role, hasDiagnosticFeature. What constitutes a diagnostic feature and the further modelling and structure of the ontology will be discussed and shown in much more detail in the next chapter. For now it will suffice to only look at this relevant portion of the ontology. Essentially a particular species has a particular diagnostic feature, but it can also have *many* diagnostic features. Normally a species has a few diagnostic features, the collective group of these contributes to distinguishing that species. In DL syntax then:

Axiom 4.1 (Bee Species and Diagnostic Features)
$\mathsf{BeeSpecies} \sqsubseteq \exists\ \mathsf{hasDiagnosticFeature}.\mathsf{DiagnosticFeature}$

Using an actual example: $\mathsf{Plesianthidium\ (Spinanthidiellum)\ volkmanni\ (Friese)\ male} \sqsubseteq \exists\ \mathsf{hasDiagnosticFeature}.\mathsf{MaleBasitarsusColour{:}Black}$ where 'Plesianthidium (Spinanthidiellum) volkmanni (Friese) male' is the species of Afrotropical bee that has a *basitarsus* (the first tarsal segment in the leg of an insect) that is black. 'MaleBasitarsusColour:Black' being the diagnostic feature in this case.

In terms of classification, when attempting to identify an organism (or any object), typically a taxonomic expert will examine and analyse the specimen and pinpoint the distinguishing characteristics. This identification is a difficult process and usually requires years of practice and experience to easily recognise the unique items. The taxonomist will then consult a key with the identified features in hopes that it will lead to the right species. The question asked would simply be:

Question 4.1
'Which species (or objects, in a general situation) exist that have this set of unique features?'

The results may include a number of species depending on the uniqueness, and the amount of features used, but as more are found and added to the question the list of species should become more refined.

When dealing with DL axioms such as the one presented above (Axiom 4.1), a query can

be formulated and asked over the axioms to retrieve the desired results. Reasoners carry out specific reasoning tasks over the axioms to eventually come to a conclusion and answer the queries put forward. If Question 4.1 above of *'which species exist that have these unique features?'* is translated using DL syntax then the query asked to the reasoner in an ontology editor such as Protégé [Protégé; Gennari et al., 2003] would be:

Query 4.1 (Find Species Query)
$\exists$ hasDiagnosticFeature.DiagnosticFeature (or hasDiagnosticFeature some DiagnosticFeature in Protégé as seen in Figure 4.1), where DiagnosticFeature is the feature selected and input by the user.

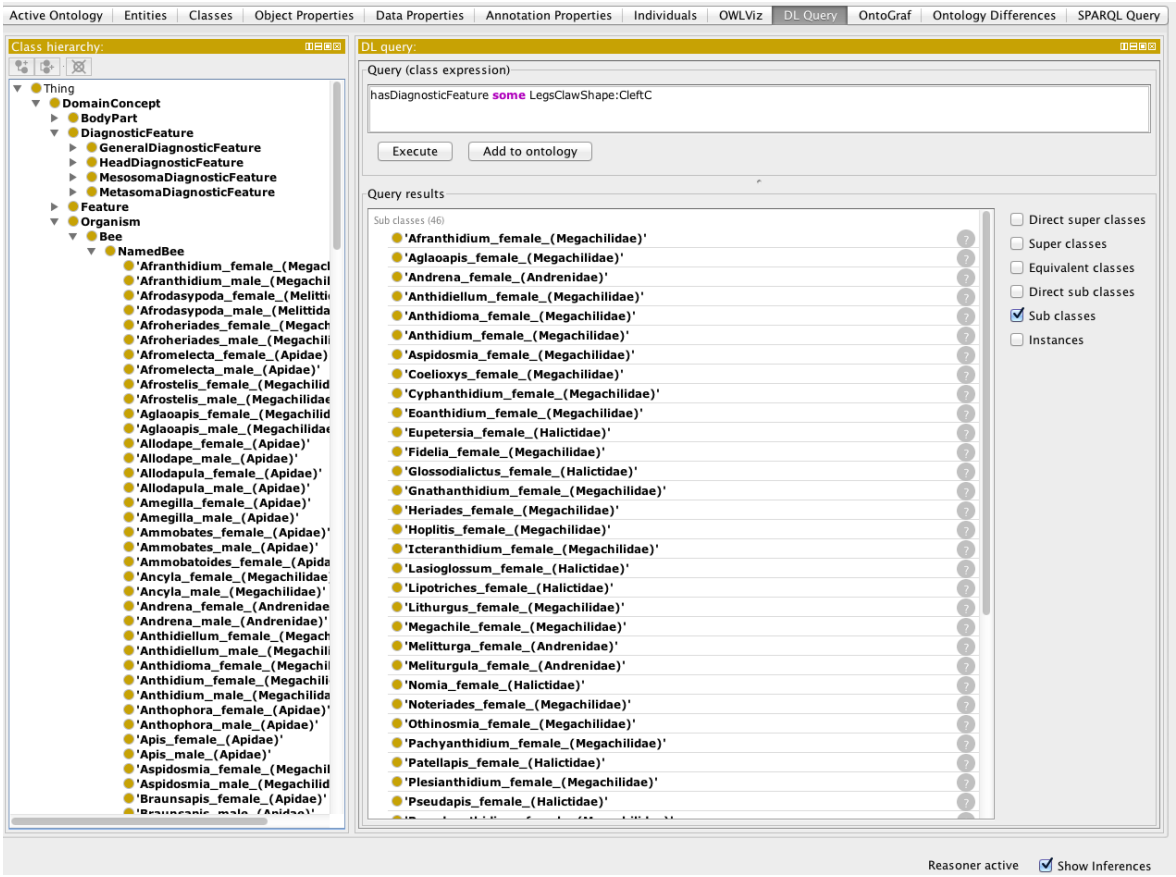


Figure 4.1: Query 4.1 as would be carried out in Protégé.

The reasoner would then return all the species that are related to the given diagnostic feature(s) via the role hasDiagnosticFeature, which are the species that have the selected characteristics. Depending on the diagnostic feature, the axiom can become relatively complex when longer more complex morphological descriptions are used. This concludes the first query that is analysed.

Such classification queries are straightforward to construct or to execute in natural language or in terms of DLs and standard reasoning. In section 2.1.3 the topic of taxonomic revisions was brought up and discussed. A brief summary is presented of the process and certain aspects related to it before addressing the next question.

Taxonomic revision is the description, identification and/or revision of new species based on a combination of morphological characteristics [Maxted, 1992]. These revisions come about as a result of the discovery of new findings. Working with Dr. Eardley proved to be useful as he highlighted the requirements that would be most beneficial and helpful for him and other taxonomists. One of these requirements was the ability to pose a query to the *opposite effect of a key* or the reverse of Question 4.1. A taxonomic key identifies certain species based upon a number of key diagnostic features that are selected and input (functioning as a multi-access key). The features will have been identified from a taxon and allows the user to identify it based upon its key features. The queries involved with a key would be similar to Question 4.1 and Query 4.1.

When considering taxonomic revisions and implementing the reverse functionality of a key, a few different questions emerged. Essentially a taxon would be analysed visually with a preliminary informed guess being taken of what species it is. Once the prediction is chosen, the question asked would be:

Question 4.2
'What features does this (speculated) species possess?', or, with many taxa being examined, *'what specific features does each species possess, and which are the common shared features between them?'*

These questions are focused on in this research study and in the practical application that forms part of this study. The answers of Question 4.2 will enable the identification of the key diagnostic features of a given (Afrotropical bee) species, or of multiple species. The identification of the key features could assist with taxonomic revisions where a taxonomist has a specimen and believes it to be a specific species (or belongs to a specific genus), and needs to check instantaneously which features the species has or the features it shares with a similar species. Through the reverse functionality, a species' characteristics will be identified in order to establish if it indeed matches up to the taxonomist's belief, allowing instant access to knowledge and saving significant time and effort.

Working with Axiom 4.1 from the ontology, as was shown above,
$\mathsf{BeeSpecies} \sqsubseteq \exists\ \mathsf{hasDiagnosticFeature}.\mathsf{DiagnosticFeature}$,
the unknown now, is the DiagnosticFeature, which needs to be found. Through conventional standard reasoning it is not possible to solve Question 4.2, (of *'what features does this (speculated) species possess?'*) with a query over Axiom 4.1, as was done in the first instance using Query 4.1.

The next section discusses the abilities of the standard reasoning services with regards to the presented questions, as well as some tedious work arounds where the standard reasoning tasks are insufficient. An additional reasoning algorithm is introduced and analysed, which is able to answer Question 4.2 to assist with taxonomic revisions through the execution of queries over Axiom 4.1.

4.2 Extending Standard Reasoning For Classification Queries Over Concepts

The first part of this section focuses on the standard reasoners reasoning abilities, and how they affect the posed queries introduced in the previous section. The queries needed as part of taxonomic revisions cannot be carried out, hence two methods are presented for the remodelling of the ontology in order to allow the standard reasoners to execute all the queries. It is shown that these methods are time consuming and unnecessary thus the need for a better more direct solution.

The last part of the section concludes the chapter by introducing the existential reasoning algorithm that, through the use of its own algorithm in addition to the standard reasoner is able to solve the type of queries presented earlier. The algorithm is analysed and discussed.

4.2.1 Standard Reasoning and Classification Queries

Query 4.1 from the previous section, makes use of Axiom 4.1, and it can be seen that the query is non complex and can be executed by the existing reasoners currently available such as Hermit [Shearer et al., 2008], Pellet [Sirin et al., 2007] or FaCT++ [Tsarkov and Horrocks, 2007, 2006] in order to achieve the desired results. Axiom 4.1 states: BeeSpecies $\sqsubseteq \exists$ hasDiagnosticFeature.DiagnosticFeature, which means that "if some concept or individual in the domain is a Bee Species, then this concept has at least one diagnostic feature". The reasoning task at play when carrying out the query is subsumption testing (discussed in section 3.4), which is when checks are done to ascertain whether a concept is a sub-concept or a super-concept of another [Baader et al., 2003; Krötzsch, 2012].

Moving to the second question, Question 4.2, which asks *'what features do a particular (speculated) species possess?'* As the ontology is modelled, using Axiom 4.1 the standard reasoning services are not able to accommodate this sort of question as a query. There are two ways to enable the query through the manipulation of the modelling of the ontology, which will be pointed out shortly. The remodelling though, can become quite tedious, time consuming and unnecessary. By implementing either of these two alternatives within the ontology, a query for Question 4.2 can be made in a similar manner to Query 4.1. The two methods to remodel the ontology are presented next.

Method 1: Remodelling using inverse roles

The first alternative involves creating a second object property that would function as an *inverse* [Horrocks and Sattler, 1999] of the role hasDiagnosticFeature. It should be noted that only concepts are dealt with in the ontology and not individuals, thus the goal is to make use of these classification queries *over concepts*. The bee species and diagnostic features are all modelled as classes and not individuals. The use of individuals will be presented in the second alternative method shortly. As per Axiom 4.1, BeeSpecies $\sqsubseteq \exists$ hasDiagnosticFeature.DiagnosticFeature, a query is made (Query 4.1), where the DiagnosticFeature(s) is known, to obtain the associated BeeSpecies. Essentially to find the *super class* or *subsumer*.

Knowing the BeeSpecies and using this same axiom to acquire the DiagnosticFeature(s), there is no robust way to implement a query to solve for such. If the query is asked, the reasoner returns *'OWL Nothing'* or $\bot$. A modelling solution to this, is to introduce another object property, isDiagnosticFeatureOf, which is the inverse of hasDiagnosticFeature. Using this new role, an association is made between the diagnostic features and the bee species in a similar manner as Axiom 4.1. The resulting axiom will be:

Axiom 4.2 (Diagnostic Features and Bee Species)
DiagnosticFeature $\sqsubseteq \exists$ isDiagnosticFeatureOf.BeeSpecies

For it to work, every diagnostic feature of a particular species will need to be attached to the bee species and vice versa (and every bee species will need to be attached to their particular diagnostic features). A few examples to illustrate this follow. Examples of Axiom 4.1 are:

- Plesianthidium (Spinanthidiellum) volkmanni (Friese) male $\sqsubseteq \exists$ hasDiagnosticFeature.Male BasitarsusColour:Black
- Plesianthidium (Spinanthidium) trachusiforme (Friese) $\sqsubseteq \exists$ hasDiagnosticFeature.MaleT7 Shape:Tridentate
- Plesianthidium (Spinanthidium) neli (Brauns) $\sqsubseteq \exists$ hasDiagnosticFeature.ClypeusColour:Yellow

Examples of Axiom 4.2 would then be the opposite of the above:

- MaleBasitarsusColour:Black $\sqsubseteq \exists$ isDiagnosticFeatureOf.Plesianthidium (Spinanthidiellum) volkmanni (Friese) male
- MaleT7Shape:Tridentate $\sqsubseteq \exists$ isDiagnosticFeatureOf.Plesianthidium (Spinanthidium) trachusiforme (Friese)
- ClypeusColour:Yellow $\sqsubseteq \exists$ isDiagnosticFeatureOf.Plesianthidium (Spinanthidium) neli (Brauns)

The next step would be to compose a query to execute over Axiom 4.2. The query being:

Query 4.2 (Find Diagnostic Features Query)
$\exists$ isDiagnosticFeatureOf.BeeSpecies (or isDiagnosticFeatureOf some BeeSpecies in Protégé), where BeeSpecies is the species selected by the taxonomist.

The reasoner would then return all the diagnostic features that are related to the given species' via the role isDiagnosticFeatureOf. For instance, the DL Query,
$\exists$ isDiagnosticFeatureOf.Plesianthidium(Spinanthidium)neli(Brauns)
(or written in Protégé, *'isDiagnosticFeatureOf some Plesianthidium(Spinanthidium)neli(Brauns)'*), would return ClypeusColour:Yellow amongst many other diagnostic features of the bee species.

While this method of adding an inverse object property or role does return the correct results that are sought after, in a large ontology such as this it can become a laborious and time consuming process, and could be subject to human error and omissions due to the many axioms that have to be added repetitively. Also, the core features of the ontology environment would not be utilised as well as making proper use of the reasoning capabilities. An ontology

editor such as Protégé does allow one to define an object property as an inverse of another by simply adding this to one of the role properties. Unfortunately this still does not enable a query such as Query 4.2 to execute as needed without the manual inclusion of the axioms given in the method above. It can however work when used with individuals, which is discussed next.

Method 2: Remodelling using individuals

The second alternative method involves the use of individuals and an inverse role [Horrocks and Sattler, 1999] within the ontology. This method can also become long and tedious within a particularly big ontology. Some experiments were done to ascertain what was and was not possible through the use of the individuals with regards to the queries presented earlier.

In an ontology, individuals can be created and made to represent objects in the domain. Individuals can be instances or *types* of classes. These classes can contain a group of individuals. For example, South Africa, Kenya and Canada can be individuals that are types of countries; the class being the country.

The object property, isDiagnosticFeatureOf was created and made the inverse of the role hasDiagnosticFeature by simply adding this to the properties of the hasDiagnosticFeature role as shown in Figure 4.2. The *inverse* property is used when querying individuals.

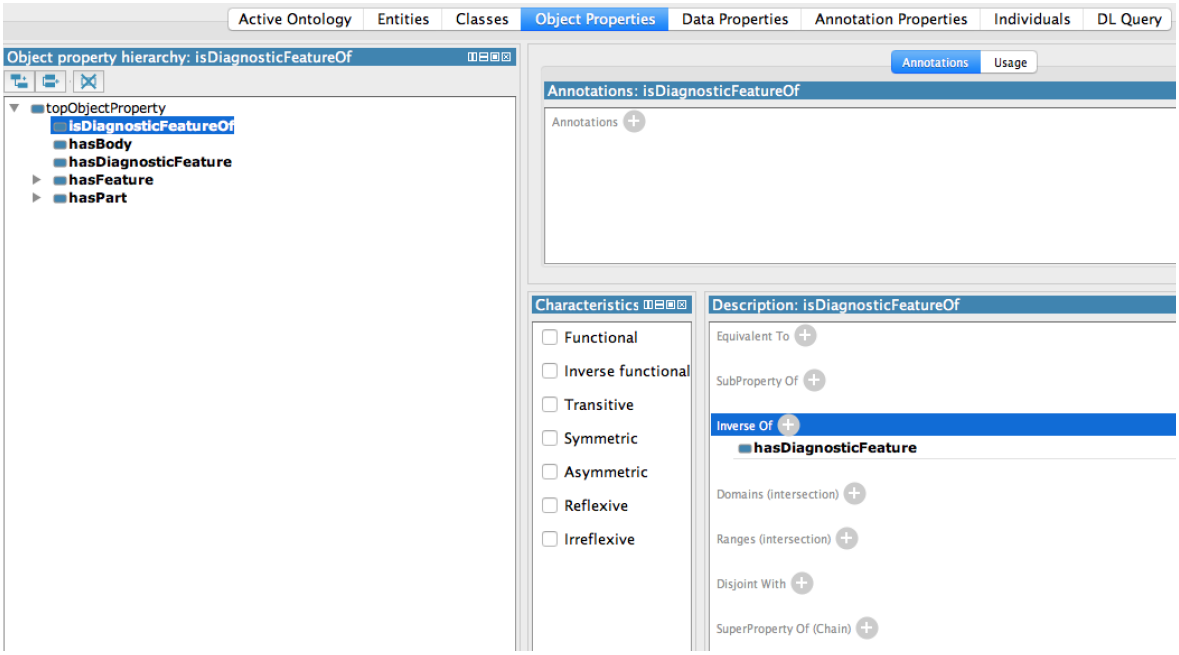


Figure 4.2: Adding an inverse property in Protégé.

Two individuals were created, the first being a type of the diagnostic feature concept, and the second being a type of bee species class. With Axiom 4.1 still in place, Query 4.1 returns the associated bee species concepts and instances, but when Query 4.2 is then run with the inverse object property, $\perp$ is still the result. In order to ensure the desired outcome, an assertion from the class to the individual needs to be made. Queries on individuals (using

value instead of *some* in the query in Protégé) can only be made when an object property assertion has been added between at least one individual. Thus with the assertion BeeSpecies $\sqsubseteq \exists$ hasDiagnosticFeature.$\{DiagnosticFeature\}$ in place, Query 4.2 can be executed and the diagnostic features, as individuals, will be the result.

The query can also be done by implementing relationships between the individuals themselves. It is possible to make an object property assertion from an individual to another individual (or from a class to an individual as above), but not from an individual to a class. An assertion can then be formulated from an individual bee species to an individual diagnostic feature as: hasDiagnosticFeature($BeeSpecies$, $DiagnosticFeature$). Having this axiom in the ontology, the query $\exists$ isDiagnosticFeatureOf.$\{BeeSpecies\}$ or Query 4.2 can be successfully run, returning the corresponding diagnostic features as instances.

In summary, for the required outcome to be achieved in this method, a few things would need to be done:

1. Create corresponding individuals for each and every diagnostic feature, and for each and every bee species.

2. Assert a relationship between the individuals, or between one of the classes and one of the corresponding individuals.

As can be seen, the two remodelling methods can deliver the appropriate output that is needed but are costly in terms of time and effort, more so when dealing with large ontologies. Dr. Horridge [Matthew Horridge] from Stanford University developed an algorithm that also functions as a plugin for Protégé to deal with these types of queries and questions. A detailed analysis of this existential reasoning algorithm and how it manages to answer the queries is presented next.

4.2.2 The Existential Reasoning Algorithm

A reasoning algorithm [Existential Query plugin] that is used as a plugin for Protégé was developed and created by Dr. Matthew Horridge [Matthew Horridge] from Stanford University. The purpose of the algorithm was to allow for queries, similar to Question 4.2, over an ontology. Queries where the *(implied) fillers* of *existential restrictions* need to be found. A standard DL query (such as Query 4.1) can only compute the subclasses or superclasses of the query class expression, but what is needed here are the fillers of a certain kind of class expression. The plugin (which can be found on github[1]) uses the standard reasoners such as Hermit [Shearer et al., 2008], Pellet [Sirin et al., 2007] or FaCT++ [Tsarkov and Horrocks, 2007, 2006] in addition to using its own algorithm to compute the query. Axiom 4.1 can be seen as: BeeSpecies $\sqsubseteq \exists$ hasDiagnosticFeature.DiagnosticFeature, where the (implied) filler would be the DiagnosticFeature, which is exactly what needs to be found in Question 4.2.

The existential reasoner is written in Java™[2] and is compatible with Protégé as a plugin.

[1] https://github.com/protegeproject/existentialquery
[2] https://java.com/en/download/

Figure 4.3 shows the existential query plugin in Protégé. The plugin takes in two arguments namely the class expression, for which the *'filler values'*, are to be retrieved, and the object property. As can be seen in the figure, a bee genus *'Allodape Apidae (Male)'* and the object property hasDiagnosticFeature are input into the plugin in order to obtain the key diagnostic features that belong to the genus via the hasDiagnosticFeature role. The features are found and displayed in the bottom right box. If more than one bee genus or species is included in the query, *'AND'* or *'OR'* can be used between each class to return the intersection or conjunction respectively, of the features related to the set of bees. A chain of object properties separated by commas can also be done for cases where multiple roles need to be queried for.

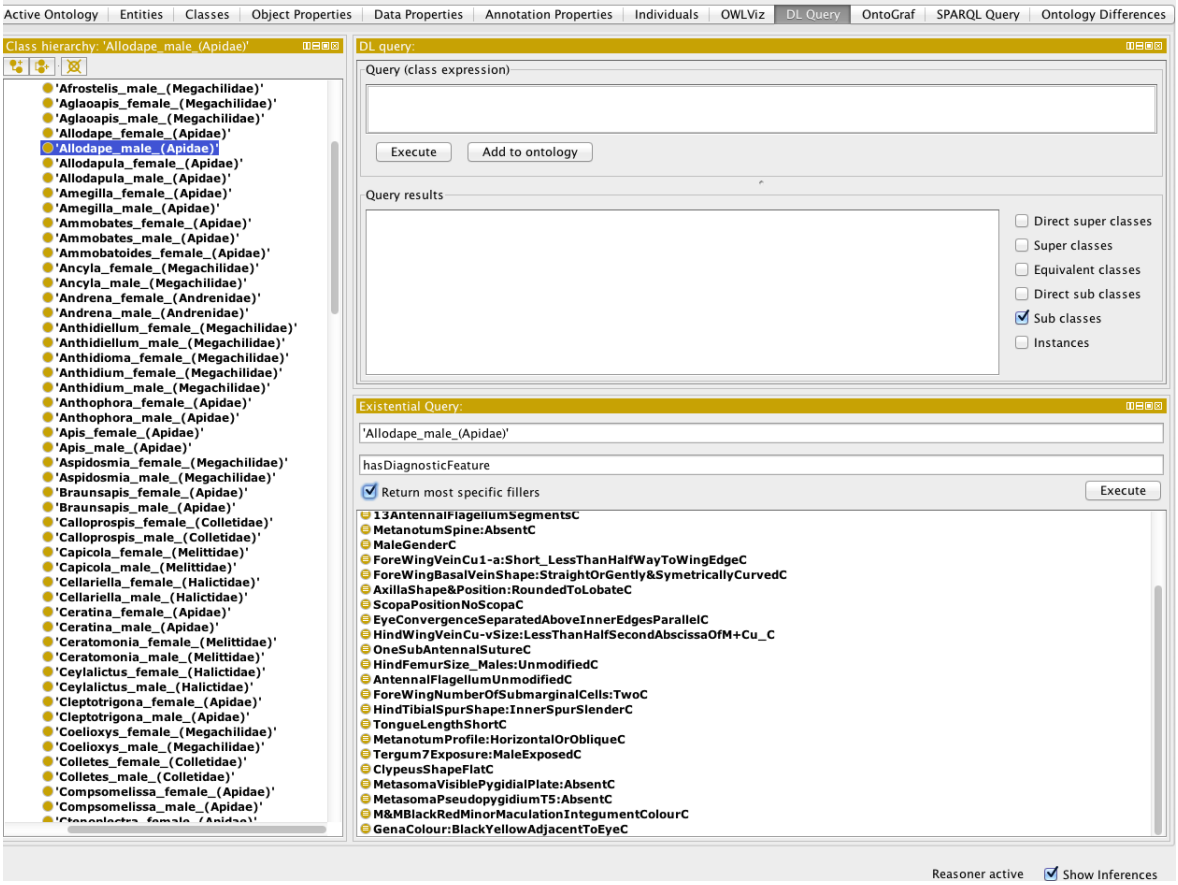


Figure 4.3: The Existential Query Plugin in use in Protégé.

An explanation is now given on how the algorithm works.

Essentially a series of entailment tests are performed using the class hierarchy to optimise the process. The class hierarchy is computed through subsumption testing by one of the standard reasoners.

Given an initial base class C and a property R the algorithm checks to see if $C \sqsubseteq \exists\ R.Thing$ is entailed. If it is entailed, iterations are made over the subclass of $Thing$. For each subclass, Cn, a check is then done to see if $C \sqsubseteq \exists\ R.Cn$ is entailed. If it is entailed the process is repeated with the subclasses of Cn. Once all the entailments have been checked, the *fillers*

in the restrictions of entailed axioms (Cn), are collected and returned as the result.

In an instance using Afrotropical bees the DiagnosticFeature is the filler that needs to be collected.

In summary, it can be seen that the use of the existential reasoning algorithm can be used ideally in large classification ontologies. The two alternative methods described earlier can work, though in a large ontology such as the Afrotropical bee ontology, it can become a tedious time consuming process, and could be subject to human error and omissions due to the many axioms and/or individuals that have to be added repetitively. The ontology structure may also have to change and be modified to have to conform in line with the two alternative methods, which may not be ideal in most situations when the ontology modelling has been done in a certain strategic way for the specific domain. The existential reasoning algorithm can be reused with other classification ontologies through the application developed as part of this study, giving the ability to query the ontologies in such a way that could previously not be done. This allows a whole new set of information to be retrieved and analysed by the users.

The application that has been developed as part of this research study is discussed in the next chapter in Section 5.3, along with the *Afrotropical bee ontology* that is used in the programme. Also, a few examples are presented to show how the existential reasoning algorithm is applied to the domain and to prove that it can be used in such an application and solve the classification queries that have been discussed, i.e. being able to identify the key features of a given Afrotropical bee species or multiple species using Axiom 4.1 in the ontology. The functionality of the application and ontology is a starting point for taxonomic applications that can assist with identifying and classifying species, as well as with providing support for the taxonomic revision procedure.

Chapter 5

Implementation

Introduction

This chapter discusses the modelling and development of an ontology that documents Afrotropical bee data. The purpose of the ontology was to capture the semantics of the Afrotropical bee information in order to classify the bee species according to a set of specific diagnostic features. Also discussed in the chapter is the implementation of a Web Ontology Classifier application (WOC). The WOC was developed as part of this research study primarily to function as an interface that utilises the ontology and the OWL reasoning capabilities to allow classification queries to be executed over the bee ontology data.

The first section of the chapter presents the primary data that was used to construct the ontology, as well as the ontology itself including the modelling and structure of it.

Section two presents a brief introduction to the Java™ based OWL API [OWL API; Horridge and Bechhofer, 2011] that is used by the WOC to give access to, reason over and query the ontology. In the final part of the chapter a full description of the implementation of the WOC is given, with images showing its various functions. A few examples are also provided to illustrate how the existential reasoning algorithm is used in the application.

5.1 The Afrotropical Bee Ontology

This section is made up of two main aspects. Firstly, information with regards to where the taxonomic data for the Afrotropical bees came from initially, and the various formats and structures that they are composed of is presented. The next part of the section focuses on how that data was translated into an ontology, and an in depth explanation of the concepts, structure and modelling of the ontology is provided.

5.1.1 The Key Data

Two fundamental publications were used to provide the data for the ontology, namely *The Catalogue of Afrotropical bees (Hymenoptera: Apoidea: Apiformes)* [Eardley and Urban, 2010] and the booklet, *The Bee Genera and Subgenera of Sub-Saharan Africa* [Eardley et al., 2010]. Both provide extensive taxonomic information on the valid names, nomenclatorial history of, and published references to the known bees of Sub-Saharan Africa and the Western Indian Ocean Islands, excluding the honey bee (Apis mellifera Linnaeus).

The Catalogue of Afrotropical bees (Hymenoptera: Apoidea: Apiformes) [Eardley and Urban, 2010] contains the species with references as well as taxonomic changes such as new name combinations, with correct latinisation and gender. The catalogue also provides the distribution of species by country, plants visited, hosts (for parasitic bee species) and parasites, as well as the type's gender, depository and country locality for each of the described species.

The booklet, The Bee Genera and Subgenera of Sub-Saharan Africa [Eardley et al., 2010] consists of crucial information pertaining to bees as well as biodiversity conservation. The booklet gives a brief introduction and overview of topics such as ecosystem functioning and management, the purpose of pollination and the significant role bees play in it, as well as a look into the motivation as to why there should be an interest in the conservation of bees. The main goal for the booklet [Eardley et al., 2010] though, is to aid novices with the identification of bees, as it serves to function as a key for the bee genera and subgenera that occur in Sub-Saharan Africa. The species and genera descriptions with regards to morphology, distribution and behaviour are typically translated into such a key that allows users to identify specific bee species or genera. In other words the key consists of the morphological descriptions of the species and genera. An example of such a key is given in Figure 5.1, which is an excerpt from the key in the booklet [Eardley et al., 2010]. In addition, as seen in Figure 5.2, the booklet also provides labelled pictures, defines certain terminology, and gives insight on how to collect bees as well as how to prepare and store specimens.

Data from the keys in the catalogue and booklet were used to create an Excel spreadsheet (depicted in Figure 5.3) in order to create the Lucid Key that is available on the Web[1]. Lucid [Lucid; Norton et al., 2012] was discussed in Section 2.1.2.

The spreadsheet consists of two main aspects. Firstly the actual bee genera or species and secondly, the key characteristics or diagnostic features that make up and describe the bee genus (or species). As can be seen in Figure 5.3 the first column is composed of the diagnostic features and the top row is composed of the Afrotropical bee genera. A specific bee genus is uniquely described or identified by a set of diagnostic features. Each bee genus is associated with a specific diagnostic feature by a *'1'* value in the bee genus column of the spreadsheet, indicating that that particular key characteristic is present on the genus. The *'0'* value indicates that it is absent, and the *'6'* value is ambiguous and is not yet defined for that genus.

To demonstrate, in Figure 5.3 the first genus Afrodasypoda female (Melittidae) in column AY, is used as an example. The associated diagnostic features of the genus, are indicated by a *'1'*. Starting from the top, the first being General:Gender (female/male):Female, 12 antennal segments, six metasomal terga, the second being General:Scopa (pollen basket) position:Hind leg, not corbicula and so on until all the diagnostic features unique to this bee genus have been related to it.

[1] http://africanpollination.org/Africanbeegenera/Key_to_african_bee_genera.html

Key to the Afrotropical bee families

1. Short-tongued (Labial palp with four similar segments) (Fig. 7D) 2
1'. Long-tongued (Labial palp with basal two segments long, apical two segments short (Fig. 7C) 5

2. Glossa bifid apically . . . . . **Colletidae**
2'. Glossa pointed apically 3

3. Two subantennal sutures . . . . . **Andrenidae**
3'. One subantennal suture 4

4. Basal vein distinctly curved . . . . . **Halictidae**
4'. Basal vein straight or nearly straight . . . . . **Melittidae**

5. Female scopa on ventral surface of metasoma, except cleptoparasitic species; male metasoma curled under distally; labrum longer than wide; mostly two submarginal cells (*Fidelia* with three submarginal cells) . . . . . **Megachilidae**
5'. Female scopa on hind leg, except cleptoparasitic species; male metasoma more or less straight; one, two or three submarginal cells; labrum mostly wider than long . . . . . **Apidae**

Figure 5.1: An excerpt from the key in the booklet by Eardley et al. [2010].

General:Scopa (pollen basket) position:Hind leg, not corbicula says that, the scopa (i.e. a pollen basket) is positioned on the bee's hind leg.

Initially only the bee genera was dealt with, with a few specific bee species being introduced at a later stage. With the addition of more species, additional spreadsheets for each Afrotopical bee genus and their associated species were created. An example of such a spreadsheet for the genus *Plesianthidium* can be seen in Figure 5.4. The spreadsheet in Figure 5.4 differs slightly from Figure 5.3 in that the diagnostic features are listed in the top row and the species are listed in the first column. There were no *'1s'*, *'0s'* or *'6s'* used in this spreadsheet. Instead, letters or symbols representing the specific diagnostic features were used to associate the features to the corresponding bee species. This spreadsheet was put together from classification keys similar to what can be found in the booklet, The Bee Genera and Subgenera of Sub-Saharan Africa [Eardley et al., 2010].

These spreadsheets collectively were used as references and as the basis when constructing and modelling the Afrotopical bee ontology.

Next, the analysis of the ontology that was constructed from the above spreadsheets is discussed.

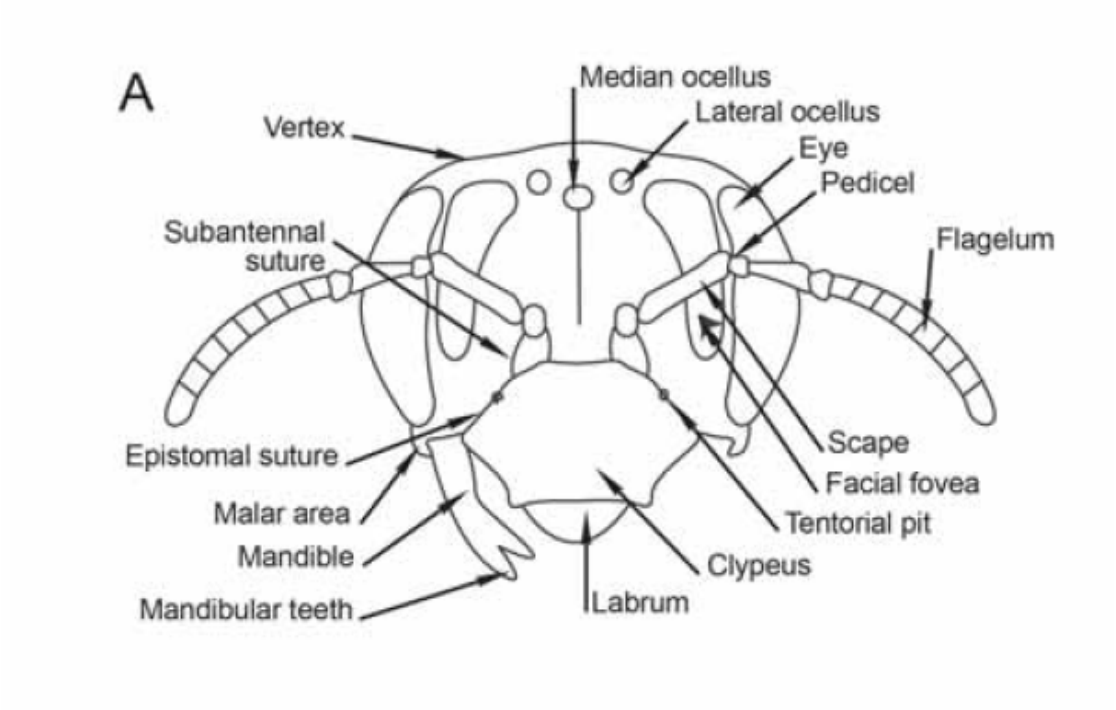


Figure 5.2: Example of a detailed image with terminology depicted in the booklet by Eardley et al. [2010].

	A	AY	AZ	BA	BB
1		Afrodasypoda female (Melittidae)	Afrodasypoda male (Melittidae)	Samba female (Melittidae)	Samba male (Melittidae)
2	General:Gender (female / male):Female, 12 antennal segments, six metasomal terga	1	0	1	0
3	General:Gender (female / male):Male, 13 antennal segments, except Pasites, seven metasomal terga	0	1	0	1
4	General:Scopa (pollen basket) position:Hind leg, not corbicula	1	0	1	0
5	General:Scopa (pollen basket) position:Hind leg, corbicula	0	0	0	0
6	General:Scopa (pollen basket) position:Metasoma, ventral surface	0	0	0	0
7	General:Scopa (pollen basket) position:Metasoma, sides	**0**	**0**	**0**	**0**
8	General:Scopa (pollen basket) position:No scopa	**0**	1	**0**	1
9	General:Integument colour, mesosoma & metasoma:Black &/or red, sometimes minor maculation	1	1	1	1
10	General:Integument colour, mesosoma & metasoma:Black, terga pale apical or subapical crossbands	0	0	0	0
11	General:Integument colour, mesosoma & metasoma:Black, extensive yellow maculation	0	0	0	0
12	General:Integument colour, mesosoma & metasoma:Metallic, mesosoma and/or metasoma	**0**	**0**	**0**	**0**
13	Head:Tongue type:Short tongue	1	1	1	1
14	Head:Tongue type:Long tongue	0	0	0	0
15	Head:Tongue (glossa) apex:Pointed	1	1	1	1
16	Head:Tongue (glossa) apex:Bifid	0	0	0	0
17	Head:Tongue (glossa) length:Short, less than one third prementum length	1	1	1	1
18	Head:Tongue (glossa) length:Long, at least half third prementum length	0	0	0	0
19	Head:Subantennal sutures:One suture, below each antennal socket	1	1	1	1
20	Head:Subantennal sutures:Two sutures, below each antennal socket	0	0	0	0
21	Head:Juxtantennal carina:Present	6	6	6	6
22	Head:Juxtantennal carina:Absent	6	6	6	6
23	Head:Antennal flagellum (male):Unmodified	6	1	6	1
24	Head:Antennal flagellum (male):Modified distally, flattened or extended	6	0	6	0
25	Head:Antennal flagellum (male):Long, over twice as long as eye	6	0	6	0
26	Head:Antennal flagellum, number of segments (male):12	**6**	**0**	**6**	**0**
27	Head:Antennal flagellum, number of segments (male):13	6	1	6	1
28	Head:Eye convergence:Well separated above, inner edges often about parallel	1	1	1	1
29	Head:Eye convergence:Converge above, not touching	0	0	0	0

Figure 5.3: Example of the first Excel spreadsheet made for a number of Afrotropical Bee Genera.

5.1.2 The Ontology

This section presents all the different facets of the Afrotropical bee ontology. A description of the general make up of the ontology is put forward, along with a few stats about it. Next,

	A	B	T	U	V	W	X	Y
1	Plesianthidium	Gender	T2-T3, with sparse distal fascia[x], without distal fascia[z]	T2-T5, with sparse distal fasciae[x], without distal fascia[o]	Gonostylus shape, curved outwards distally[x], weakly concave distolaterally [o], narrows subapically [z], concave apicolaterally [c]	T6, with small tubercle posteromedially[x], withOUT small tubercle posteromedially[z], with large posteromedially tubercle[§]	S6, with obtuse posterolateral angle and posteromedian angle[x], posterior edge forming obtuse-angle[z]	S5 posterior edge shape, shallowly concave[x], distinctly concave[o], posterior edge not distinctly concave, pointed posteromedially[z]
2	*Plesianthidium (Spinanthidiellum) volkmanni* (Friese)	female	x					
3	*Plesianthidium (Spinanthidiellum) volkmanni* (Friese)	male		o	z	x		
4	*Plesianthidium (Spinanthidiellum) rufocaudatum* (Friese)	female	x					
5	*Plesianthidium (Spinanthidiellum) rufocaudatum* (Friese)	male		o	z	x	x	
6	*Plesianthidium (Carinthanthidium) cariniventris* (Friese)	female						
7	*Plesianthidium (Carinthanthidium) cariniventris* (Friese)	male			x			
8	*Plesianthidium (Plesianthidium) fulvopilosum* Cameron	female	x					
9	*Plesianthidium (Plesianthidium) fulvopilosum* Cameron	male		x	o			
10	*Plesianthidium (Spinanthidium) richtersveldi* sp.n	female	z					
11	*Plesianthidium (Spinanthidium) richtersveldi* sp.n	male		o	c	x		
12	*Plesianthidium (Spinanthidium) callescens* (Cockerell)	female	z					
13	*Plesianthidium (Spinanthidium) callescens* (Cockerell)	male		o	c	§		o
14	*Plesianthidium (Spinanthidium) trachusiforme* (Friese)	female	z					
15	*Plesianthidium (Spinanthidium) trachusiforme* (Friese)	male		o	c	§		x
16	*Plesianthidium (Spinanthidium) bruneipes* (Friese)	female	x					
17	*Plesianthidium (Spinanthidium) bruneipes* (Friese)	male		x	c	x	z	z
18	*Plesianthidium (Spinanthidium) aff. calvini* sp.n	female						

Figure 5.4: Example of the spreadsheet made for the genus, Plesianthidium.

the hierarchical structure and modelling of the ontology is discussed. The rest of the section provides a description and examples of each of the concepts and object properties that make up the ontology and the relationships that exist in it.

The core concept in a taxonomic key for Afrotropical bees is the diagnostic feature, which is the unique description represented in the first column of the spreadsheet in Figure 5.3. These descriptions, more specifically describe the morphological characteristics of the body parts of the bee species and can become quite complex. Since ontologies provide semantics and structure to a domain and they are well suited to capturing such qualitative data and descriptions, it was investigated how such morphological key data could be captured in a formalised ontology.

The ontology was built using Protégé [Protégé] and queried using one of the standard packaged reasoners, the FaCT++ [Tsarkov and Horrocks, 2007, 2006] reasoner as well as the existential reasoner by Dr. Matthew Horridge [Existential Query plugin]. The ontology was modelled and constructed using the basic approach by Horridge and McGuinness [Horridge et al., 2009; Noy et al., 2001]. An essential part of modelling the ontology that was kept in mind, was to keep a pragmatic approach in order to allow the domain expert to understand and participate and to keep it as close to the existing domain data's structure as possible. It was essential to work closely with Dr. Eardley to understand the domain and the requirements as well as to keep it as similar to the actual taxonomy as possible. Some concepts and modelling ideas were borrowed from the HAO [HAO; Yoder et al., 2010b], but the key data used was provided from the Excel spreadsheets shown previously.

A few key statistics and metrics of the ontology are listed below:

- Annotation Assertion axioms count: 2087
- Axioms: 10426

- Logical axiom count: 7129
- Class count: 1182
- Object property count: 21
- DL Expressivity: $\mathcal{ALCHQ}$

It can be seen from the figures that it is a relatively large ontology with over 10 000 axioms and over 1000 classes. An analysis of the modelling and make up of the ontology is given next.

A bees body is made up of body parts that can be attached to other body parts, and so on, and these body parts have certain distinguishing morphological features. There are particular body parts with specific features that make up the diagnostic features, which can be grouped together with other diagnostic features to uniquely identify a bee genus/species. In other words a set of diagnostic features uniquely describes a specific bee genus/species.

For example, the diagnostic feature: 'Head:Tongue (glossa) length:Short, less than one third prementum length' broken down, is the *tongue* (that is a body part, attached to another body part, the head) *length* (which is the morphological feature of the part) that is *'Short, less than one third prementum length'*.

The diagnostic feature concept was modelled as a body part (that can be part of other body parts), which has a feature such as a colour or shape. It is this defining *'feature - bodypart'* combination, (depicted as a diagnostic feature in the ontology) that is used to classify and identify the bee genus/species. To relate the body part concepts and the feature concepts, an object property hasFeature (and its sub properties) was used. hasFeature is introduced along with other object properties used within the ontology near the end of the section.

The concept hierarchy of the ontology and more specifically the diagnostic feature class, derived from the Excel spreadsheet, is presented in Figure 5.5 (seen in the left panel in Protégé) consisting of GeneralDiagnosticFeature, HeadDiagnosticFeature and so on. This concept taxonomy is the hierarchy asserted by the ontology engineer, but the 'inferred' class hierarchy can also be shown when a reasoner has been invoked and inferences made, thus possibly revising the asserted taxonomy. Any concepts inconsistent with the asserted hierarchy would be highlighted in red.

As discussed, a diagnostic feature is a body part (body parts can be attached to other body parts) that is associated with some feature. Each of these concepts needs to be defined in order to model the ontology and the diagnostic features. The hierarchy, the different classes and the relationships between them are presented next.

Body Parts

The body parts of the Afrotropical bees are sub classes of the BodyPart class. These sub classes consist of:

- Appendage

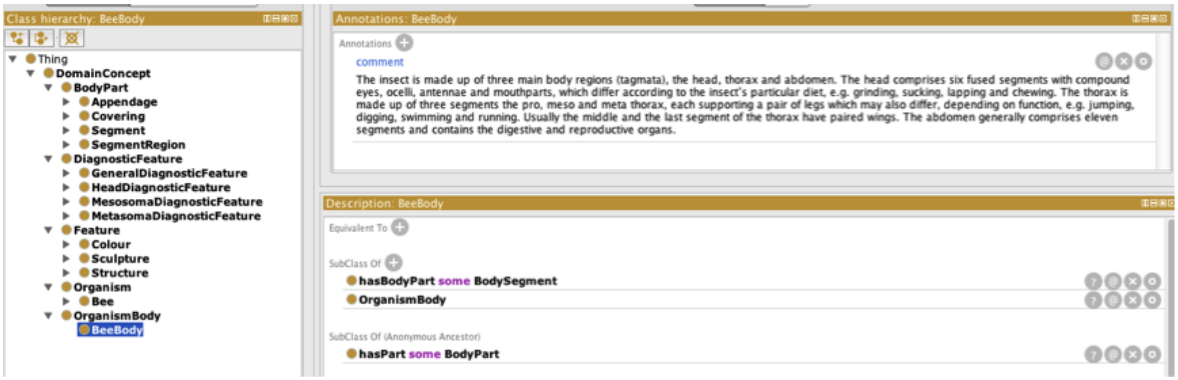


Figure 5.5: The Afrotropical bee ontology hierarchy shown in Protégé.

- Covering
- Segment
- Segment region

In terms of DLs, Appendage $\sqsubseteq$ BodyPart, Covering $\sqsubseteq$ BodyPart and so on.

The body parts make up the different parts of the bees body and consist of further sub classes. For example SegmentRegion is further divided up into the different segment regions of the bees body, such as :

- Antenall region
- Head region
- Leg region and so on.

The actual body parts associated with the specific region are subclasses of the various regions. For example, the subclasses of HeadRegion are the body parts: Eye, Clypeus, Mandible and so on, all residing on the head of a bee. Thus Eye $\sqsubseteq$ HeadRegion and Clypeus $\sqsubseteq$ HeadRegion.

Morphological Features

The morphological features used to describe the body parts are sub classes of the Feature class. These sub classes consist of:

- Colour
- Sculpture
- Structure

Each general feature has a set of more specific subclasses, for example under the Structure class, exists:

- Length
- Shape

- Size
- Punctation and so on.

Even further, within the Length class for example, there are classes such as: GenaLength, MesopleuronLength, or TongueLength. TongueLength for instance, can be either short or long (short, being less than one third prementum length and long being at least half third prementum length).

Bee Concepts

The bee genera and species are represented as concepts in the ontology. A specific genus, *Plesianthidium* and a few of its species can be seen in Figure 5.6 in the left panel of Protégé.

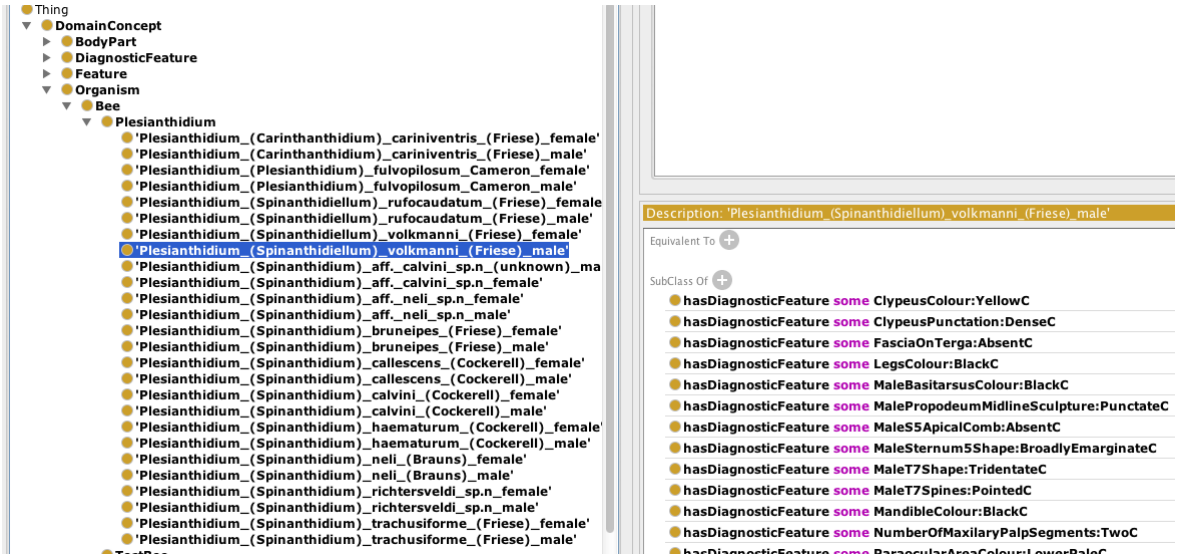


Figure 5.6: The species 'Volkmanni (Friese) male' of the Plesianthidium genus, with its associated diagnostic features.

Diagnostic Features

The core concept, the diagnostic features consist of:

- General Diagnostic Feature
- Head Diagnostic Feature
- Mesosoma Diagnostic Feature
- Metasoma Diagnostic Feature

The sets of diagnostic features are related to their respective parts, i.e. to the head or mesosoma. For example, under HeadDiagnosticFeature subclasses such as EyeSizeFeatures, ClypeusColourFeatures or ClypeusPunctationFeatures exist and so on. Taking EyeSizeFeatures for instance, the subclasses are Enlarged Eyes, or Unmodified (Normal size) Eyes., which are the two sizes an eye can be.

The diagnostic features are a combination of body parts and a feature, related to each other using the object property hasFeature or one of its sub properties, which is discussed next.

Thus Diagnostic Feature ≡ Body Part ⊓ ∃ hasFeature.Feature

Annotations were used in the ontology to document the modelling choices of the concepts. In addition the annotations of the diagnostic features are used for display purposes in the application, thus they are aimed to be kept as close to natural language as possible for easier recognition by the taxonomists or users.

Object Properties

A few of the object properties that are used within the ontology, are highlighted here.

hasFeature or one of its sub properties are used in defining the diagnostic features. The sub properties of hasFeature consist of:

- hasLength
- hasColour
- hasSize
- hasPunctation and so on.

Figure 5.7 shows the hierarchy of all the object properties used within the ontology. A simple example is the *eye size* diagnostic feature that was already introduced, where EnlargedEyes is a an *eye that is enlarged* or an eye that has a size of 'EnlargedEyeSize'. This is then equivalent to:

Eye and (hasSize some EnlargedEyeSize) (Protégé syntax).
or EnlargedEye ≡ Eye ⊓ ∃ hasSize.EnlargedEyeSize

In this instance EnlargedEye is the diagnostic feature, Eye is the body part, hasSize is the object property (the sub property of hasFeature) and EnlargedEyeSize is the feature. Altogether they constitute the diagnostic feature.

This principle is applied to all the diagnostic features, some becoming more complex when more body parts are involved. Another example can be seen in Figure 5.8, which shows the equivalence axiom (in the bottom right panel) of the *clypeus punctation* diagnostic feature.

The role hasPart consists of a sub property hasBodypart, which in turn includes sub properties such as:

- hasAntenna
- hasMetasomaTergum and so on.

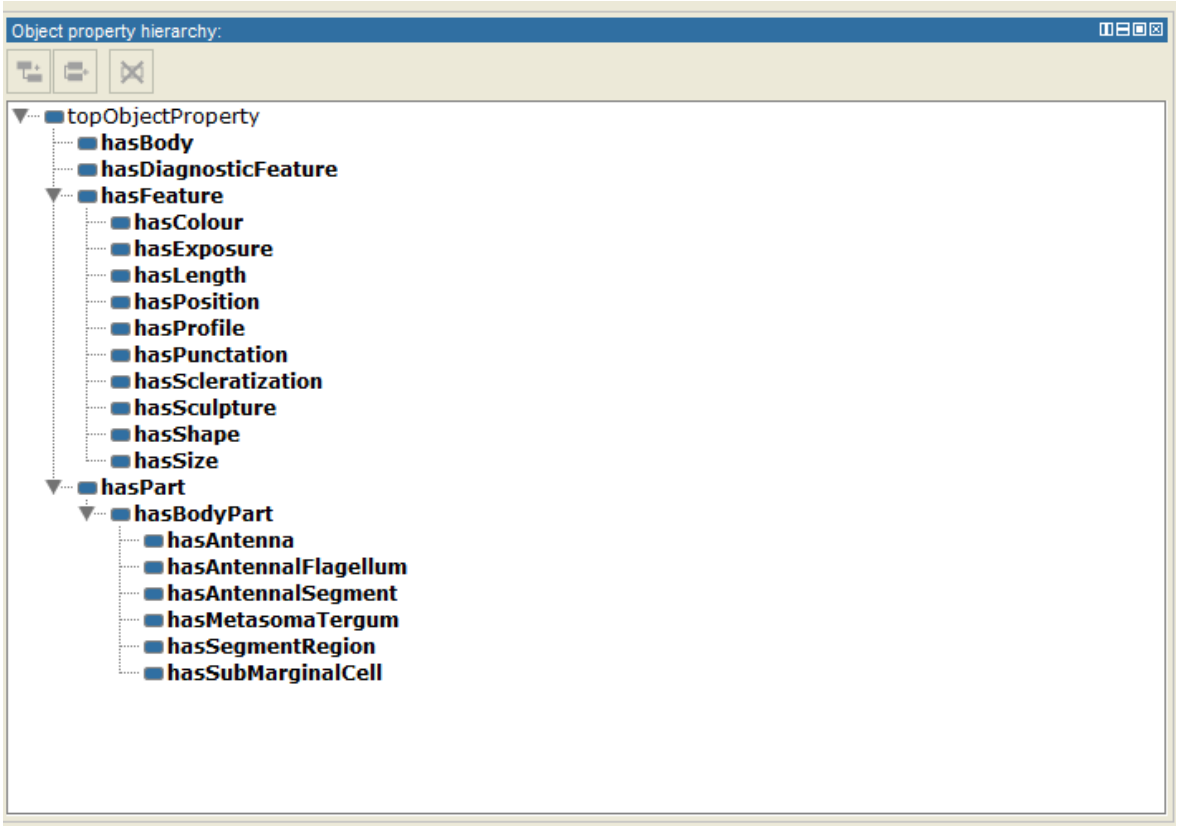


Figure 5.7: The object property hierarchy shown in Protégé.

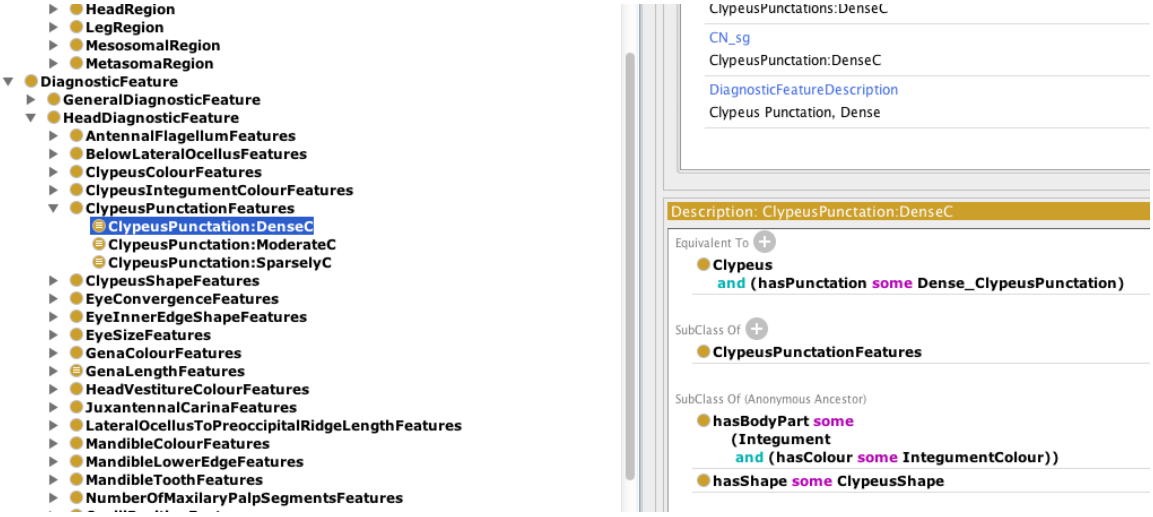


Figure 5.8: The modelling of the Clypeus Punctation diagnostic feature shown in Protégé.

The object property hasBodypart and its sub properties are used to relate body parts to other body parts or to more specific parts such as the antenna in the case of hasAntenna.

The object property hasDiagnosticFeature is the object property that relates the bee species concepts to their specific sets of diagnostic features. For example a species that has *enlarged eyes*, will look like this in Protégé:

Bee Species SubClass Of hasDiagnosticFeature some EnlargedEyes.
or in DL syntax, Bee Species $\sqsubseteq \exists$ hasDiagnosticFeature.EnlargedEyes

This axiom will be amongst other similar axioms for the bee species, relating other diagnostic features. It can be seen more clearly in Figure 5.6, in the bottom right panel of Protégé, where the set of diagnostic features of the bee species, Plesianthidium (Spinanthidiellum) volkmanni (Friese) male are shown.

Given the Protégé metrics, the current Afrotropical bee ontology has more than 1 000 classes, and more than 10 000 axioms. The expressivity is $\mathcal{ALCHQ}$, and the classification time with FaCT++ given a Macbook Pro platform with 8G RAM is a little over 60 seconds.

Extension of the Ontology Structure

The ontology proves useful in representing the taxonomic data, but is a work in progress and it is acknowledged that the structure and model will continue to be refined as more information is added. The ontology structure can be reused by other domains, typically but not limited to, taxonomic domains that are interested in some sort of classification. Holistically speaking the ontology is not as complex as other ontologies out there, as it was kept as close to the taxonomist's data structure as possible for pragmatic reasons. For this reason the ontology can be used as a good base to begin initially, and can be gradually modified if needs be, to suit the particular domain. Another feature of using and adapting this ontology structure for any other domain, is that the WOC application (developed as part of this research study, presented in Section 5.3) can be used in conjunction with it. The ontology structure would need to be consistent with our ontology and conform to the basic structure with certain *'hooks'*. These hooks are the hasDiagnosticFeature role and the DiagnosticFeature concept, which is the super class of all the key features. As long as the ontology contains the hierarchical structure with both hooks, and the objects are related to the key features via the hasDiagnosticFeature role, the WOC application can be used with the ontology.

The next section summarises the OWL API, which is used to manipulate, reason, and subsequently query an ontology.

5.2 The OWL API

The OWL API [OWL API; Horridge and Bechhofer, 2011] (the latest version is version 4.0.1 as of writing) is a Java™ based Application Programming Interface (API) that was developed for creating, editing and managing OWL ontologies. The API consists of a number of interfaces that enable the manipulation of, and reasoning over OWL ontologies. The OWLOntologyManager interface is a key part of the API and provides the means for loading, creating, and editing the ontologies as well as enables access to all the axioms in an ontology. This means that through the OWLOntologyManager, an ontology can be manipulated and tasks such as the addition or removal of axioms can be carried out. Within our application, the WOC, the OWLOntologyManager was used to gain access to the OWLReasoner interface for the loading and classification of an ontology. The OWLReasoner interface was needed to

provide access to the inference services of the Fact Plus Plus reasoner, which is needed for running queries over the ontology. The OWLReasoner interface is responsible for giving access to the standard reasoning methods offered by many of the DL reasoners currently available, for example, ontology classification and consistency checking. The existential reasoner also makes use of the OWLReasoner interface in order to access the FaCT++ reasoner [Tsarkov and Horrocks, 2007, 2006].

In the following section the implementation of the Web Ontology Application (WOC), which makes use of the OWL API to query the Afrotropical bee ontology, is discussed.

5.3 Implementation of The Web Ontology Classifier Application (WOC)

The implementation of an application that was developed as a classification tool or an *ontology based taxonomic key*, is presented in this section. The beginning of the section introduces the software and the aims of it as well as the steps needed to enable the successful installation and functioning of the application. A full description is then given of the tool and covers aspects such as, instructions on how it works, the layout of the various interfaces and the types of queries involved. A number of examples are also illustrated to show how Dr. Horridge's existential reasoner is used within the application.

The section and chapter concludes with a small part on further work to make the tool more efficient and effective for taxonomists.

The WOC was first developed as a proof of concept that was a smaller stand alone Java™ application, which only functioned as a multi-access classification key enabling the classification and identification of Afrotropical bee genera by user selected diagnostic features. In other words, a user could make a selection of a set of diagnostic features and the application would then display the associated bee genera which possess such features. The prototype application did the classification using an ontology and the DL reasoning abilities.

The final software application was developed as a web based application to make it more accessible and to allow it to reach a larger audience. Another aspect that was added was the ability to run additional queries to specifically cater for and assist with taxonomic revisions. The added functionality is to execute a reverse query to a key (without changing the ontology) on the same axioms that were used initially. The reverse query functionality allows a user to make a selection of a set of bee species, and the application would display the corresponding common, *uncommon* and remaining diagnostic features of the selected species. The exact details of the queries were presented in section 4.1.2. As discussed, with the ontology modelled in the way that it is, the existing standard reasoning services are not able to perform reverse queries and deliver the required results necessary for taxonomic revisions. Thus the *existential reasoning algorithm* (covered in section 4.2) is used in addition to the classical reasoning tasks to enable such functionality. A few examples will be demonstrated with illustrations to show how the application functions.

The WOC assists users with taxonomic revisions. Taxonomists do not have to memorise all the unique features of a specific bee species, and can easily check the various features common to a group of species that they are currently studying. Quick checks as needed are made easier and to compare two or more different species to each other can be done rather quickly as well.

With regards to the implementation and running of the WOC there are a few simple prerequisite steps that need to be done, which are:

- Java™[2] needs to be installed.
- A .dll (or .jnilib for MAC) file needs to be copied to a directory on your computer.

The .dll file is to allow the FaCT++ reasoner to be used on your system. The file is easily downloadable from *http://owl.man.ac.uk/factplusplus/* or *https://code.google.com/p/factplusplus/* where the user will need to click on the *'Google Drive'* link and then select the appropriate zip folder for their particular operating system. Once the file is downloaded, it will need to be moved to a destination folder depending on the operating system.

For Mac users, the file *libFaCTPlusPlusJNI.jnilib* needs to be placed in the directory: *System/Library/Java/Extensions*.

For Windows users, the file *FaCTPlusPlusJNI.dll* needs to be placed in the directory: *C:/Windows/System32/*.

Once the above steps have all been done the application should function properly. The source code as well as the ontology and instructions to setup the application offline can be found on github[3]. At the time of writing, the application has not been made available online as of yet, as provisions for the hosting of it on a server are still being made.

Instructions on how to use the tool along with a few screenshots and examples are presented next.

5.3.1 WOC Application and Interface

The WOC application consists of 3 pages. The first page being the home page as can be seen in Figure 5.9, which has basic instructions on how to use the application. This page can be accessed by clicking the *'Home'* link in the menu bar.

By clicking the next menu bar link, *'Classify and Query,'* the user will be directed to an upload page presented in Figure 5.10 where an ontology can be chosen, uploaded and classified.

[2] https://java.com/en/download/
[3] https://github.com/Nish01/OntologyClassifierApplication.git

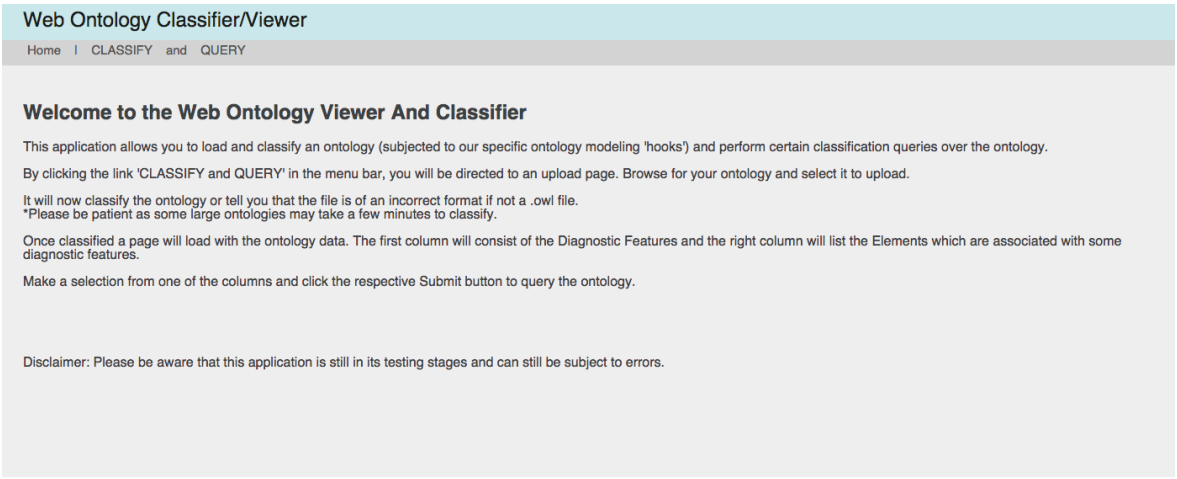


Figure 5.9: The home page of the web ontology classifier.

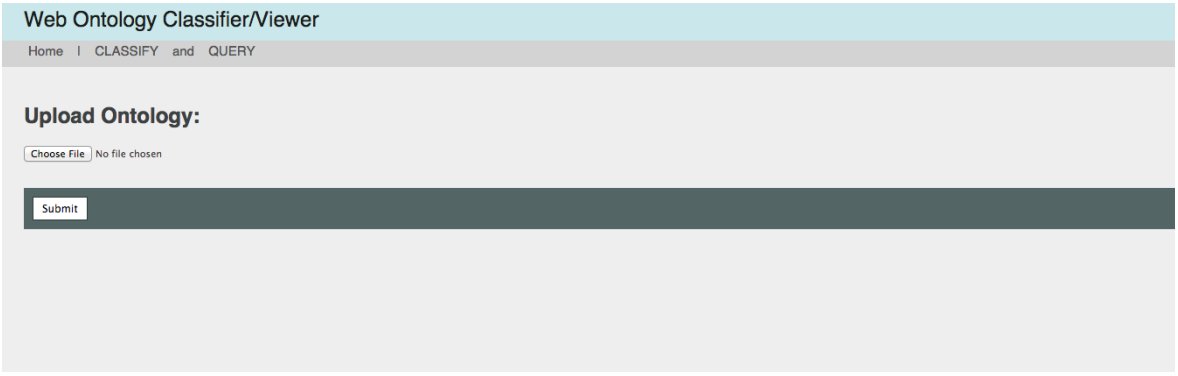


Figure 5.10: The ontology upload page of the application.

Upon selecting the ontology and clicking the 'submit' button, the ontology begins to classify and can take a number of seconds depending on the size of the ontology. When using the Afrotropical bee ontology, approximately +- 50 seconds were taken to classify. In order to classify the ontology the FaCT++ reasoner is used and makes use of subsumption testing to compute the ontology hierarchy. Once classified the ontology will be displayed on the page shown in Figure 5.11.

The diagnostic features are listed in the first column on the left, and the bee species are listed in the second column on the right. On the far right is a key showing what each colour represents. The user has two options here. He/she can then choose to either select a few diagnostic features from the first column, or, make a selection of bee species from the second list, and then click the *'submit'* button below the respective list. In the first case where a few diagnostic features are selected, the page will be updated and look similar to Figure 5.12.

In the instance in Figure 5.12 two diagnostic features were selected (shown in blue at the top of the left column), and four bee specie elements were found that satisfied the query (also shown in blue at the top of the right column). The elements listed in red in the first column are the diagnostic features that are *common* to all the found bee species. The green elements are the remainder of the diagnostic features possessed by each of the identified bee species, but which are not common to all the found species. In other words the diagnostic features that are

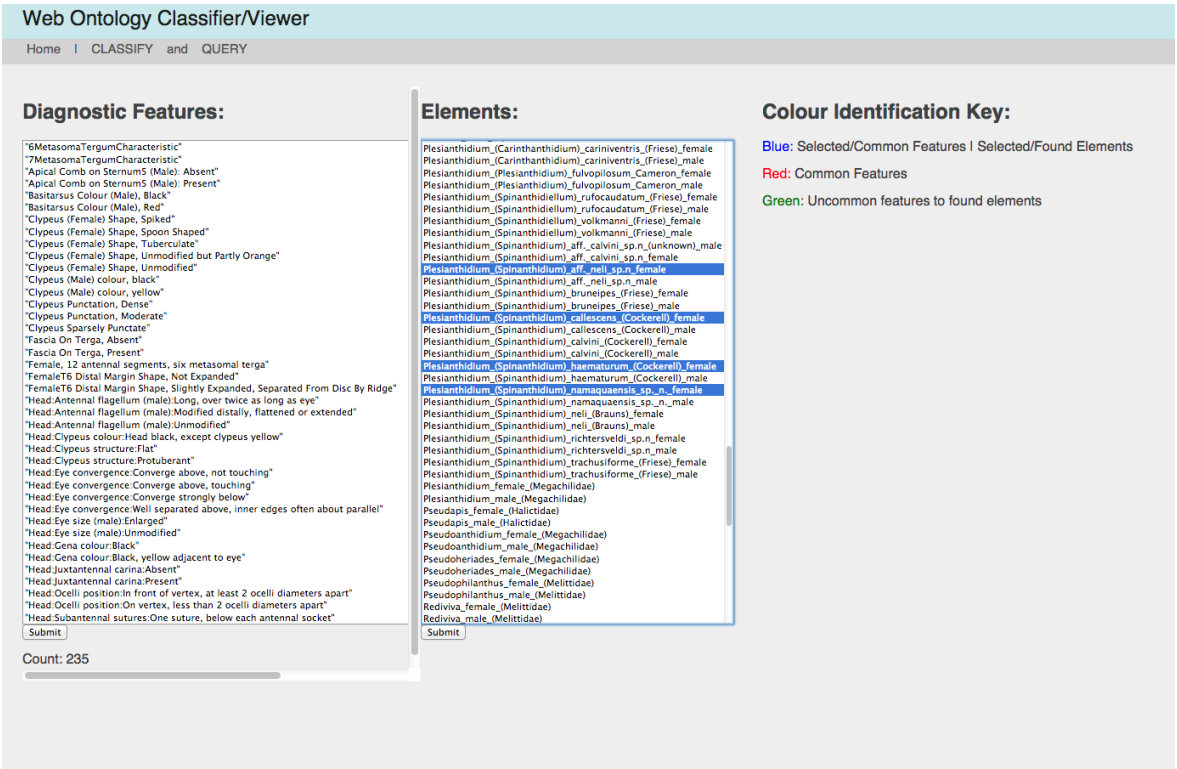


Figure 5.11: The web ontology classifier's main interface with a selection of bee species.

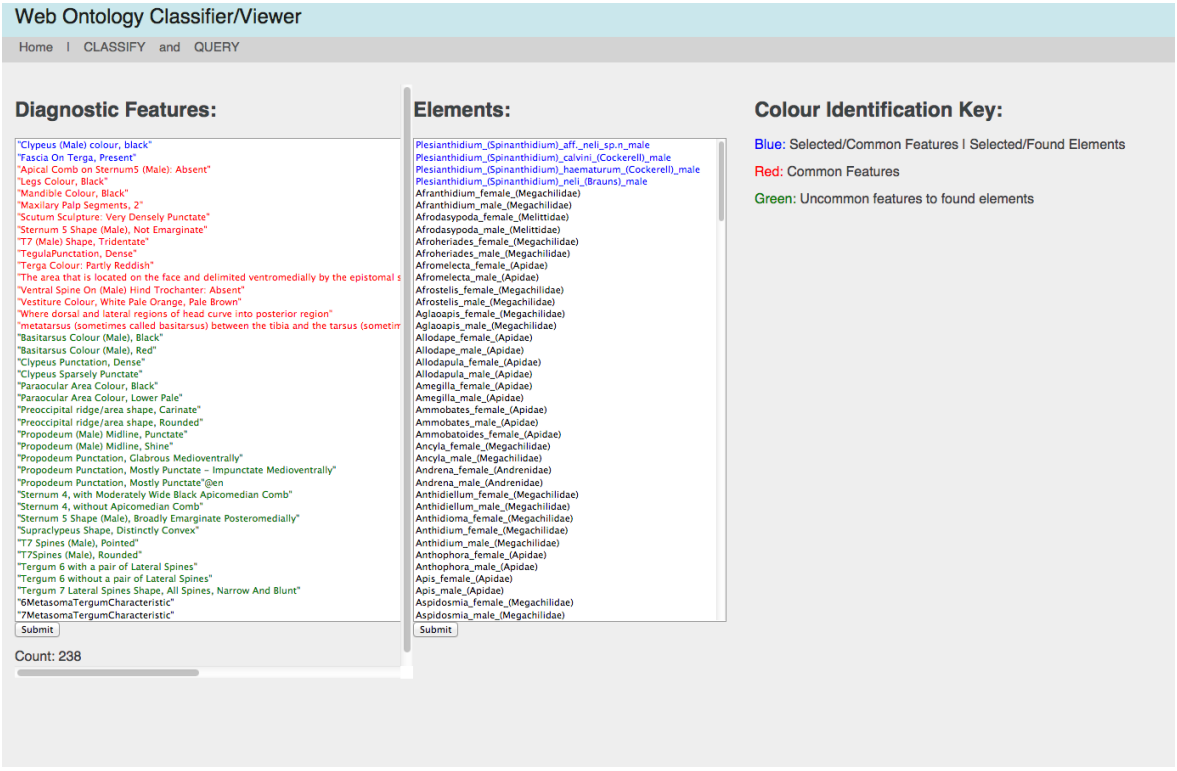


Figure 5.12: The interface showing a query of 2 diagnostic features and the found bee specie elements.

held by *at least one* of the bee species but not shared by the whole group of identified taxa.

FaCT++ utilises a standard reasoning service, subsumption testing to perform the queries and to retrieve the particular bee specie results. The query that is formulated and executed is:

$\exists$ hasDiagnosticFeature.DiagnosticFeature

In other words, the reasoner finds the concepts that are linked to a DiagnosticFeature (which the user has selected) via the object property hasDiagnosticFeature. Thus the bee species are found from the following axiom in the ontology:

BeeSpecies $\sqsubseteq \exists$ hasDiagnosticFeature.DiagnosticFeature

BeeSpecies is subsumed by $\exists$ hasDiagnosticFeature.DiagnosticFeature, thus standard subsumption testing can be carried out here to find the corresponding bee species.

If the user decides to select the second option, i.e. making a selection of bee species from the second list, and clicking the *'submit'* button below the Elements list, the page will then be updated to look similar to Figure 5.13.

In Figure 5.13 it can be seen that two bee species were selected (displayed in blue at the top of the right column) and used in the query. The results are displayed in the first column where the first few diagnostic features in red represent the *common* features (or shared features) of the selected species. The remainder of the results are listed in green, which are the diagnostic features possessed by *at least one* of the selected bee species, but which are not common to *all* the selected species.

For this query subsumption testing alone does not give the required results, which is why the existential reasoning algorithm is used to perform the query along the same axioms used in the first instance. The same axiom being BeeSpecies $\sqsubseteq \exists$ hasDiagnosticFeature.DiagnosticFeature

As depicted in Figure 5.13, the first two bee species, namely, *'Plesianthidium (Spinanthidium) haematurum (Cockerell) male'* and *'Plesianthidium (Spinanthidium) neli (Brauns) male'* were selected and the resulting common features identified were *'Apical Comb on Sternum5 (Male): Absent'*, *'Basitarsus Colour (Male), Black'*, *'Clypeus (Male) colour, black'* and so on, all highlighted in red. These results show the practical use of the existential reasoning in this application, since the axiom that was traversed in the query was:

'Plesianthidium (Spinanthidium) haematurum (Cockerell) male' $\sqsubseteq$
$\exists$ hasDiagnosticFeature.DiagnosticFeature
where DiagnosticFeature is the *role filler* that needed to be found.

In the final part of this section and chapter a few additional features of the application are highlighted, some of which have been included (at a lesser scale) but can be implemented at a further stage. These extra features would allow the programme to be more user friendly,

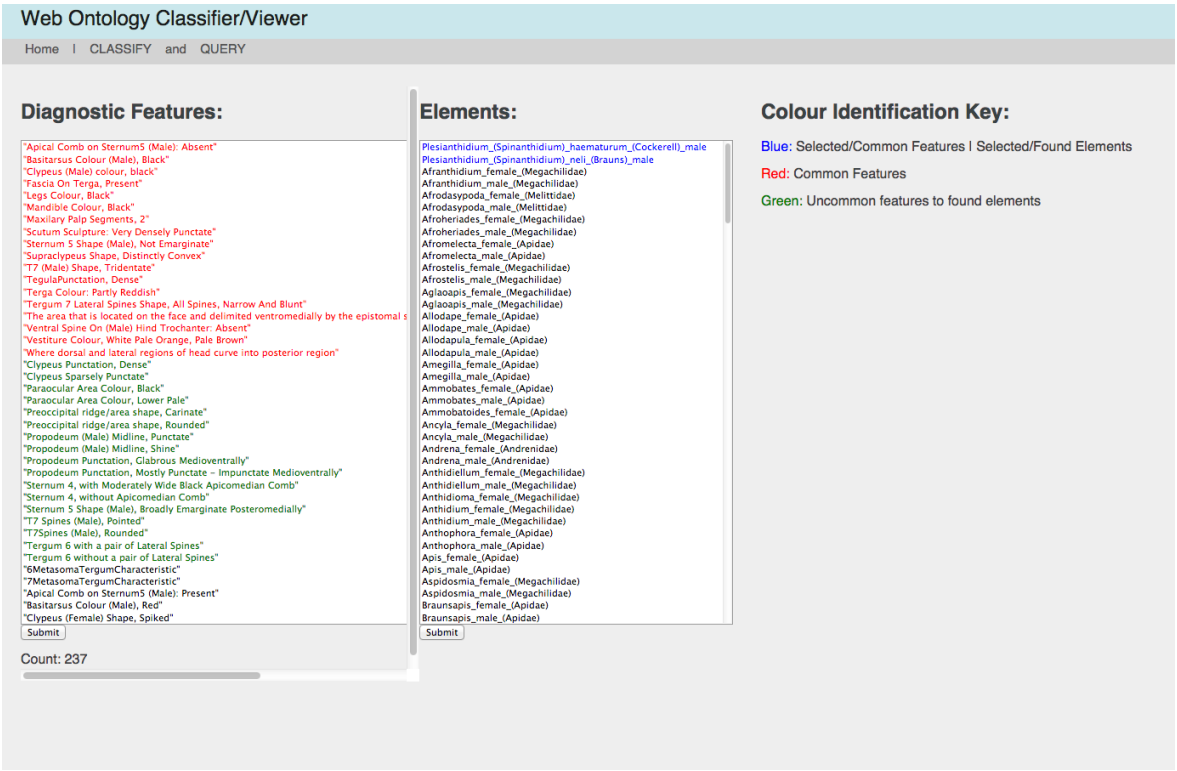


Figure 5.13: The interface showing a query of the diagnostic features of 2 bee specie elements and the found diagnostic features.

efficient and functional.

5.3.2 Further Work and Application Adaptability

A brief look at a few added and extra features is presented, which could be implemented into the application at a later stage to give an idea of the growth and extra functionality that is possible.

It was apparent that the user interface of the WOC would need some improvement to enhance the user friendliness and to make it more appealing, especially for taxonomists who are not conventionally technologically inclined people. The need to reevaluate the interface can be seen from the evaluation in the next chapter. Apart from the interface design, a feature that Dr. Eardley expressed interest in, was including images in the search results of the queries. For example when a set of bee species are found to match the users diagnostic feature selection, illustrations could be displayed to show the particular bee species, or a map showing where the species are predominantly found could be displayed. An example of such images can be seen in Figure 5.14.

Another component that was brought to attention was the ability to *capture* and access a full description of a species. Essentially, the diagnostic features of a species (or genus) make up the description of it. This description defines each species and it's useful for taxonomists if they are able to have simple access to such descriptions for certain documents. In terms of including this in the application, the ability to copy, paste (and edit) the descriptions

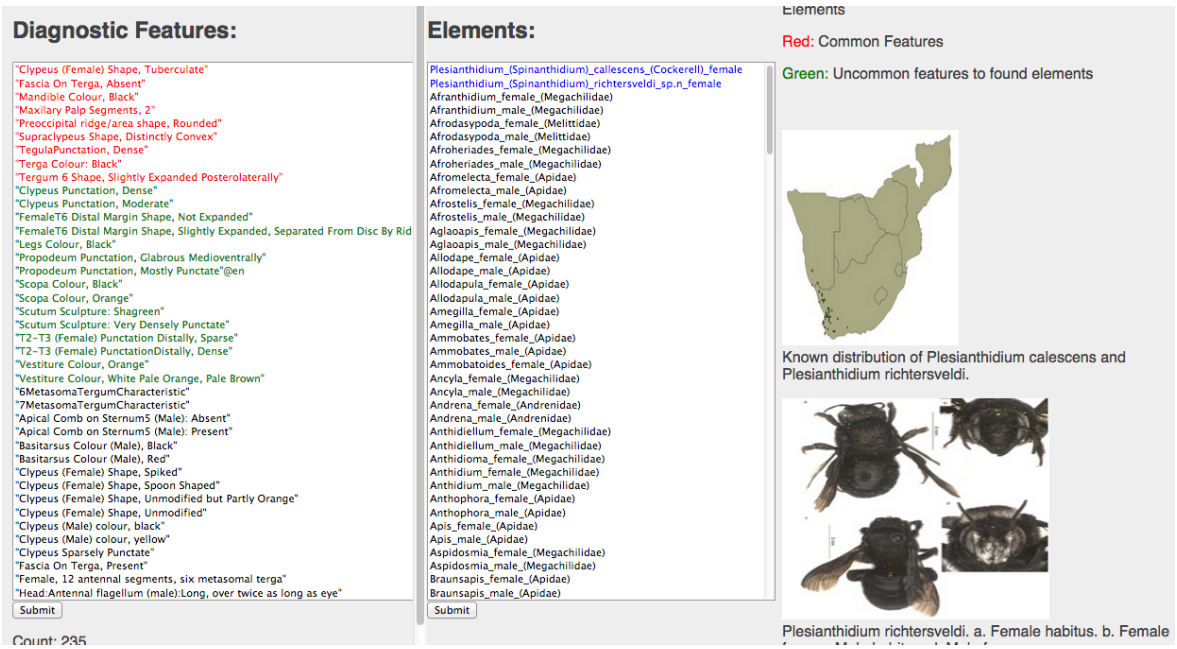


Figure 5.14: The ontology classifier application interface with added associated bee illustrations.

would be advantageous to have. Figure 5.15 shows the interface with a highlighted list of all the diagnostic features that constitute the description of a user selected bee species. That description can then be copied and used elsewhere at the taxonomists disposal.

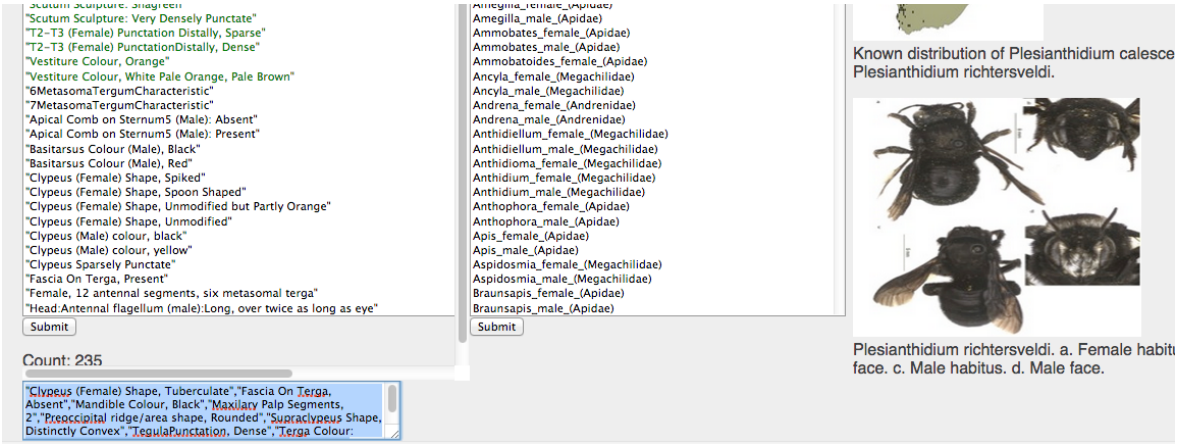


Figure 5.15: The ontology classifier application interface with added species description.

5.3.3 Application Adaptability

With regards to the ontology, it was identified that it would be quite useful if the taxonomists could construct their own ontologies or rather add to existing ontology structures for whichever taxa they be dealing with. Ideally this process should be kept as simple as possible and have a dedicated and simple interface to enable the modification of the ontologies. By doing this the ontologies could be kept up to date and uniform data structures could be implemented

allowing the storage of large amounts of various taxonomic information. These ontologies could then be used in such applications as the WOC.

It should be noted that the WOC can be used within almost any domain where classification of objects is needed. The only requirement is that the constructed ontology would need to conform to a basic structure with certain *'hooks'*. These hooks are the hasDiagnosticFeature role and the DiagnosticFeature concept, which is the super class of all the key features. As long as the ontology contains the hierarchical structure with both hooks, and the objects are related to the key features via the hasDiagnosticFeature role, the application will run smoothly. Thus an ontology can be modelled for the new domain and even adapted from our ontology, to be used with the WOC for that specific domain.

Comparison to Lucid

As has been discussed, Lucid is a commercial software suite of tools that is marketed as powerful and highly flexible knowledge management software applications designed to help users with specimen identification or diagnostic tasks. The WOC is similar to Lucid in a few ways and can provide the same outcome as a Lucid multi-access key. A few differences between the two do exist such as the raw data formats of each and how the data is controlled. The conceptual process of defining a concept in an ontology, and defining a taxon is similar. This is due to the fact that a taxon is defined by specifying all of its unique, defining characteristics and a concept in an ontology is defined and grouped by declaring facts or characteristics about the concept, so that every element that belongs to the concept inherits those characteristics. This is one of the reasons why ontologies are suited to taxonomic information. A significant difference between the WOC and Lucid is the ability to classify and identify the features of a given taxon or multiple taxa, which is useful throughout the taxonomic revision procedure. Since the WOC is a multi-access key that allows the user to choose where exactly he/she would like to start in the classification or identification process, it does not suffer from the drawbacks of a single access key.

The ontology and the WOC can be reused across multiple domains, and can also allow the taxonomists to edit and add to the ontologies (though an interface to do so is part of the future work) where as Lucid can be seen as somewhat fixed in terms of manipulating the raw data by the users.

This chapter detailed the Afrotropical bee data and the ontology that was developed from it. An overview of the WOC application that was developed to manipulate the ontology and assist with taxonomic revisions was also presented. An important aspect of the software was how the application made use of an external reasoning algorithm to complement the standard reasoners in order to execute a particular query. In the following chapter an evaluation is presented by expert taxonomists on the above software in order to identify its usefulness.

Chapter 6

Evaluation

6.1 Application evaluation

Introduction

The WOC application and ontology was evaluated by a bee taxonomist (Dr. Connal Eardley) and was also assessed by a group of taxonomists whom the application was demonstrated to. Valuable feedback was gathered from the taxonomists via a questionnaire, which is summarised and reported further on in this section.

In the first part, an evaluation made by Dr. Eardley is provided where he presents some feedback on the project and how he views it as beneficial to him and to taxonomy. The next section highlights the results received from a demonstration of the ontology and the application, at the Foundational Biodiversity Information Programme (FBIP) Forum where a few taxonomists were given an evaluation form that they completed and returned after the presentation.

6.1.1 Dr. Eardley's Evaluation

Dr. Connal Eardley was asked to use the application. He provided feedback and an insightful evaluation of the project. His full evaluation and comments follow:

Species are usually separated from one another by a combination of characters, each shared with one or more different species in complex networks. Currently revising a group of species, usually all those in a genus, requires remembering each species' unique combination of characters and how these are shared. This is near impossible for groups of more than a few species, which results in the data being recorded on 'paper' (today MSWord or MSExcel). Thereafter the species are described according to these combinations of characters after being sorted after 'on paper' sorting exercises. Finally, identification keys are drafted from the same datasets using the same dated techniques, but only paired combinations are used in each step in keys. Consulting lists of species and characters time and again is very time consuming and a source of error. The process will become more efficient and more accurate when computers are used for this process. Currently taxonomic revisions are always published as hardcopy documents or, more commonly today, as computerised pdf files.

This project was designed to bring order and the ability of repetitive processing (i.e., improved

scientific methodology) to taxonomic research, and to pave the way for the production of interactive taxonomic revisions. It provides:

1. Orderly datasets of species and their full suite of diagnostic characters.
2. Mechanisms to reshuffle the characters according to the species' being compared.
3. Description taken directly from one dataset.
4. Electronic, multi-access identifications key.

None of this exists in one application. Apart from the above, this programme should lead change in the way taxonomic revisions are published setting the stage for users to interrogate datasets and acquire the unique product they need for their purpose, and for the most recent data to be incorporated in the results they produce for themselves.

6.1.2 FBIP Forum Evaluation

The application was demonstrated and presented at the Foundational Biodiversity Information Programme (FBIP) Forum, which is managed by the South African National Biodiversity Institute (SANBI)[1] and is attended by approximately 100 participants from 29 different South African institutions including universities, museums, science councils and other research organisations. A presentation was given, firstly introducing the philosophy of ontologies as well as a bit of background on ontologies and their use within taxonomy. The second part of the presentation was a demonstration of the application where all the functionality was displayed to the audience, using a few examples to show how it could be used. Dr. Eardley also gave a brief talk about how the application (and future continued work in this line) could influence his (taxonomic) work and ultimately apply to and assist the field of taxonomy.

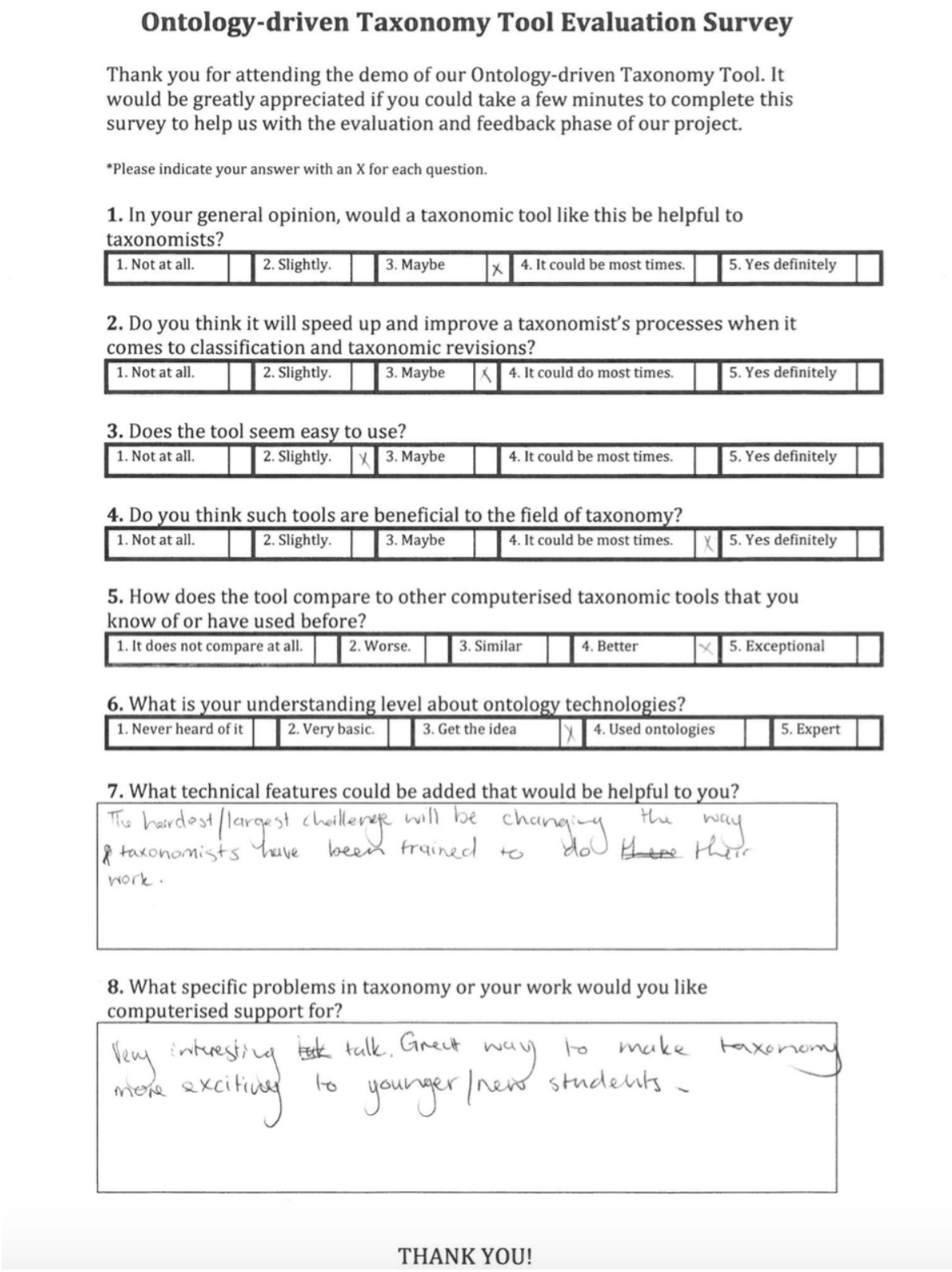
A number of attendees completed an evaluation questionnaire that aimed to distinguish if the application would have a positive effect on their field as well as the field of taxonomy, and if it would be beneficial to them in their jobs. An example of one of the completed forms can be seen in Figure 6.1 and Figure 6.2.

A total of 15 people managed to complete the questionnaires, providing valuable feedback. Going in to the demonstration a preconceived idea of some of the outcomes and criticisms that would most likely be raised was had. The main issues being:

- The user interface would require some work. It would need to be much more user friendly and appealing.
- The taxonomists would like to have more nifty functionality, for instance the ability to connect and display associated images, locations and more.

These preconceived ideas turned out to be accurate as many requests and comments were in line with the above points. A summary of the results is given next.

[1] http://www.sanbi.org/



Ontology-driven Taxonomy Tool Evaluation Survey

Thank you for attending the demo of our Ontology-driven Taxonomy Tool. It would be greatly appreciated if you could take a few minutes to complete this survey to help us with the evaluation and feedback phase of our project.

*Please indicate your answer with an X for each question.

1. In your general opinion, would a taxonomic tool like this be helpful to taxonomists?

1. Not at all.		2. Slightly.		3. Maybe	X	4. It could be most times.		5. Yes definitely	

2. Do you think it will speed up and improve a taxonomist's processes when it comes to classification and taxonomic revisions?

1. Not at all.		2. Slightly.		3. Maybe	X	4. It could do most times.		5. Yes definitely	

3. Does the tool seem easy to use?

1. Not at all.		2. Slightly.	X	3. Maybe		4. It could be most times.		5. Yes definitely	

4. Do you think such tools are beneficial to the field of taxonomy?

1. Not at all.		2. Slightly.		3. Maybe		4. It could be most times.	X	5. Yes definitely	

5. How does the tool compare to other computerised taxonomic tools that you know of or have used before?

1. It does not compare at all.		2. Worse.		3. Similar		4. Better	X	5. Exceptional	

6. What is your understanding level about ontology technologies?

1. Never heard of it		2. Very basic.		3. Get the idea	X	4. Used ontologies		5. Expert	

7. What technical features could be added that would be helpful to you?

The hardest/largest challenge will be changing the way taxonomists have been trained to do ~~there~~ their work.

8. What specific problems in taxonomy or your work would you like computerised support for?

Very interesting ~~talk~~ talk. Great way to make taxonomy more exciting to younger/new students.

THANK YOU!

Figure 6.1: Example of the evaluation questionnaire completed by taxonomists at the Foundational Biodiversity Information Programme (FBIP) Forum

Evaluation Outcomes

As can be seen in Figures 6.1 and 6.2, six multiple choice questions (with a scale rating) were asked. The questions asked were:

1. Would a taxonomic tool like this be helpful to taxonomists?
2. Will it improve and speed up a taxonomist's processes when it comes to classification and taxonomic revisions?
3. Does it seem easy to use?
4. Are such tools beneficial to the field of taxonomy?
5. How does it compare to other computerised tools that you know of or have used before?
6. What is your understanding level of ontologies?

The first two questions were mostly answered positively, meaning that, in their opinion the taxonomic tool will be helpful and can improve the taxonomist's processes. A breakdown of the responses for the two questions are shown in Figure 6.3 and 6.4 respectively.

The third question received some favourable answers, but the general consensus was that while it seemed relatively easy to use the user interface needs some improvement. All of the taxonomists agreed that such tools would assist the field of taxonomy for the next question. Question five was used to see how it faired against other computerised tools that the taxonomists may have used previously. Most responses were that they thought it was similar or better, compared to other applications, but there were a good number of people who had not actually used other computerised applications. The last of the questions was for interests sake, to determine how well versed these taxonomists were with ontologies. After the brief presentation and introduction to ontologies most of them had a fairly decent idea of what they entail, but there were still a few people who were unsure and had never heard of them.

The last two questions asked:

1. What technical features could be added that would be beneficial to you?
2. What specific problems in taxonomy or your work would you like computerised support for?

There were a few different answers and comments made for these last questions. A few of the common observations were that the taxonomists would like to be able to edit and add to the information in the ontology through the application. Additionally the taxonomists would like the interface to be made more user friendly and attractive with more images, and electronic keys (with provision to capture their data) would be of great help to them.

Through these evaluation forms a lot of positive and helpful feedback was received. The response was mostly positive with many people open to using such an application and thinking

that it could be beneficial to their work and to the field of taxonomy and biodiversity in general.

Ontology-driven Taxonomy Tool Evaluation Survey

Thank you for attending the demo of our Ontology-driven Taxonomy Tool. It would be greatly appreciated if you could take a few minutes to complete this survey to help us with the evaluation and feedback phase of our project.

*Please indicate your answer with an X for each question.

1. In your general opinion, would a taxonomic tool like this be helpful to taxonomists?

1. Not at all.		2. Slightly.		3. Maybe		4. It could be most times.		5. Yes definitely	X

2. Do you think it will speed up and improve a taxonomist's processes when it comes to classification and taxonomic revisions?

1. Not at all.		2. Slightly.		3. Maybe		4. It could do most times.		5. Yes definitely	X

3. Does the tool seem easy to use?

1. Not at all.		2. Slightly.		3. Maybe		4. It could be most times.	X	5. Yes definitely	

4. Do you think such tools are beneficial to the field of taxonomy?

1. Not at all.		2. Slightly.		3. Maybe		4. It could be most times.		5. Yes definitely	X

5. How does the tool compare to other computerised taxonomic tools that you know of or have used before?

1. It does not compare at all.		2. Worse.		3. Similar		4. Better	X	5. Exceptional	

6. What is your understanding level about ontology technologies?

1. Never heard of it		2. Very basic.	X	3. Get the idea		4. Used ontologies		5. Expert	

7. What technical features could be added that would be helpful to you?

Scan in descriptions, choose characters common to species in their descriptions. Not necessarily pick out the characters – which is the traditional approach.

8. What specific problems in taxonomy or your work would you like computerised support for?

As in 7.

THANK YOU!

Figure 6.2: Example of the evaluation questionnaire completed by taxonomists at the Foundational Biodiversity Information Programme (FBIP) Forum

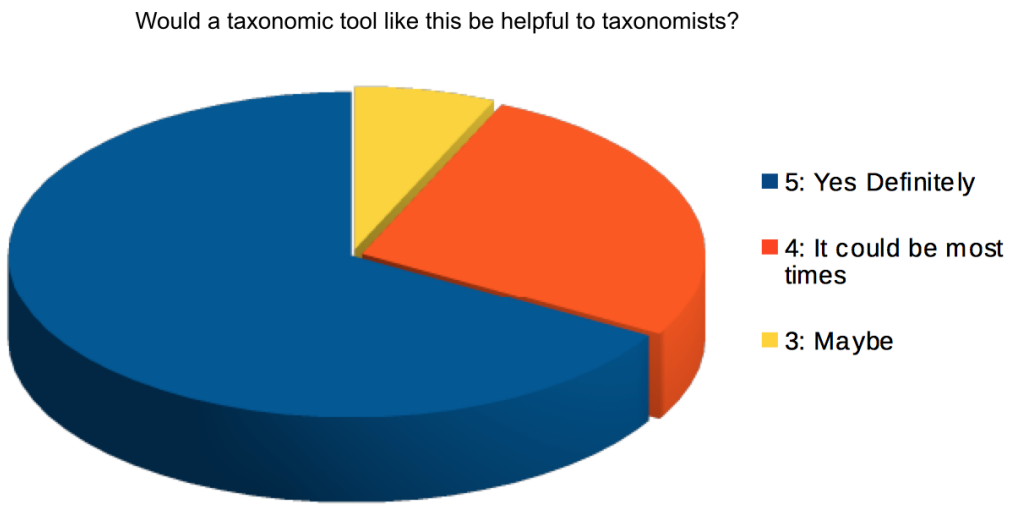


Figure 6.3: Pie chart showing a breakdown of the answers for question 1 of the evaluation form.

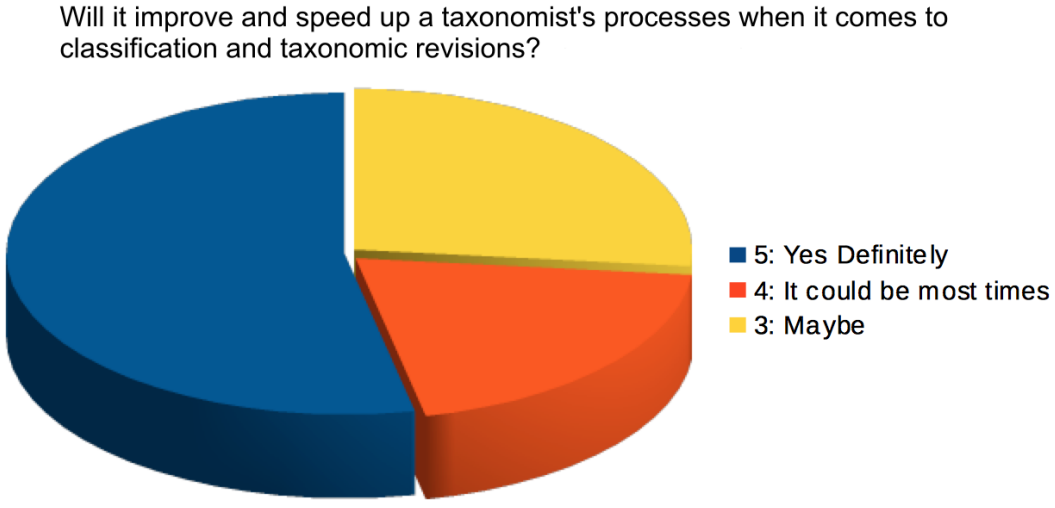


Figure 6.4: Pie chart showing a breakdown of the answers for question 2 of the evaluation form.

Chapter 7

Contribution and Conclusion

In this dissertation a case study of Afrotropical bees was presented as well as research on how to use morphological data about the bees to model an 'Afrotropical bee ontology.' An investigation into how particular classification queries can be answered over an ontology was done throughout the study. Questions such as *'what key features does a particular (speculated) species possess?'* or *'what are the individual as well as common shared features of multiple species?'* were asked and posed as queries over the Afrotropical bee ontology. The queries could not be solved through standard reasoning alone, thus a few solutions to enable the queries were investigated and presented. One of the most viable solutions found, included the use of an *existential reasoning algorithm*, which extends from the standard reasoning services to answer the queries. Part of the investigation was how an ontology-based multi-access taxonomic key application can be developed for Afrotropical bees, which assists taxonomists with the taxonomic procedure, taxonomic revisions. The application uses the created ontology as well as the existential reasoning algorithm to do this.

This chapter concludes the research study and summarises all the findings and contributions of the research. The various contributions are given first with the conclusion and summary of the basic points presented next. Finally an outline of projected work and research in this domain, with regards to the ontology and application is put forward.

7.1 Research outcomes and Contributions

At the start of this research study a few questions were presented in Section 1.4. A recap of these questions will be done with their corresponding answers being addressed, showing the contributions of the study. The main research question and contribution of this study is:

- How can classical reasoning be extended for classification queries over ontologies?

The three sub-questions that make up the primary research question are:

1. How can Afrotropical bee morphological data be modelled into an ontology for classification purposes?
2. How can reverse classification queries over concepts be executed over an ontology given the case of an ontology for Afrotropical bee morphological data?
3. How can a multi-access taxonomic key application be developed for Afrotropical bees that assist taxonomists with taxonomic revisions?

Taxonomic Requirements

Initially, discussions were had with a taxonomist to identify and understand the specific requirements that would support taxonomists. Some of the requirements included:

- A form of storage for all of the taxonomic morphological knowledge, i.e. an ontology.
- A computerised taxonomic key application that uses the stored knowledge.
- A computerised taxonomic key application that functions specifically as a multi-access key.
- An application that allows for the reverse queries that can be used for taxonomic revisions.

Once identified, we were able to support taxonomists better using computer technology since we could understand what taxonomists need in order to do their work.

Question 1. How can Afrotropical bee morphological data be modelled into an ontology for classification purposes?

In Chapter 2 the application domain was introduced and scoped this research paper. The case study covered aspects of biodiversity and taxonomy, becoming more specific with the case of Afrotropical bees. In Chapter 5, it was shown how an Afrotropical bee ontology was modelled and developed to accommodate the morphological data and to allow for classification. The ontology was successfully built and used within an application, although it is a work in progress. Through the ontology a great knowledge base exists that captures not only the Afrotropical bee knowledge, but it is a standardised format or *template* that can be reused by other domains with similar goals. What this means is that the ontology structure can be used to capture other taxonomic knowledge for instance, and allow other fields to make use of the ontology's capabilities, providing more room for growth of the ontology and of the knowledge.

Question 2. How can reverse classification queries over concepts be executed over an ontology given the case of an ontology for Afrotropical bee morphological data?

Chapter 4 discussed the types of *classification questions* and queries that are typically asked in classification tools such as taxonomic keys. The classification questions are questions such as

- *'Which species (or objects) exist that have a group of selected unique features?'*
- *'What key features does a particular (speculated) species possess?'*
- *'What are the individual as well as common shared features of multiple species?'*

The questions were translated into DL queries and performed over the Afrotropical bee ontology. The standard reasoning managed to use subsumption testing to answer the first question (*'Which species (or objects) exist that have a group of selected unique features?'*). The remaining questions could not be answered successfully through the standard reasoning

alone. Thus in order to perform the reverse classification queries utilising the classical reasoners, two solutions were identified. The two solutions involved significant remodelling of the ontology with the addition of many axioms. Though the solutions provided the correct results, the methods were rather time consuming and laborious, especially when dealing with large ontologies.

A final solution was found using an existential reasoning algorithm developed by Dr. Matthew Horridge of Stanford University. The algorithm managed to perform the particular *reverse* classification queries over the existing axioms and concepts in the ontology with no remodelling required. Since no remodelling of the ontology was needed, substantial time and effort was saved in constructing the ontology. The advantage of using the existential reasoner extends to the reuse of the ontology, where the ontological structure can be reused a lot more efficiently without all the remodelling, saving other domains the time and effort of having to consider the many additional axioms. The workings of the existential reasoner are covered in Chapter 4 and a few examples of how the algorithm is used in an application with the Afrotropical bee ontology are illustrated in Chapter 5.

Question 3. How can a multi-access taxonomic key application be developed for Afrotropical bees that assist taxonomists with taxonomic revisions?

Chapter 5 documented the development and the implementation of a web ontology classifier application. The application was developed to function as an ontology-based multi-access key and aid users with identifying and classifying species, as well as with taxonomic revisions. The WOC application assists with taxonomic revisions by answering such questions as *'what key features does a particular (speculated) species possess?'* and *'what are the individual as well as common shared features of multiple species?'* The application is able to answer these questions through the use of an existential reasoning algorithm that enables *reverse* classification queries over concepts. The WOC makes up the main contribution of this study as it incorporates all the different aspects of the research and is ultimately beneficial to users in taxonomy. Through the WOC, classification and taxonomic revisions can be made easier and more efficient, allowing more access to morphological taxonomic knowledge. The application could also be used to support taxonomic repositories such as ZooKeys. An evaluation by a range of taxonomists and an acknowledged bee taxonomist, Dr. Connal Eardley, was presented in Chapter 6 where favourable comments were received on the application. Dr. Eardley believes such a tool can be beneficial to taxonomists and to the field of taxonomy as a whole. Through the evaluation and feedback it was established that the research study can be used as a base with the prospect of having much more functionality added to it.

Though the application and ontology was focused on the Afrotropical bee taxonomy they can be used in almost any other domain where classification is applied. The ontology structure would be kept relatively the same but would need to be modified with the appropriate data and terminology specific for the domain. An example where it has been done tentatively is with (computer) network attacks. In brief, particular *network attacks* have certain characteristics. When dealing with such an attack the characteristics are analysed in order to determine what kind of attack it is. Thus a network attack ontology coupled with the WOC could prove to

be a convenient means to identify the particular virus.

7.2 Research Design

The research questions were answered using an experiment. Initially several requirements for features that support taxonomy were collected from the domain experts. Part of the experiment included the modelling and storing of taxonomic knowledge as an ontology to be used for classification. The Afrotropical bee data was used and modelled as an ontology, using some of the principles and concepts from other ontologies such as the HAO [HAO; Yoder et al., 2010b]. Certain classification questions that are usually asked when classifying organisms and when performing the taxonomic procedure of taxonomic revisions were translated to DL queries to be used over the ontology. The queries necessary to support the taxonomic revision functionality were investigated and it was found that the standard reasoning services were unable to execute the queries, as the required results were not found. Various alternative solutions to enable the successful running of the queries were tested and implemented, including the use of an algorithm that extends from the standard reasoning, which proved to be the most viable solution. The final phase of the experiment included the development of an ontology-based web application, (the WOC) that supports taxonomy and its processes. Essentially the WOC functions as a multi-access taxonomic key that incorporates the various classification queries over the ontology, to support classification and taxonomic revisions.

The research was evaluated using three mechanisms:

1. An investigation into whether the extension of the standard reasoning rendered the required results through the queries over the ontology, which could not be solved using only the standard reasoning. This was done through the modelling of the ontology and the implementation of the WOC with the existential reasoning algorithm. The WOC incorporates the ontology as well as the existential reasoning algorithm, and we found that together the necessary functionality does render the required results that can assist with classification and taxonomic revisions.

2. A detailed usage evaluation by a bee taxonomist. It was found that Dr. Eardley believes the application and research can be beneficial to taxonomists and to the field of taxonomy as a whole.

3. A demonstration and questionnaire presented to several taxonomists was used to receive feedback on the usefulness and functionality of the approach. Again, favourable feedback was received on the research as well as key improvement areas.

7.3 Conclusion

The various questions that were introduced at the start of this dissertation have been addressed throughout the research. The primary objective of the research study was to solve the question, 'how can classical reasoning be extended for classification queries over ontologies?' The question was solved through a non standard 'existential reasoning algorithm' that was shown to enable the classification queries over an Afrotropical bee ontology and deliver the

required results. The algorithm and the ontology were combined formally in the WOC application to show that the queries can be answered and used by taxonomists. The research addressed the sub-questions that made up the main research question, resulting in the creation of an ontology that documents the morphological data of Afrotropical bee species. Also developed was the application that utilises the Afrotropical bee ontology and enables classification queries over the ontology. It was found that the WOC can assist taxonomists with the classification and identification of Afrotropical bee species as well as with taxonomic revisions. It was also shown that the WOC and ontology structure can be re-used within other domains.

7.4 Future Work and Continuous Research

In terms of the ontology that has been developed, it can be expanded or duplicated to add a much bigger range of data and can include data from other sub domains. The ontology itself could also be refined and utilise more research into the optimum modelling of it, as there are many ways it can be structured and arranged, especially when dealing with larger volumes of data.

In Section 5.3.2, a few extra features of the application that could be implemented and added in the future were highlighted. In addition to a more user friendly interface and other nifty features such as adding images, it follows from there that the tool could be improved upon to add more functionality such as the ability to edit and modify the ontology using a simple web interface. Utilising a custom easy to use interface would be better than say Protégé for taxonomists and users who are not well versed in some technical applications. A custom web interface will enable the users to focus specifically on their data and not worry about the more intricate details and the working of Protégé. Another aspect that would be advantageous is to have the tool hosted on a sufficient web server to allow access to users anywhere in the world.

Bibliography

Apollo. [online] Available at: http://apollo.open.ac.uk/.

Baader, F. and Sattler, U. An overview of tableau algorithms for description logics. *Studia Logica*, 69(1):5–40, 2001.

Baader, F., Buchheit, M., and Hollander, B. Cardinality restrictions on concepts. *Artificial Intelligence*, 88(1):195–213, 1996.

Baader, F., Calvanese, D., McGuinness, D., Nardi, D., and Patel-Schneider, P. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge university press, 2003.

Bard, J. and Rhee, S. Ontologies in biology: design, applications and future challenges. *Nature Reviews Genetics*, 5(3):213–222, 2004.

Beisswanger, E., Schulz, S., Stenzhorn, H., and Hahn, U. Biotop: an upper domain ontology for the life sciences. *Applied Ontology*, 3(4):205–212, 2008.

Bennett, B. Foundations for an ontology of environment and habitat. In *FOIS*, pages 31–44, 2010.

Berners-Lee, T. The world-wide web. *Computer Networks and ISDN Systems*, 25(4): 454–459, 1992.

Berners-Lee, T., Hendler, J., Lassila, O., et al. The semantic web. *Scientific american*, 284 (5):28–37, 2001.

Biodiversity Brief. Education for sustainable development information brief: Biodiversity. [online] Available at: http://www.unesco.org/education/tlsf/extras/img/DESDbriefBiodiversity.pdf [Accessed 21 March 2014].

Boero, F. Light after dark: the partnership for enhancing expertise in taxonomy. *Trends in Ecology and Evolution*, 16(5):266–267, 2001.

Brachman, R. What is-a is and isn't: An analysis of taxonomic links in semantic networks. *Computer;(United States)*, 16(10), 1983.

Brachman, R. J. and Levesque, H. Expressiveness and tractability in knowledge representation and reasoning. *Computational intelligence*, 3(2):78–93, 1987.

Braun, S., Schmidt, A., Walter, A., Nagypal, G., and Zacharias, V. Ontology maturing: a collaborative web 2.0 approach to ontology engineering. *Ckc*, 273, 2007.

Calvanese, D., Giacomo, G. D., Lembo, D., Lenzerini, M., Poggi, A., and Rosati, R. Ontology-based database access. pages 324–331. In Proceedings of the 15th Italian Conference on Database Systems (SEBD 2007), 2007.

CBD: Linnaeus Lecture Series. Linnaeus: The Father of Sustainable Development. [online] Available at: http://www.cbd.int/doc/publications/linnaeus-brochure-en.pdf [Accessed 13 March 2015], 2007.

Chavan, V., Rane, N., Watve, A., and Ruggiero, M. Resolving taxonmic discrepancies: Role of electronic catalogues of known organisms. *Biodiversity informatics*, 2, 2005.

CO-ODE Project. [online] Available at: http://owl.cs.manchester.ac.uk/research/co-ode/.

Collins, A., Joseph, D., and Bielaczyc, K. Design research: Theoretical and methodological issues. *The Journal of the learning sciences*, 13(1):15–42, 2004.

Conte, Y. L. and Navajas, M. Climate change: impact on honey bee populations and diseases. *Revue scientifique et technique (International Office of Epizootics)*, (27):485–97, 2008.

Culverhouse, P., Williams, R., Reguera, B., Herry, V., and González-Gil, S. Do experts make mistakes? a comparison of human and machine identification of dinoflagellates. *Marine Ecology Progress Series*, 247(17-25):5, 2003.

Daniel, P. Classification. *The Internet Encyclopedia of Philosophy, ISSN 2161-0002*, 2016.

Davis, P., Heywood, V., et al. Principles of angiosperm taxonomy. Technical report, JSTOR, 1963.

de Queiroz, K. The linnaean hierarchy and the evolutionization of taxonomy, with emphasis on the problem of nomenclature. *Aliso*, 15(2):125–144, 1997.

Deans, A., Seltmann, K., Yoder, M., Miko, I., Forshage, M., Bertone, M., Agosti, D., Austin, A., Balhoff, J., Borowiec, M., et al. A hymenopterists' guide to the hymenoptera anatomy ontology: utility, clarification, and future directions. *Journal of Hymenoptera Research*, 27:67, 2012.

Disney, H. Hands-on taxonomy, 2000.

du Plessis, L., Škunca, N., and Dessimoz, C. The what, where, how and why of gene ontology? a primer for bioinformaticians. *Briefings in Bioinformatics*, pages 723–735, 2011.

Eardley, C. Afrotropical bees now: what next? *Pollinating bees: The Conservation Link between Agriculture and Nature. Ministry of the Environment, Brazil. Brasilia, DF*, pages 105–114, 2002.

Eardley, C. A taxonomic revision of the southern african species of the subgenus creightonella cockerell (apoidea: Megachilidae: Megachile latreille). *Zootaxa*, 3159:1–35, 2012.

Eardley, C. and Urban, R. Catalogue of afrotropical bees (hymenoptera: Apoidea: Apiformes). *Zootaxa*, (2455):1–548, 2010.

Eardley, C., Kuhlmann, M., and Pauly, A. The bee genera and subgenera of sub-saharan africa. *ABC Taxa.*, (7), 2010.

Eardley, C. Discover Life. [online] Available at: http://www.discoverlife.org/who/CV/Eardley,_Connal.html [Accessed 15 April. 2015], 2007.

EOL. Encyclopedia of life. [online] Available at: http://www.eol.org.

Ereshefsky, M. *The poverty of the Linnaean hierarchy: A philosophical study of biological taxonomy*. Cambridge University Press, 2000.

Existential Query plugin. [online] Available at: https://github.com/protegeproject/existentialquery.

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

Franz, N. and Thau, D. Biological taxonomy and ontology development : scope and limitations. *Biodivers. Informatics.*, 7:45–66, 2010.

Gardiner, T., Tsarkov, D., and Horrocks, I. Framework for an automated comparison of description logic reasoners. In *The Semantic Web-ISWC 2006*, pages 654–667. Springer, 2006.

Gaston, K. and O'Neill, M. Automated species identification: why not? *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 359(1444): 655–67, 2004. ISSN 0962-8436.

Gennari, J., Musen, M., Fergerson, R., Grosso, W., Crubézy, M., Eriksson, H., Noy, N., and Tu, S. The evolution of protégé: an environment for knowledge-based systems development. *International Journal of Human-computer studies*, 58(1):89–123, 2003.

GO: The Gene Ontology. [online] Available at: http://www.geneontology.org/.

Godfray, H. Challenges for taxonomy. *Nature*, 417(6884):17–19, 2002.

Godfray, H., B.R.Clark, Kitching, I., Mayo, S., and Scoble, M. The web and the structure of taxonomy. *Systematic biology*, 56(6):943–55, 2007. ISSN 1063-5157.

Gomez-Perez, A., Fernández-López, M., and Corcho, O. *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Science & Business Media, 2006.

Grau, B. C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., and Sattler, U. Owl 2: The next step for owl. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):309–322, 2008.

Griffiths, G. The future of linnaean nomenclature. *Systematic Biology*, 25(2):168–173, 1976.

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

Guarino, N. *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*, volume 46. IOS press, 1998.

Guerra-García, J., Espinosa, F., and García-Gómez, J. Trends in taxonomy today: an overview about the main topics in taxonomy. *Zoologica baetica*, 19:15–49, 2008.

HAO. Hymenoptera Anatomy Ontology Portal. Available at: http://portal.hymao.org/projects/32/public/ontology/.

Hoagland, K. The taxonomic impediment and the convention on biodiversity. *Association of Systematics Collections Newsletter*, 24(5):61–62, 1996.

Horridge, M. and Bechhofer, S. The owl api: A java api for owl ontologies. *Semantic Web*, 2 (1):11–21, 2011.

Horridge, M., Jupp, S., Moulton, G., Rector, A., Stevens, R., and Wroe, C. A practical guide to building owl ontologies using protégé 4 and co-ode tools edition 1. 2. *The University of Manchester*, 2009.

Horrocks, I. and Sattler, U. A description logic with transitive and inverse roles and role hierarchies. *Journal of logic and computation*, 9(3):385–410, 1999.

Horrocks, I., Patel-Schneider, P., and Harmelen, F. V. From shiq and rdf to owl: The making of a web ontology language. *Web semantics: science, services and agents on the World Wide Web*, 1(1):7–26, 2003.

Horrocks, I., Kutz, O., and Sattler, U. The even more irresistible sroiq. *KR*, 6:57–67, 2006.

International Code of Zoological Nomenclature. International Commission on Zoological Nomenclature (Inst). [online] Available at: http://www.nhm.ac.uk/hosted-sites/iczn/code/ [Accessed 10 March 2015], 1999.

Iqbal, R., Murad, M., Mustapha, A., and Sharef, N. An analysis of ontology engineering methodologies: A literature review. *Research journal of applied sciences, engineering and technology*, 6(16):2993–3000, 2013.

ITIS. Integrated taxonomic information system (itis). [online] Available at: http://www.itis.gov.

Kalyanpur, A., Parsia, B., Sirin, E., Grau, B., and Hendler, J. Swoop: A web ontology editing browser. *Web Semantics: Science, Services and Agents on the World Wide Web*, 4 (2):144–153, 2006.

Kapoor, V. *Principles and practices of animal taxonomy*. Science Publishers, 1998.

Kleene, S. *Mathematical logic*. Courier Corporation, 2002.

Krötzsch, M. *OWL 2 Profiles: An introduction to lightweight ontology languages*. Springer, 2012.

Krötzsch, M., Simancik, F., and Horrocks, I. A Description Logic primer. *arXiv preprint arXiv:1201.4089*, 2012.

Lassila, O., Swick, R., Ralph, R., et al. Resource description framework (rdf) model and syntax specification. 1998.

Levesque, H. and Brachman, R. *A fundamental tradeoff in knowledge representation and reasoning*. Laboratory for Artificial Intelligence Research, Fairchild, Schlumberger, 1984.

Linnaeus, C. *Systema naturae. 10th ed.*, volume v.1. Laurentii Salvii, Stockholm, 1758.

Lucid. Lucid central: Lucid. [online] Available at: http://www.lucidcentral.com [Accessed 20 Jan. 2014].

Matthew Horridge. Stanford university profile. [online] Available at: http://web.stanford.edu/~horridge/.

Maxted, N. Towards defining a taxonomic revision methodology. *Taxon*, pages 653–660, 1992.

May, R. How many species inhabit the earth. *Scientific American*, 267(4):42–48, 1992.

McGuinness, D. Ontologies Come of Age. *Spinning the semantic web: bringing the World Wide Web to its full potential. MIT Press, Cambridge, MA*, 2005.

Merriam-Webster Dictionary. [online] Available at: http://www.merriam-webster.com/dictionary/classification [Accessed 20 May. 2015].

Midford, P., Dececchi, T., Balhoff, J., Dahdul, W., Ibrahim, N., Lapp, H., Lundberg, J., Mabee, P., Sereno, P., Westerfield, M., et al. The vertebrate taxonomy ontology: a framework for reasoning across model organism and species phenotypes. *J. Biomedical Semantics*, 4:34, 2013.

Mora, C., Tittensor, D., Adl, S., Simpson, A., and Worm, B. How Many Species Are There on Earth and in the Ocean? *PLoS Biol*, 9(8), 2011.

NCBI. National Center for Biotechnology Information: Organismal Classification. [online] Available at: https://bioportal.bioontology.org/ontologies/NCBITAXON.

Neon Project. The NeOn Toolkit. [online] Available at: http://neon-toolkit.org/wiki/Main_Page.html.

Norton, G. Ii. invasive species: the role of lucid identification keys. *Extension Bulletin-Food & Fertilizer Technology Center*, 561:7–10, 2005.

Norton, G., Patterson, D., and Schneider, M. Lucid: A multimedia educational tool for identification and diagnostics. *International Journal of Innovation in Science and Mathematics Education (formerly CAL-laborate International)*, 4(1), 2012.

Noy, N., McGuinness, D., et al. Ontology development 101: A guide to creating your first ontology, 2001.

OWL API. [online] Available at: http://owlapi.sourceforge.net/.

Page, R. A taxonomic search engine: Federating taxonomic databases using web services. *BMC bioinformatics*, 6(1):48, 2005.

Patterson, D., Cooper, J., Kirk, P., Pyle, R., and Remsen, D. Names are key to the big new biology. *Trends in ecology & evolution*, 25(12):686–91, 2010. ISSN 0169-5347.

Pennisi, E. Taxonomic Revival. *Science*, 289(5488):2306–2308, 2012. ISSN 00368075.

Protégé. The Protégé Ontology Editor. [online] Available at: http://protege.stanford.edu, 2012.

Roskov, Y., Abucay, L., Orrell, T., Nicolson, D., Flann, C., Bailly, N., Kirk, P., Bourgoin, T., DeWalt, R., Decock, W., Wever, A. D., and eds. Species 2000 & itis catalogue of life, 2016 annual checklist. *Species 2000: Naturalis, Leiden, the Netherlands, ISSN 2405-884X*, 2016.

Rumbaugh, J., Jacobson, I., and Booch, G. *The Unified Modeling Language Reference Manual*. Pearson Higher Education, 2004.

Schaerf, A. Reasoning with individuals in concept languages. In *Advances in Artificial Intelligence*, volume 728 of *Lecture Notes in Computer Science*, pages 108–119. Springer Berlin Heidelberg, 1993.

Schmidt-Schauß, M. and Smolka, G. Attributive concept descriptions with complements. *Artificial intelligence*, 48(1):1–26, 1991.

Schulz, S., Stenzhorn, H., and Boeker, M. The ontology of biological taxa. *Bioinformatics*, 24(13):i313–i321, 2008.

Shadbolt, N., Hall, W., and Berners-Lee, T. The semantic web revisited. *Intelligent Systems, IEEE*, 21(3):96–101, 2006.

Shearer, R., Motik, B., and Horrocks, I. Hermit: A highly-efficient owl reasoner. In *OWLED*, volume 432, page 91, 2008.

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

SNOMED CT. [online] Available at: http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html.

Sodhi, N., Brook, B., and Bradshaw, C. Causes and consequences of species extinctions. *The Princeton Guide to Ecology*, pages 514–520, 2009.

Sowa, J. *Principles of Semantic Networks: Explorations in the representation of knowledge*. Morgan Kaufmann, 2014.

Spackman, K., Campbell, K., and Côté, R. Snomed rt: a reference terminology for health care. In *Proceedings of the AMIA annual fall symposium*, page 640. American Medical Informatics Association, 1997.

Species 2000. [online] Available at: https:www.sp2000.org.

Spyns, P., Tang, Y., and Meersman, R. An ontology engineering methodology for dogma. *Applied Ontology*, 3(1-2):13–39, 2008.

Stork, N. How many species are there? *Biodiversity & Conservation*, 2(3):215–232, 1993.

Suárez-Figueroa, M., Gomez-Perez, A., and Fernandez-Lopez, M. The neon methodology for ontology engineering. In *Ontology engineering in a networked world*, pages 9–34. Springer, 2012.

The OBO Foundry. [online] Available at: http://www.obofoundry.org.

Timmermann, K. and Kuhlmann, M. Taxonomic revision of the african bee subgenera patellapis, chaetalictus and lomatalictus (hymenoptera: Halictidae, genus patellapis friese 1909). *Zootaxa*, 2099:1–188, 2009.

Tobies, S. The complexity of reasoning with cardinality restrictions and nominals in expressive description logics. *Journal of Artificial Intelligence Research*, pages 199–217, 2000.

Tsarkov, D. and Horrocks, I. Fact++ description logic reasoner: System description. In *Automated reasoning*, volume 4130 of *Lecture Notes in Computer Science*, pages 292–297. Springer Berlin Heidelberg, 2006.

Tsarkov, D. and Horrocks, I. Fact++ description logic reasoner. [online] Available at: http://owl.man.ac.uk/factplusplus/, 2007.

Tudorache, T., Vendetti, J., and Noy, N. Web-protege: A lightweight owl ontology editor for the web. In *OWLED*, volume 432, 2008.

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

VTO. Vertebrate taxonomy ontology. [online] Available at: https://bioportal.bioontology.org/ontologies/VTO.

W3C. The world wide web consortium. [online] Available at: http://www.w3.org/.

Walter, D. and Winterton, S. Keys and the crisis in taxonomy: extinction or reinvention? *Annual review of entomology*, 52:193–208, 2007. ISSN 0066-4170.

WebProtégé. [online] Available at: http://webprotege.stanford.edu/.

Wheeler, Q., Raven, P., and Wilson, E. Taxonomy: impediment or expedient? *Science (New York, NY)*, 303(5656):285, 2004.

Yoder, M., Miko, I., Seltmann, K., Bertone, M., and Deans, A. A gross anatomy ontology for hymenoptera. *PloS one*, 5(12):e15991, 2010a.

Yoder, M., Miko, I., Seltmann, K., Bertone, M., and Deans, A. A gross anatomy ontology for hymenoptera. *PloS one*, 5(12), 2010b.