Metadata-guided Species Distribution Mapping

Blanca Pérez Lapeña

April, 2004

Metadata-guided Species Distribution Mapping

by

Blanca Pérez Lapeña

Thesis submitted to the International Institute for Geo-information Science and Earth Observation in partial fulfilment of the requirements for the degree in Master of Science in *Geoinformatics*.

Degree Assessment Board

Thesis advisor	dr. ir. Rolf A. de By
Thesis examiners	Fabio Corsi, M.Sc. dr. ir. Maurice van Keulen

INTERNATIONAL INSTITUTE FOR GEO-INFORMATION SCIENCE AND EARTH OBSERVATION

ENSCHEDE, THE NETHERLANDS

Disclaimer

This document describes work undertaken as part of a programme of study at the International Institute for Geo-information Science and Earth Observation (ITC). All views and opinions expressed therein remain the sole responsibility of the author, and do not necessarily represent those of the institute.

Acknowledgements

This work would not have been possible without the constant support, encouragement and guidance of my first supervisor dr. ir. Rolf de By.

I want to thank Fabio Corsi, my second supervisor, for willing to share all his knowledge in all sorts of biodiversity issues. He has helped me to understand a domain that was new for me at the beginning of this thesis.

I am grateful to prof. dr. Menno-Jan Kraak, head of the GIP department, Gerrit Hurneman, M.Sc. and dr. Theo Bouloucos, Student Advisor, for giving me the possibility to pursue this thesis work.

I would also want to thank my colleagues in GIP, for being so supportive; Ellen-Wien Augustijn, Wim Bakker (miau) and Marijke Smit for their support and concern during all this time. I am also grateful to Arta Dilo, for the time spent and for our discussions regarding this thesis work.

I am very thankful to all my friends, for being close to me in all the good and the bad moments. There have been very special conversations that I will never forget... Finally, thanks to my parents, my brother and Carolina for being always 'there', any time.

Abstract

As has become apparent to both the popular and scientific press, there exists grave concern for our planet's environmental well-being. At alarming speeds the Earth is being depleted from many of its non-renewable resources, quenching the thirst of growth-based economies and human populations on the increase.

It is in this context that we have to understand the importance of man's study of ecosystems. The understanding of the occurrence and distribution of plant and animal species plays, obviously, an important role in the study of ecosystems. To understand why a species occurs in some ecosystem means to better understand its ecological requirements and dependencies. This brings us, at least here, to species distribution mapping ...

In this thesis work, we report on our attempts to contribute to methodical consistency, specifically in that of repeatable, instantaneous computer-aided species distribution mapping, in scenarios where new data sets become available regularly. We do not attempt to answer ecological problems here, but rather want to provide flexible methods supporting ecologists in their mapping procedures, in the hope of deriving a procedural understanding that could eventually be (better) automated.

Specifically, we address the issue of automatically constructing a reliable method for determining (anew) a species distribution map, using a GIS, from spatial foundation data, species knowledge, mapping method knowledge and map purpose. We work under the assumption that any of the latter four inputs may change overnight, possibly resulting in redetermination of the output, the species map.

We investigate formalisms that would allow us to describe and manipulate data and their metadata together; that would accommodate the description of taxonomic data (as in data taxonomies, or ontologies); that would be related to formats already in use for (geo)data exchange on the internet; and that would allow to reason over such descriptions. In short, the formalism that we were looking for had to be declarative, and preferably logic-based, and fit for data exchange. We apply one of the formalisms of the family of description logics, namely $\mathcal{SHIQ}$.

We focus on a *deductive* approach for species mapping procedures, which main characteristic is the use knowledge on species ecological preferences. We represent this knowledge in an ontology (the species ontology) and we use RACER to run various queries against the ontology.

We conclude the work by viewing the problem of mapping a species distribution as a *configuration problem*, and apply description logics to this domain.

Keywords

species model, GIS, description logics, ontology, knowledge base, reasoning, configuration

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and motivation

1.1.1 Planet Earth in peril

As has become apparent to anyone with even the slightest interest in world matters, from both the popular and scientific press, there exists grave concern for our planet's environmental well-being. At alarming speeds the Earth is being depleted from many of its non-renewable resources, quenching the thirst of growth-based economies and human populations on the increase.

Indeed, essentially already for some decades or even longer, man's use (and abuse) of natural resources such as cultivatable land, freshwater, natural forests, fishing grounds, numerous mineral resources and especially fossil fuels has been labelled irresponsible, unsustainable and irreversible [52]. While for some of these phenomena — for instance, deforestation [13] — the figures are overwhelming and irrefutable (see Table 1.1 and Figure 1.1), sceptics with undisclosed agendas will not quickly sign off on more difficult to quantify but equally alarming news on related phenomena, such as global warming [2, and many more].

Fact of the matter is that many of the world's ecosystems are undergoing dramatic changes at unprecedented pace, commonly into directions with bleak outlooks. Another fact is that man's understanding of these ecosystems is far from complete, and that concern is mounting about whether we can ever complete this understanding, before it is too late.

It is in this context that we have to understand the importance of man's study of ecosystems. An ecosystem, under one definition, is 'a dynamic and interrelating complex of plant and animal communities and their associated non-living environment' [`www.hyperdictionary.com`]. The understanding of the occurrence and distribution of plant and animal species plays, obviously, an important role in the study of ecosystems. To understand why a species occurs in some ecosystem means to better understand its ecological requirements and dependencies. This brings us, at least here, to species mapping ...

But this thesis is not aiming to address any of the truly phenomenal issues raised above directly, nor is its intent to provide answers that may eventually become part of the big puzzle's solution. Rather, it is an attempt to contribute

Table 1.1: Deforestation per region in figures. 'Frontier forest' is defined as 'ecologically intact, natural forest, relatively undisturbed and large enough to maintain all of its biodiversity'. 'Original forest cover' is defined as forests in existence before human impact on them started to take place. Source: Global Forest Watch, `www.globalforestwatch.org`.

	Forest cover $[\mathrm{km}^2]$		
Region	*Original*	*Current total*	*Current frontier*
Africa	6,799	2,302	527
Asia	15,132	4,275	844
North America	10,877	8,483	3,737
Central America	1,779	970	172
South America	9,736	6,800	4,439
Europe	4,690	1,521	14
Russia	11,759	8,083	3,448
Oceania	1,431	929	319

to methodical consistency, specifically in that of computer-aided, species distribution mapping. We do not attempt to answer ecological problems, but rather study how ecologists have gone about their mapping procedures, in the hope of deriving a procedural understanding that could eventually be (better) automated.

1.1.2 Fields of application

Biodiversity research

Guisan and Zimmermann described in [34], including references to other authors, the importance of predicted geographical modelling as a tool to:

- assess the impact of accelerated land use change and other environmental changes (e.g., climate) on the distribution of organisms,
- to test biogeographic hypotheses,
- to improve floristic and faunistic atlases, and
- to set up conservation priorities.

In the field of ornithology, Isler [43] stated that distributional knowledge and an ability to interrelate spatial data are vital to a wide range of studies including:

- reviews of systematics and phylogeny requiring detailed geographic knowledge of (historic) opportunities for gene flow,
- definitions of endemism, central to analysis of historical biogeography,
- examinations of geographic variation in a species' morphology as it relates to, for example, habitat and climate,

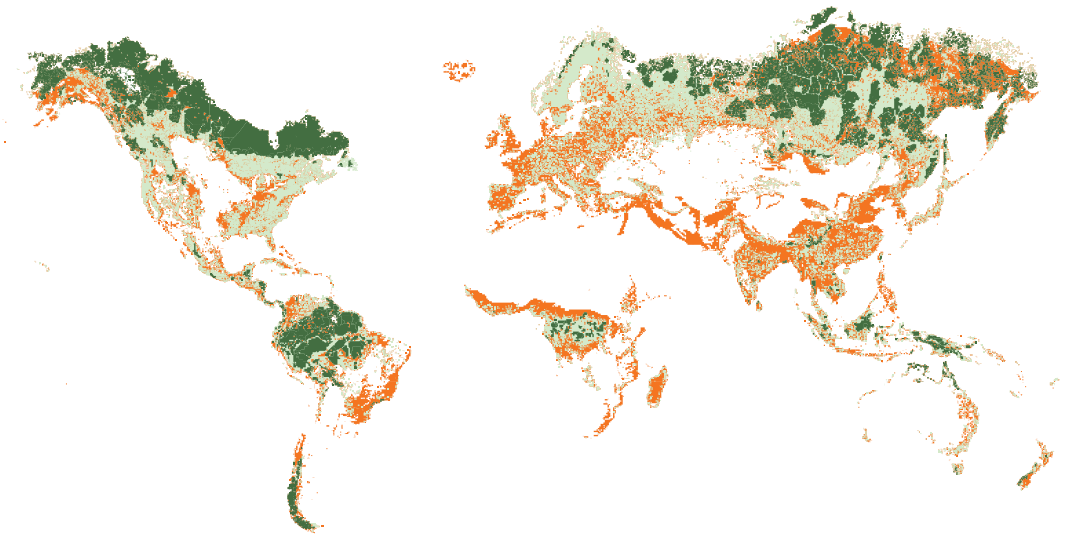

Figure 1.1: Global deforestation mapped. In orange, original forest cover lost; in light green, current non-frontier forests; in dark green, current frontier forests. Observe that map projection used (Mercator) overly emphasizes areas at higher latitudes. Map data © by World Resource Institute on behalf of Global Forest Watch.

- regional identifications of concentrations of endangered species used to establish priorities for acquisition of conservation areas, and
- broad-scale analysis that interrelate species distributions to, for example, climate and vegetation.

Global health matters

As an aside, we want to mention that the concern for biodiversity and studies of ecological nature are an, albeit, important reason for attempts to automate species mapping procedures, they are not the only reason. Another important reason can be found in human epidemic health risks, especially in the fight against vector-borne viral infections that are (in part) carried over to humans by animals. Malaria (through 60 species of mosquitoes *Anopheles*), dengue fever (various mosquitoes of the subfamily *Stegomyia*, but especially *Aedes aegypti*), Chagas' disease (various 'kissing bugs' from the subfamily *Triatominae*), African tripanosomyiasis (through tsetse flies *Glossina* spp.), SARS (severe acute respiratory syndrome; believed to involve a Chinese species of civet — either *Paradoxurus hermaphroditus* or *Viverra* sp.), West Nile fever, West Nile encephalitis and West Nile meningitis (all by mosquitoes *Culex spec.*), leprosy (the Nine-banded Armadillo *Dasypus novemcinctus* being a suspect in some areas), bilharzia (flatworms of the genus *Schistosoma*, in collaboration with snails of the genera *Bulinus*, *Biomphalaria* and *Oncomelania*), Avian influenza A (especially by domesticated chicken *Gallus* and duck *Anas*) are just a number of high-profile human diseases in which improved understanding of the distribution of related animal species is, or may become, important.

1.1.3 The need for methods

In all of the cases mentioned above, if a species mapping procedure (SMP) was involved, it was in all likelihood carried out as a one-off procedure, paying attention to all the case-specific details, not to be immediately or quickly repeated, and certainly not to be applied without change to other SMP cases. No attention was probably paid to repeatability, or development of 'SMP methodology'. In recent years, quite some research on SMP methodology has been published [60, 12, 33, 19, 17, 47, 34, 59, 65, 18, 9, 50, 54], however, not always with automation as the primary target. We believe that automation of SMPs is wanting for various reasons. Here are some:

Formalization and consistency SMP is a demanding, time-consuming and error-prone process that can be helped with automated support, so as to formalize it.

Lack of expert capacity There is not enough expert capacity to manually execute SMP for all organisms that we are interested in; various organisms require regular updates of their spatial distribution.

Growth of data availability We can expect a definitive growth in available (geo)data sources that capture more ecological parameters, or that capture them better. Similarly, updates of observation sets may lead to renewed executions of SMP. Whenever such new data sources become available, we would like to test whether they can improve our SMP results.

Conservational decision-making There is a conservation need for executing what-if SMPs under a multitude of parameters, allowing better forecasting, and thus decision-making.

Responding to ecosystem changes Human activities are causing high impact changes on ecosystems, which is reason for continuous monitoring, in which SMP automation will be useful.

Over the last centuries, many resources have been invested in building biological collections. Nowadays, much data is available in paper libraries, databases and natural history museums. The total number of biological data sets is very high, but not all this data is available and some of it exists only in less suitable formats. Biologists, conservationists and environmental decision-makers know that the study of biodiversity requires analysis of trends in time/space and of relationships between species. Therefore, much effort is being made into digitizing biological data from existing collections.

1.1.4 The role of technology

New technology is now also available to obtain biological foundation data. For example, remote sensing methods provide geospatial data in the form of raster images that can be used to obtain information related to vegetation, climate and elevation. These types of data are important as in combination they can

provide us with an insight of relationships between species and with various environmental variables.

One of the main issues of importance in documenting species is to locate their occurrence geographically. If we look at the old descriptions of species' observations (e.g., two hours in canoe up along the Río Napo), we can see that positioning technology has started to play an important role in improving such descriptions. GPS technology is, obviously, one of the main techniques. Thus, more and more species observation data is being georeferenced.

With the costs involved in generation of new data, time constraints and the increasing need for collaborative research, much effort is being put into techniques of data sharing. Standards have been developed and are being maintained. For example, the Federal Geographic Data Committee (FGDC) has taken action in defining terminology and in providing definitions for the documentation of metadata content related to geospatial data. For biological purposes, standardization work is being carried out and is providing initial results such as extended elements and a biological profile of the FDGC. Data exchange languages were also previously developed, such as the eXtensible Markup Language (XML). This metalanguage enables data exchange of all sorts, and provides good solutions for defining and structuring data as described in the FGDC metadata standard, allowing the exchange of data between different systems and across the Internet.

1.2 Problem formulation

There exists a vast amount of distributional data, but the biological data sets are most of the time patchy and incomplete [55]. One approach of maximizing the use of the available data on species in predicting their distribution is to use models that are based on their ecological preferences. These models allow to map the potential distribution of species in areas where no observations have been made, or are available.

Species distribution mapping is and has been an expertise area, characterized by elaborate, manual work, with input of expert knowledge that is difficult to formalize, with ad hoc decision-making every time a new map is generated, possibly not always with the best practice of handling the whole process methodically.

Although species distribution mapping has been improved by the use of Geographic Information Systems (GIS), the development of environmental models and the availability of geospatial data, there seems to be an emerging need for (further) standardizing and (semi)automating the procedures. Moreover, metadata descriptions of the data, of the model functionality, the procedures applied, and the mapping results are most of the time not available nor provided. To make the mapping process reusable at any moment, we would like to keep track of the complete procedure and of its constituents.

The problem that we address in this thesis, therefore, is the design of *a generic system for species distribution mapping*. Can such a system be constructed, and become operational in a multi-purposes, useful way?

The keyword here is *generic*. The ultimate purpose is to create a (semi)automatic system that can be used for SMP for any life form, being sensitive to both the purpose of the exercise (what type of map is required?), and the input data sources available.

We will later, in Chapter 4, make the observation that any species distribution mapping procedure is, in the most general and simplistic way, determined by a combination of

Data (D)–Method (M)–Output (O).

The D component comprises the available or required data sources for obtaining a specified output O (e.g., the mapped probability of occurrence of a species), while M represents the method used to generate output O with data D. The method M could be, for instance, a precisely described statistical method.

The task of generating a distribution map is not straightforward. Typically, a large number of combinations of the (D, M, O) components could make sense. For instance, concerning the data sets, we will have to make choices amongst them, basing our decision on which ones to use on characteristics such as the phenomena they represent, their spatial extent and/or their quality. The data format together with the phenomena represented may also dictate the different processing steps that are needed to generate the eventual map.

All three components will have to be properly described; in fact, since we will be looking at alternatives, for each component — D, M and O — we will have a number of candidates to fill the respective slot. We are looking therefore for appropriate characterizations of every candidate. And this brings us to metadata.

1.2.1 Hypothesis

The working hypothesis of this thesis work can be formulated as follows.

> *Appropriate metadata on potential D, M and O components can be used as high-level signature information to allow the adequate (automated) composition of actual species mapping procedures.*

1.3 Research objectives

In the project, we have aimed at defining and developing automated support for species distribution mapping by:

1. defining and developing metadata characteristics of species mapping procedures; applying standardized procedures that are parameterized by data characteristics (metadata values),

2. administering expert knowledge, obtained from previous mapping exercises (if the lineage has been carefully administered),

3. using metadata structures/standards to associate with the various types of data (e.g., species, geospatial data and map type) other characteristics that are important to the mapping process, and

4. developing an inference process (metadata mediator) that attempts to try find a sensible combination of D, M and O.

1.4 Research questions

In this research work, we attempted to give answers to several questions relating to the aforementioned objectives. We can assign the questions to three groups, namely issues on *mapping procedures*, issues on *expert knowledge* and issues on *metadata structures* for D-M-O.

1.4.1 Mapping procedures

We will look at existing mapping procedures described in the literature. This allows us to analyze their main characteristics and study the differences amongst them, to understand the level of genericity needed. We were expecting to find such differences in the models applied and in their data requirements. We wanted to identify and formalize the rules for making proper combinations between D, M and O and for determining the required processing steps.

More precisely, we looked at:

- Which are (the most important) existing models for species distribution mapping?
- Which types of species distribution map can be generated by using these models?
- Are certain models better suited than others for certain types of distribution map?
- Which parameters influence or even determine the species mapping procedure? (observation types, species types, environmental data types, distribution map types, model types)
- Does a taxonomy of mapping procedures exist? If so, can we express it?
- Can we define a (hierarchical) type system for/in the metadata that would help the metadata mediator? Can rules be formalized that help the metadata mediator?
- What is a suitable system architecture for a (mediating) environment for species mapping procedures?

1.4.2 Expert knowledge

The aim of species distribution mapping is to find the areas where a species is most likely to occur. In the past, researchers were studying and describing the possible relationships between the species and the environment to carry out different types of ecological studies. For our system, we wanted to find a suitable structure to store species-environment relationships.

Other types of expert knowledge concern the identification of which data sets are most appropriate for a given mapping request, and which processing steps are best applied to that choice.

Some of the questions we wanted to address were:

- Can expert knowledge be described in the proposed system that supports the species mapping procedure?
- In which stages can the species mapping procedure be automated? How can we make use of metadata to guide us through the process?
- Where do we take care of this expert knowledge, in the data or in the metadata?
- In which stages can expert knowledge be considered in the mapping procedure? For example, should it be considered within application of a model or even in the choice of an appropriate model?

1.4.3 Metadata structures for D-M-O

We believe that the description of the main actors in the species mapping procedure and their relationships (type of output, model description and functionality and data requirements) can be captured within proper metadata structures. Such structures should consider more high-level properties of these three components, capturing the understanding and the semantics of the domain knowledge. Therefore, we wanted to obtain an answer to the following questions:

- Can the parameters relevant for species mapping procedures be embedded in metadata structures?
- What is a suitable knowledge representation model for these three groups? More specifically, are ontologies suitable for this purpose?

1.5 Outline of the thesis

The thesis is structured as follows:

Chapter 2 describes the factors causing complexity in modelling animal and plant distributions. We raise the importance of the spatial and temporal scale of analysis, as elements to consider when studying such distributions. The second part of the chapter describes the main phases of SMPs, identifying the technical challenges that must be accounted for and addressed when building a (semi)automatic system for species mapping.

Chapter 3 reviews the group of formalisms known as Description Logics. We focus on the logic $\mathcal{SHIQ}$ to model UoDs. We discuss the syntax, semantics and pragmatics of the language in fair detail. The second part of the chapter is devoted to RACER, a reasoning system that allows to reason over descriptions in $\mathcal{SHIQ}$.

Chapter 4 looks at the application of Description Logics in our UoD: the automation or semi-automation of species mapping procedures. We start by describing the base data we use in the theses work (from the African Mammals Databank [9]). Then, using OWL, and its underlying formalism $\mathcal{SHIQ}$ we build ontologies for the species and the spatial data set that we worked on.

Chapter 5 describes the problem of mapping a species distribution as a *configuration problem*. We follow a technique proposed in the literature, and apply Description Logics to this domain.

Chapter 6 provides a discussion and concluding remarks on the research work, raising as well some remaining questions that were left for further research.

Chapter 2

Complexity in Species Mapping

The ability to model animal and plant distributions plays an important role in understanding ecosystems. Modelling animal and plant distributions, however, is far from being an easy task, due to the complexity of relations between factors inherent to the ecological systems under study.

In the first part of this chapter, we review some of the factors causing complexity and affecting the distribution of organisms. In the second part of the chapter, we focus more on the technical aspects of the species distribution mapping domain. We start by discussing the role of technology such as Geographic Information Systems (GIS) and Remote Sensing (RS) within this context. Then, we analyse the main phases in the species mapping procedure (SMP). This helps us in identifying the viability of a (semi-)automatic system for SMP and in defining the specific challenges that must be accounted for and addressed when building such as system. An example of such a challenge is the issue of how to deal with spatial data characteristics.

2.1 Species distribution mapping

Studying the distribution of species (animal, plants and micro-organisms) is a long standing objective for wildlife ecologists. It seems that answering the question "where does species X occur" should not be too complicated. In fact, the understanding of where a species occurs and why this is the case is fraught with many difficulties [61]. This section aims at discussing a few of these difficulties, describing the inherent complexity of ecological systems.

The total number of *known* species worldwide is estimated in 1,770,000 [14]. Information about their distribution is in itself incomplete, as wild populations of plants, animals and micro-organisms depend on environmental conditions for their existence and evolution [49].

2.1.1 Factors involved

We can identify two important factor groups that limit the distribution of species: *abiotic* and *biotic* factors. Abiotic factors include non-living chemical and physical factors such as temperature, water, light and availability of nutrients. Biotic factors, on the other hand, are formed by living organisms that play a role in the occurrence of a species.

Many species have temperature tolerance limits; aquatic species typically also show sensitivity to water salinity and acidity levels. In plants, sunlight plays an important role as it affects their development and behaviour (e.g., in photosynthesis). The physical structure, texture and mineral composition of soils affects the distribution of plants and the animals that depend upon them. Predators, as a biotic factor, can also limit the distribution of pray species [44].

One important issue is the *scale* at which a species interacts with the environment. An elephant, obviously, interacts with the environment at another scale than does a butterfly. The environmental variables affecting the distribution of elephants may include certain vegetation types and the existence of water areas. For butterflies, we may need more detailed information such as the existence of certain types of flowers. This has an effect on the data required for mapping their distribution and on the method used to collect such data. For elephants, remote sensing techniques may be used to obtain information related to vegetation whereas for butterflies, this technique may not be sufficient, as it is not able to capture the spatial detail reflecting the species requirements. Therefore, the ecological variables that affect a species' distribution should be studied at an appropriate scale. For instance, wood mice seem not to have a preference amongst several types of croplands, but within each of these types they choose areas with a high abundance of certain plants. It would then be folly to relate this species to a certain type of cropland as the specific sites where the species is observed may just happen to be more common in that cover type than in others [30]. Instead, the relationship should be defined at the scale of occurrence of the food plants.

Similar examples can be found throughout the field of ecology, for instance, in the study of fungi distributions. The ecological variables important to *chanterelle* species also exist at different geographical scales [27]. Approximate predictions of chanterelle distribution at small scales can be obtained by using vegetation composition and condition information. But the patchy distribution of fungal individuals, however, indicates that other factors at the microclimate scale (such as relation to coarse woody debris and micro-topography) may be equally or even more important and, therefore, should be taken into account when mapping their distribution at more detailed scales [27].

Fungi are *static*, mostly immobile organisms. Then, what about animal populations? The vast majority of them are *mobile* in space, which complicates the study of their distribution. Many bird species, for instance, have different seasonal patterns because they are migratory, either geographically or elevationally, and their ecological requirements may well be different depending on season.

Looking at the life requisites for animal species we see that food, resting

sites, cover, and reproduction sites are amongst their main requirements [46]. We would expect their spatial distribution to reflect these requirements, depending on the species behaviour we want to study. For instance, the Red-tailed Hawk *Buteo jamaicensis* displays a shift in cover type usage in the spring-summer and autumn-winter seasons; their food mainly consists of small mammals; it nests in forests or forest patches with trees having more than 50 cm dbh[1] [26].We may have different purposes for mapping the distribution pattern we want to study, e.g., whether it is mapping nesting habitat or mapping migration routes. Each of these purposes requires certain ecological characteristics to be analysed, and others to be discarded because they are irrelevant for the purpose (e.g., nesting sites are important only in the breeding season). In spite of this distinction of mapping purposes, we must realize also that one cannot completely separate between types of behaviour: a nesting bird needs to feed occasionally, so food needs to be available within a certain distance. On the Red-tailed Hawk, it has been found that food-producing areas and reproductive areas must be located within an average of 1.2 km of each other [26].

Other important factors that may affect a species' occurrence are topographic barriers. We may find bird species that do not cross deep mountain valleys, small mammals like rodents that cannot cross too wide rivers or seas, and numerous animal and plant species being restricted to single islands or island groups. Although the existence of these barriers should be first carefully analysed (as it is very difficult to assess when a topographic barrier has an effect on the species distribution) we may also want to include them, together with their ecological requirements, in the mapping procedure.

2.1.2 The time dimension

We have seen that ecological requirements for species vary across spatial scales but they also do so across *temporal scales*. For instance, the American Black Bear's *Ursus americanus* ecological requirements can vary annually, and the data on ecological variables accumulated over the years may yield wrong, i.e., too optimistic, results [30]. These and other factors then have to be accounted for when studying species over time. We may be interested in studying a particular year's territories (snapshot distributions), territories over years (accumulative distributions) or the changes that a species population has suffered over time (historic distributions). Yet quite a few other species, like moths (e.g., Hummingbird Hawk-moth *Macroglossum stellaratum*), butterflies (e.g., Painted Lady *Cynthia cardui*) and birds (e.g., crossbills *Loxia* and waxwings *Bombycilla*) particularly, and of course the classic example of the Norway Lemming *Lemmus lemmus*, display patterns of irregular eruptions, i.e., mass movements far away from the normally occupied areas. Sometimes these are believed to be related to food scarcity, at other times to abnormal weather patterns. But our knowledge of (our data on) these phenomena may be largely non-existent.

Ecological systems themselves are clearly *dynamic*. For example, human decisions to change land uses, the introduction of conservation practices, and

[1] Dbh: diameter at breast height

changes in global weather patterns can all cause ecological changes that affect the distribution of species.

Other factors that should be considered as well when mapping species distributions include human disturbance, exploitation, predation and competition.

We can conclude from the above discussion that species mapping involves many different parameters, which should all be accounted for. Purpose of the map, landscape scale of the species activity, type of the species behaviour, the temporal factor: each of these needs to be considered. Thus, the type of phenomenon that we want to map is therefore far from simple.

2.2 Enhancement of species mapping procedures with GIS and RS

2.2.1 Distribution map types

According to [62] there are four traditional methods for mapping the distribution of species:

- dot distribution maps,
- grid-based maps,
- hybrid dot distribution maps, and
- range maps.

Dot distribution maps place dots in the map where the species of study has been recorded. In only these points are we certain that the species has been observed, but obviously, nothing can be said about other locations in the area covered by the map. Dot maps may have accumulative legends, meaning that the size of the dot for a cell is indicative of the number of observations. In *grid-based maps*, the territory is divided into uniform units ('cells') of a certain dimension (e.g., 10×10 km). Grid cells typically are square, but may be rectangular. When a species has been observed in a locality, a dot is placed in the centre of the corresponding cell. This method has also limitations, as it provides less information of where the species was really observed. Sometimes these maps are used to protect sensitive or otherwise threatened species. *Hybrid dot distribution* and *range maps* show locality records of species but enclose them within a boundary. The limit is determined by boundaries of major biomes (e.g., forests and deserts).

2.2.2 Explanatory environmental variables and GIS

As we can see, all these types of map are based on observations of the species. We obviously cannot expect to obtain an accurate, even perfect, distribution result as it is impossible to survey all localities where the species may be present. Moreover, we have already noted that species distributions are dependent upon varying suites of environmental factors that relate to both the physical and

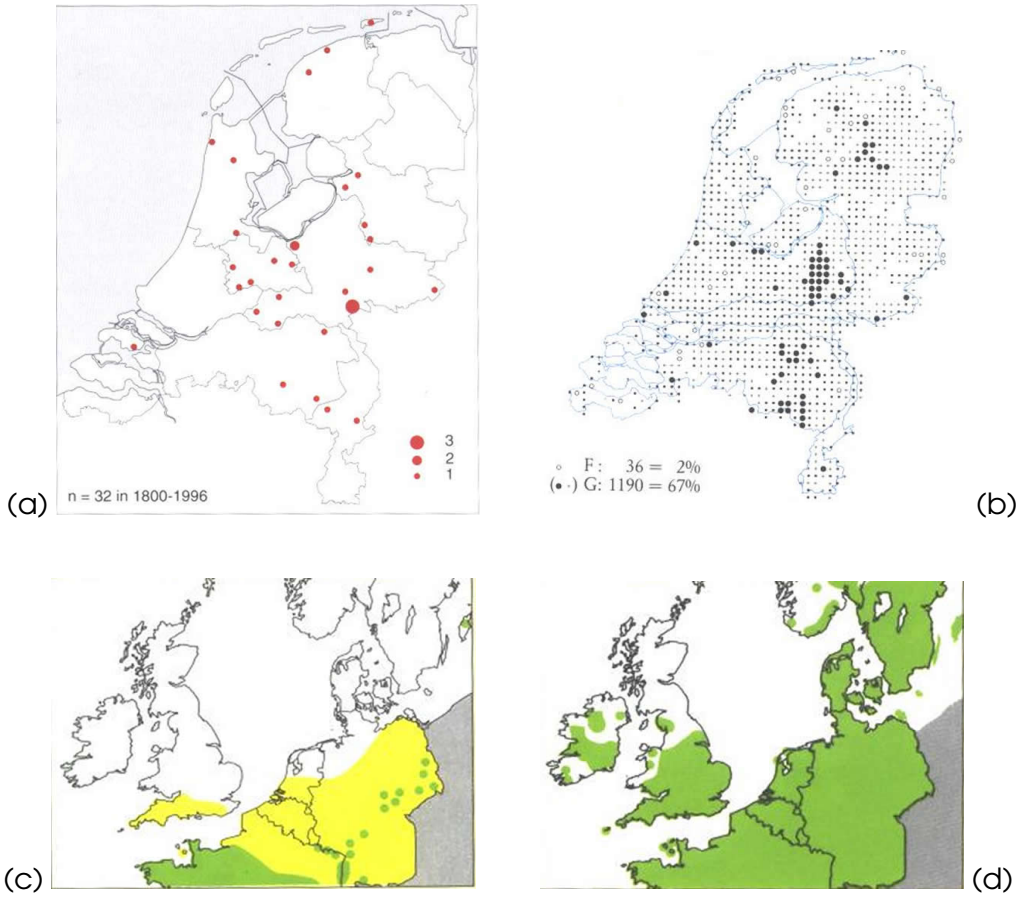

Figure 2.1: Different species distribution maps. (a) Distribution of records of Sociable Lapwing (*Vanellus gregarius*) in the Netherlands over the years 1800–1996 (63) (b) distribution of the Hobby (*Falco subbuteo*) in month of August (Netherlands) over the years 1979–1984 (4) (c) Distribution of the Short-tailed Blue (*Everes argiades*) in NW-Europe (5) (d) Distribution of the Brimstone (*Gronepteryx rhamni*) in NW-Europe (5).

non-physical environment. Therefore, a common approach for mapping species distributions is to relate the taxon under study to a set of (assumed explanatory) environmental variables, and not only to the locations where the species has been recorded. This allows us to extend the mapping exercise to larger geographic areas.

The area of Geographic Information Systems (GIS) has improved species distribution mapping in many aspects. They are systems capable of storing and representing information in digital form, fundamentally different from traditional paper maps. Besides the computer-aided cartographic support they offer, their strength lies also in their analytical capabilities. They allow to combine different spatial data sets, known as 'spatial data layers', and derive new spatial information from them. For our purpose, the layers may represent environmental information, and can thus be used to derive important ecological relationships that may be difficult and time-consuming to identify with traditional methods. Different spatial data sets, from different sources (e.g., topog-

raphy, satellite imagery and aerial photographs) can be handled and analysed at appropriate scales within the same system.

Methods for collecting data associated to species have also been improved with Remote Sensing (RS) techniques. For instance, the spectral reflectance of different vegetation types can be captured in multispectral images, providing the means for vegetation classification and mapping [65]. Images from different (subsequent) years can be (thoughtfully) combined to study, for instance, changes in species distribution patterns.

Advantages of satellite imagery include repeat viewing, digital format, information over large areas and good geometric properties [65]. But we also have to be aware of the limitations of RS data for use in mapping procedures. For instance, imagery may not be available for certain characteristics, such as species microhabitat, simply because current satellite platforms do not provide images at the appropriate resolution.

2.3 Phases in species mapping procedures

In an attempt of building a (semi-)automatic system for species mapping procedures, we should first carefully analyse the phases involved. In this section, we look at SMP within a GIS context. We look at the phases that constitute them and we provide a general overview of the different choices available for the user.

Conceptually, developing a distribution map with the use of GIS can be summarized as follows: the existence of several "layers," each of them describing the distribution of an environmental variable (such as vegetation or elevation), and the species' ecological preference being defined respective to these environmental "layers." The final distribution map is then constructed to show the areas where this preference is met, either actual (when there is evidence of presence) or potential (where the species has not been observed) [18].

GIS models, according to [18], can be classified according to the methodology used to build them. They fit into two main groups: inductive and deductive models. *Inductive models* make use of observations of the species to derive the ecological preference from the environmental characteristics in the particular locations of observation. In the *deductive models*, the ecological preference is considered known a priori, either by extraction from the literature, or as provided by expert opinion. In both models, once the species' ecological preferences have been determined, a next step identifies the areas where the ecological preference characteristics are met.

In the first phase of the model, one has to identify which are the environmental parameters that potentially take part in the species' mapping procedure. Guisan and Zimmermann [34] indicate that the choice of parameters to be included can be based on the scale of the species' distribution under study. For instance, it is shown that the distinction between topographic and bioclimatic variables may affect the distribution map at different scales. They conclude that for modelling (vegetation) distributions at large scales and in complex to-

pography, indirect variables[2] may give better predictions while for small scales, it is more appropriate to use direct and resource variables[3]. Corsi et al. [18] also discuss this matter, stating that factors that are important to consider when mapping the distribution of a species vary according to scale. For instance, they provide an example in which the factors to consider at a continental scale can be related solely to climate. At larger scales, land form and topography play an important role [1], whereas at still larger scales, local features such as a single tree or a channel are considered more significant [38]. These considerations emphasize the selection of an appropriate scale for the analysis and as well as that of the explanatory variables that are important in building the model at the specified scale.

The main difference between the inductive and deductive approaches is in the way that the species' ecological preferences are defined. Therefore, we make a distinction between these two approaches in the following discussion.

2.3.1 Species-environment relationships

In a deductive approach, the environmental variables (and their values) presumed to affect the species' distribution are known a priori. Therefore, the phase of variable identification stops at this point. In an inductive approach, this identification is not so straightforward. There, observations of the species are used to derive such variables, and their corresponding values. One technique for obtaining such an ecological characteristic makes use of available environmental layers, and existent species observations stored in a GIS [18] to calculate the mean of each variable using the points of observation. An ecological preference in such a case could be, for instance, "the species is know to live in montane and intermediate forest up to 2500 meters."

In both approaches (deductive and inductive) the ecological preference is defined as if the variables taken into account are of equal importance. Refined techniques may give more importance to specific variables defining the ecological preference.

In the deductive approach, techniques such as multi-criteria decision-making, the nominal group technique (NGT) and Delphi [18] can be used for this purpose. As an example, the Delphi technique is a procedure that takes into account expert opinion. It asks for inclusion of the appropriate variables in the model, the ranking of the values within each variable, and the weight that each variable has in relation to the other variables. The method calculates the median of these opinions and confronts the experts with the result for another round of estimates. This is done several times, eventually using the median of the final round as the best answer [25]. The ecological preference is therefore defined as a weighted combination of variables, where the weight for each variable determines its rank.

[2] I.e., variables that have no direct physiological relevance for a species performance — e.g., slope, aspect and elevation [34].

[3] Resource variables include matter and energy consumed by plants or animals. Examples are nutrients, water, light for plants, food and water for animals. Direct variables are environmental parameters that are of physiological importance, but are not consumed. For instance, temperature and acidity [34].

In an inductive approach, statistical techniques can be used to both analyse the variables affecting the species distribution, as well as the relative importance of each variable. The data required to perform such analysis consists of species observations and environmental information. These data can be already at hand (digitized topographic maps, remote sensing data), or they can be obtained from field surveys. In the latter case, a sampling strategy for collecting such data is useful [34], and leads to improvements of the resulting distribution map.

There exists a vast range of statistical methods for obtaining the species ecological preference. In some cases, the model predictions can be greatly improved by applying a particular statistical approach [34]. Some statistical methods include: generalized regression, neural networks, ordination and classification methods, Bayesian models, locally weighted approaches (e.g., GAM), environmental models [34] and the Mahalanobis distance [18]. For instance, linear regression is one of the oldest statistical techniques but is limited by three main assumptions: the error in the measurement is assumed to be identically and independently distributed, the response variable (e.g., the species presence) is assumed to be normally (Gaussian) distributed and the regression function is linear in the predictors (e.g., vegetation and elevation) [33]. Generalized linear models (GLM) are considered to be more flexible as they allow other distributions of the response variable (e.g., Poisson) [34]. Therefore, the choice of methods requires certain considerations.

In any statistical approach, the variable selection is of high importance and several techniques are available for identifying which predictors should enter in the final model (e.g., ridge regression applicable to GLM) [33]. Once the variables have been carefully chosen, the coefficients for the selected statistical method can be calculated. After this phase, the species ecological preference can be defined.

2.3.2 Building the distribution map

Once the ecological preferences have been defined, with either approach, the next phase is to predict the species' occurrence at unsampled locations with a GIS, thereby obtaining the species distribution map.

The steps required are dependent on the techniques used as discussed in the previous section. For instance, one may assign the presence/absence to values in each environmental layer under consideration followed by an overlay operation that gives as a result those areas where the environmental characteristics for the species are met. We have also seen that other modelling techniques can assign a rank to the values within each layer, and perform an overlay, attributing different importance (weight) to the layers involved. Ecological preferences obtained from statistical methods can be also mapped with a GIS. For instance, GML models can be easily implemented by multiplying each coefficient with each related predictor variable (although some transformations may be required to obtain probability values (between 0 and 1), or to obtain the same scale of the original response variable) [34]. An overview of GIS implementations, and their limitations, of the several statistical methods

can be found in [34].

2.3.3 Evaluation

An important phase in the species mapping procedure is the evaluation of the final map. There exist two approaches to the evaluation of the prediction performance, when using statistical methods: one of them makes use of two independent data sets (one is used to build the model and the other is used for evaluation), while the second one makes use of a single data set (for both building the model and evaluation). A technique falling within the first group includes the split-sample approach (when both data sets are obtained from splitting the original data set) and an evaluation is made to see the fit of the observed values with the evaluation data set. Techniques falling in the second approach include the Jack-Knife, cross-validation (CV) and bootstrap [34] approaches.

In the evaluation procedure, we also have to consider the errors committed in GIS context, meaning the error resulting from: geometric and radiometric error from remotely sensed data, time lag between environmental data collection and species observation, digitization error of analog data sources, and conversion error between raster and vector data sets. This means that an accuracy assessment of the original data has to be carried out. Moreover, when combining layers in a GIS (e.g., in overlay operations) propagation of error takes place. Error propagation analysis techniques are discussed in [18], and serve to identify the level of accuracy of the final distribution map. Other techniques, such as sensitivity analysis, can be used to define the reliability of the final map by analysing the variability of the predictions when modifying the model's parameters [18].

2.4 Challenges for a (semi-)automatic system for SMP

We have looked at the complexity of ecological systems and the phases required for generating distribution maps, including the choices amongst different approaches. In this section, we analyse some of the more technical challenges that we face when building a (semi-)automatic system for species mapping. We look at them from four different perspectives: species data, distribution models, the spatial data required for building the model and the generation of the final map.

2.4.1 Species data

In this section, we want to raise some of the limitations inherent to species data. Let us start with the scenario in which a user wants to generate a historic distribution map. In this case, we would most probably rely on information about species recorded a long time ago. If we look at such historical records, we must observe that digitizing such biological collections can be a cumbersome task. Old records are held in museums, most of the time in paper libraries and collected within many different time periods. Geographic references may be

lacking, may be incomplete, or may have become untraceable, for instance because topographic names have changed over time. In the following discussion, we leave out the fact that such information may be stored in paper libraries and we assume that the information is already in digital form.

One of the problems regarding the combination of species information assembled through different time periods, from different sources and collected by different communities is the instability of taxonomic nomenclature [23, 24, 53]. The association of a name with a particular taxon, or the decision as to which group of organisms actually belongs a single species, involves an element of subjectivity, and may change over time and between scientists' opinions. This may mean that the species referred to as '*Astragalus aboriginum* Sprengel' in one database is known as '*Astragalus forwoodii* Watson' in another [28]. This semantic ambiguity, therefore, may cause problems when data has to be combined within an SMP.

Another limitation refers to the way in which descriptions of locations (where the species has been observed) are described in species records. For instance, we know that less than 5% of the location information associated with museum specimens (plants and animals) is described by means of geographic coordinates [56] (of known spatial reference system). Most of these data exists only in textual form, which makes it difficult to use it (directly) within a GIS, to map species distributions or to perform some sort of spatial analysis. We may have information where the spatial description of a species observation is of the form: "two hours in canoe up along the Río Napo" or "1 km west of San Llorenç de Montgai." Although software applications have been developed for georeferencing this type of textual description (e.g., refer to CAS' retrospective geo-referencing project [56]), this task is still far from simple. Some of the reasons are that localities may have changed their name over time and the same locality description may have been expressed differently amongst observers. For instance, the database at the Herpetology Department at CAS contained 47,000 unique locality descriptions in California. When they were able to remove those descriptions originally used to refer to the same place, the number of unique descriptions was reduced to 10,107 [56].

In the case where point data for observations is already available (with geographic coordinates), important metadata information may have been lost. For instance, the spatial and temporal scale, error estimation concerning the observations, the sampling scheme adopted in the field survey, or the interpretation of the data (e.g., a null value indicates that the species is known not to occur in a particular locality or a null value means that the species has not been observed) may all be unknown or be poorly described.

Instead of observations, we may be interested in descriptions of a species' ecological preferences already available in the literature or provided by expert opinion. Natural language descriptions are very varied and therefore standard expressions of species ecological characteristics are difficult to find. Below we provide a few examples, extracted from the African Mammals Databank [41] and from The Kingdon Field Guide to Africa Mammals [45].

Miopithecus talaponin is a strictly riverine species: its preferred

habitat is inundated forest, but it also occurs in dense riparian vegetation throughout woodland and cropland areas.

Theropithecus gelada prefers montane grasslands and shrublands between 1500 and 4000 m altitude. It also occurs in cultivated land, while it seldom enters forested areas.

Canis aureus lives in dry, open country from sea level to over 3000 meters, flourishes around villages and small towns.

Aonyx capensis is absent from several large rivers and many lakes where the a combination of factors exclude them, including parasites, predators (e.g., crocodiles) and particularly, waters that are too fast or otherwise unsuited to their prey or hunting techniques.

Smutsia temminckii occurs in both high- and low-rainfall areas with both sandy and rocky soils, in woodlands, savannas and grasslands. The main determinant of this species'range is an abundance of ants and termites of a few specific types.

Ammotragus lervia lives in desert hills and mountains, stony plateau (*hammada*) and the slopes of valleys (*wadis*) well away from mountains. They avoid the sand deserts(*ergs*), which seems to have acted as barriers between regional populations.

2.4.2 Species distribution models

GIS models for species distribution mapping, as described in section 2.3, fall within two main groups: inductive and deductive models. In both cases, the implementation of such models in a (semi-)automatic way presents many challenges from an information technology perspective.

Let us first look more closely at the deductive approach. To apply such a model, one first needs to have the ecological preference description stored in a system. From a data modelling perspective, this requires handling spatio-temporal information, relating species to different types of phenomena (such as vegetation and elevation), storing constraints concerning these relations (e.g., restrictive relations) and overall, attempting to standardize the descriptions without loss of information.

Closer scrutiny of examples extracted from the literature reveals what exactly we must model within a system for (semi-)automatic species mapping procedures. Ortigosa et al. [54] provide a *winter* distribution model for the Ibex (*Capra ibex*), an ungulate species, in the Adamello Natural Park Italy. This model relates four environmental variables (elevation, aspect, slope and vegetation). Values for each environmental variable separately are assigned a suitability score. For example, suitable elevation ranges for the Ibex are between 2400 and 2600 m and between 1200 and 1500 m. Highly suitable ranges are between 2200 and 2400 m and between 1500 and 1800 m. The optimum range is between 1800 and 2200 m. Similarly, suitability scores are provided for aspect,

slope and vegetation cover. Finally, the winter distribution map is generated by combining the scores for each variable with a formula, leading to suitability scores for all areas.

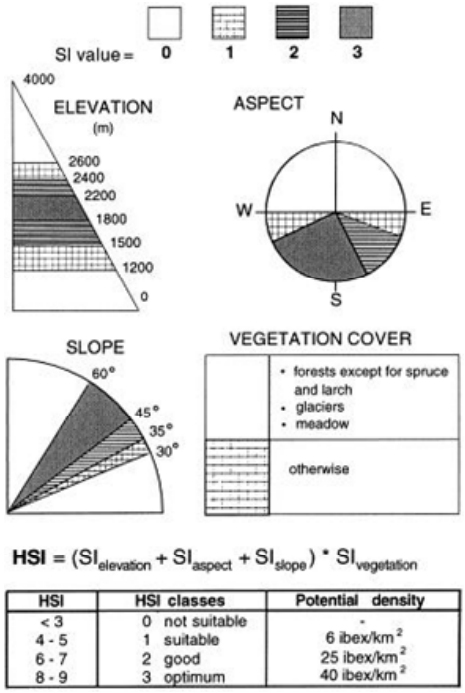

Figure 2.2: Model for assessing the winter areas of *Capra ibex* (54).

The challenge to model such ecological preferences from the combination of its constituent parts includes:

- how to relate the species to each of the environmental variables,
- how to store values (e.g., for the elevation variable),
- how do we expect to have the values made available, as a range or as single values,
- how are allowed values represented (for aspect, do expressions contain only values, are they given together with a unit, or do they even include information such as compass directions), and
- how to associate the vegetation values to a certain classification system.

Another example, in this case extracted form the Africa Mammals Databank project [9], relates the Gelada Baboon (*Theropithecus gelada*) to environmental variables, assigning suitability scores to values within each of them. For instance, this species prefers grasslands and grassland mosaics above 1500 m altitude as first choice. Secondly, the species prefers bushlands, but also forest and croplands above 1500 m altitude. It avoids, though, woodlands and all vegetation types below 1500 m altitude. This second example seems to be easier

to model than the first, but still we have to consider that the (semi-)automatic system should be able to handle both cases. Moreover, the reader is referred to the examples mentioned in section 2.4.1, realizing the complexity of accommodating those cases as well.

In an inductive approach, on the other hand, models are constructed from observations of the species, deriving the ecological preference by some sort of statistical analysis. Challenges in this approach arise from the decision of a particular statistical method for deriving the species ecological signature. This decision may be based on the type of response variable (e.g., the species presence) and its associated probability distribution in relation to the environmental variables affecting its distribution (e.g., Gaussian or Binomial) [34].

2.4.3 Spatial data required for the model

Any of the approaches described above require spatial data sets either for characterizing the ecological preference from base data for the species under study (in an inductive approach) and/or for generating the final distribution map (in both the inductive and deductive approaches). This means that (spatial) data has to be discovered, accessed and integrated for the analysis required in the mapping procedure.

Any search for suitable data sets faces several difficulties. One of the challenges that has received the attention in the geo-information community is that of *geodata interoperability* [64], which is defined as the ability to exchange information amongst users and systems. The reason is that geodata availability is high but the heterogeneity of the data makes it difficult to discover, assess its fitness for use, and integrate with other sources. The problems that arise from data heterogeneity can be grouped in three categories:

- differences in syntax — e.g., differences in data format;
- differences in structure — homonyms, synonyms or different attributes in database tables, and
- differences in semantics — e.g., intended meaning of terms in a special context or application [64].

A user may be interested to generate a distribution map for a particular region. How one expresses the area of interest, may vary between users. It may be a named geographic region, a named locality, but it could well be a particular area expressed by means of a string of bounding coordinates. The area of interest is an important parameter to take into account in search for data sets. Potential data sets that fall within the requested spatial extent may or may not cover the whole area (think of a request for a distribution map at a continental scale). In the latter scenario, this may require the assembly of several data sets to allow a complete cover of the whole area. The derivation of such a cover comes with all the limitations that are related to this type of procedure (e.g., issues of edge matching).

In a deductive approach, we have seen that the species ecological preference is assumed to be known a priori. The procedure to generate a distribution map,

therefore, requires spatial data sets that represent these environmental characteristics. This raises many semantic issues that need to be accounted for. For instance, a species may be related to a certain type of vegetation cover. The expert responsible for this entry in the ecological preference may have used a certain classification system. A search for data sets matching that particular description may not find spatial vegetation data sets that match the semantics used, but rather have been classified according to another classification scheme.

Another limitation refers to the spatial and temporal scale of the data sets for a given SMP. Descriptions in the ecological preference may require data sets at a specified resolution, or within a resolution range. For instance, the ecological preference for a species may state that it occurs close to large permanent water bodies but yet another species may be related to small streams only. The temporal scale may also influence the search for spatial data sets. For instance, a vegetation map (containing the characteristics stated in the ecological preference) from the year 1920 should not be used for attempts to mapping the current distribution of a particular species.

Assuming that potential data sets representing the characteristics in the ecological preference description have been found, important considerations also follow in discriminating amongst them. They may be different, for instance, in format (e.g., vector and raster), in the way they represent their phenomenon (e.g., point data or contour lines for elevation information) in resolution and in quality characteristics. This makes it even more difficult in choosing proper data sets for use in the SMP. Moreover, it illustrates that there is a need for automated detection and conversion in these cases.

2.4.4 Building the model

When building a model (either following the inductive or the deductive approach), the user may have to (possibly) process and combine the data sets obtained from the search. Data sets at different resolutions need to be brought to a common scale to perform the analysis and conversions should be applied to the data. A possible scenario is that we may be having different data sets with different metadata values, sometimes even in different data formats. Intermediate steps may require, for instance, coordinate transformations to a common spatial reference system, conversion from vector to raster and interpolation procedures (e.g., for elevation data represented as point features). These are the main challenges of this phase as the last step in the mapping procedure (model predictions) are implemented in the GIS either by a simple overlay or by writing macros in the case of more complicated models [34].

2.5 Summary

In this chapter we have briefly described the complexity of species mapping procedures as a whole. We first looked at the factors that limit species distributions. We have raised the importance of the spatial and temporal scale of analysis, as important elements to consider when studying such distributions.

In the second part of the chapter we have reviewed the main phases of SMPs, identifying the technical challenges in building a (semi)automatic system for species mapping. We have looked at these challenges from four different perspectives, considering as well the two different approaches (inductive and deductive) for generating species distribution maps.

We are aiming at building a (semi)automatic system for species distribution mapping steered by metadata. In the next chapter, we look at Description Logics, particularly the logic $\mathcal{SHIQ}$ for this purpose.

Chapter 3

Metadata and its formalisms

In this chapter, we take a closer look at what metadata is, what formal language(s) is available for defining/describing it, what are the formal semantics of such a language, and what pragmatic rules of thumb govern the use of the language.

3.1 Representing knowledge through metadata

Numerous schemes, languages or formalisms for knowledge representation — or whatever one would like to call them — have been proposed since Quillian's original work in the mid-1960s [57] on semantic networks. In such networks, as in most if not all follow-up work, an attempt is made to capture the meaning of relevant terms in a Universe of Discourse (UoD), through graph-like structures (nodes and edges) with predefined semantics. A node, for instance, may represent an object, a class of objects, an object's property, or even the value of an object property. Edges in such nets may represent relations between the phenomena represented by the nodes; for instance, that an object is a member of an object class.

It would be a humongous task to provide an overview of the proposals for knowledge representation (KR) formalisms published since Quillian's work. It is not so difficult to provide a short list of typical application domains for these formalisms. We can identify roughly three important areas:

Artificial intelligence KR has always been important here because computer applications were considered being able to act intelligently only if they possessed an internal knowledge base, which captured some human intelligence, that would drive the intelligent behaviour of the system. A seminal work was the one on the KL-ONE language [10]. Expert systems are the best known, and most successful exponents of this area.

Databases and Information Systems In designing databases and, more generally, information systems, conceptual data models have always been considered important. They are formal languages to describe, in an implementation-independent way, the data/information that will be managed in the system

under development. Early proponents of this line of work were the Entity-Relationship data model [15], and various semantic data models [40].

Software engineering As a somewhat wider domain than the previous area, SE too has had to deal with issues of properly capturing 'real world semantics' to make the software under design faithfully behave in that 'real world'. Especially object-centered models were designed with that objective in mind. The earliest of such models became the kernel of the Simula-67 language [22]. Later examples were presented for Smalltalk [8, 31], and obviously all exploded in the late 1980s and early 1990s with object-oriented models like the Object Modeling Technique (OMT) [48, 6]. The data modelling language en vogue these days, UML [58], is a direct descendant of OMT.

Lately, with internet access having become such an ordinary commodity, we have seen that a fourth area for knowledge representation formalisms is becoming important. We might call this area *information sharing across the internet*. More and more, companies, organizations and individuals are exchanging information, through data files, under circumstances where there has been no time to organize and define the information exchange beforehand. Consequently, one cannot expect there is a mutually agreed upon understanding of terms, or so to say, a shared object model. This is typical for use cases in e-commerce, but actually it is typical for internet operations at large. Information traffic on the internet is not so structured and organized, also because it offers such a wide range of information exchange possibilities.

What we therefore witness are attempts to make the exchanged information be *self-descriptive*. This means that whenever information is shared in cases where the model has not been a priori agreed upon, the information itself carries a (semantic) definition of terms that allows to understand the sent information (better).

Clearly, the formalisms for these semantic definitions require a fair bit of syntactic standardization themselves. The eXtensible Markup Language (XML) [11] can be understood as such a standard. It is itself based on ISO standard 8879, SGML, the Standard Generalized Markup Language, and forms a subset of it that has been devised for easy exchange across the internet. XML can be understood as a metalanguage, as it allows to define special purpose languages for document exchange. Such languages have been defined, and are being defined, for all sorts of domains. In our field of science, this includes the geodata domain, with the Geography Markup Language (GML) [20].

RDF, DAML+OIL [16] and OWL [3] are also XML-based exchange languages. They have a specific purpose as they are meant for exchanging semantic/ontologic information across the internet. Thus, they serve to define a terminology (ontology) in a formal way, so that it can be shared.

All the languages mentioned above are synthetic, non-natural languages. As such, they need to be provided with a formal semantics, i.e., an unambiguous definition of how to (mathematically) interpret the language's expressions. In recent years, *Description Logics* have surfaced as an important family of logic formalisms as they accommodate well the description of UoDs in terms of

objects, classes, their relationships and restrictions on these, for all mentioned application areas. A Description Logic (DL) is simply one of the many logic languages defined within this family.

In the sections below, we will pick one such DL and define it in full. It provides the mathematical theory behind such languages as DAML+OIL and OWL. We discuss the syntax, its semantics, but also some of the pragmatics related to using the language. Lessons learnt are immediately applicable to using the web ontology languages just mentioned.

3.2 Description logics

Description Logics is a large family of logical formalisms. All of them have in common a view of the world — to be precise: of the UoD under study — in which one describes classes and objects, as well as their properties. In DL terms, these are called, respectively, *concepts*, *individuals* and *roles*. When a roles displays a functional behaviour — i.e., is not an arbitrary binary relation — it is sometimes called a *feature* or *attribute*. The distinction between the latter two is made on the basis of the range of the function: either a set of individuals (feature) or a set of base values (attribute).

The rest of this section is devoted to explaining what DL is, and how it can be used. We do so by picking one example DL, known as $\mathcal{SHIQ}$, and discuss it in fair detail. The reason why we picked this logic is that it is (a) highly expressive, (b) it has a sound and complete axiomatics (proof system), and (c) it has been implemented in a public domain reasoner (RACER (Renamed ABox and Concept Expression Reasoner) [35]) that we wanted to use for the project.

It needs to be emphasized, however, that much of what is described for $\mathcal{SHIQ}$ below applies in fact to many DLs.

The logic $\mathcal{SHIQ}$ ('chique') also has an official, systematic name in which the experts can read its capabilities. That systematic, 'chemical' name is $\mathcal{ALCQHI}_{\mathcal{R}^+}$ ('Alc-choir'). This name indicates that $\mathcal{SHIQ}$ has basic DL ($\mathcal{AL}$) as its foundation, is extended with concept construction by negation ($\mathcal{C}$), allows qualified number restrictions ($\mathcal{Q}$), allows to declare role hierarchies ($\mathcal{H}$), has provision for declaring inverse roles ($\mathcal{I}$) and, finally, provides means to declare roles as transitive roles ($_{\mathcal{R}^+}$). Explanation of all these features is provided below.

3.2.1 Syntaxis of $\mathcal{SHIQ}$

Concepts

The UoD of a DL is made up of classes of objects, known as *concepts*. A concept can be understood as *the description of a collection of individuals*. Two *built-in concepts*, top ($\top$) and bottom ($\bot$), are always present. They are descriptions for, respectively, the collection of all objects, and the collection of no objects.

Further, the user of the logic may postulate an arbitrary number of *atomic concepts*; this is done simply by dropping a name. Examples could be: Clock, Car, Church, Challenge, Cinema, and Croatian. After such postulates, the

system will know of these concepts by name; obviously, not by intrinsic meaning, such as the fact that cinemas show movies from time to time.

A third and final way of describing concepts is by *concept construction*. This means that we use already available concept descriptions to syntactically build more complicated ones. For instance, if Cinema has been postulated before, we may write **not** Cinema to denote the concept representing all objects that are not cinemas. The various syntactic forms for concept constructions in $\mathcal{SHIQ}$ are summed up in Table 3.1.

Table 3.1: Concept descriptions allowed in $\mathcal{SHIQ}$. A denotes an atomic concept; C and D denote concept descriptions obtained using this syntax; R denotes a role description and S denotes a simple role description (for both of which see below).

C, D	::=	A	|	(atomic concepts)
		$\top$ | $\bot$	|	(built-in concepts top and bottom)
		$\neg C$	|	(concept negation)
		$C \sqcap D$	|	(concept intersection)
		$C \sqcup D$	|	(concept union)
		$\exists R.C$	|	(existential role quantification)
		$\forall R.C$	|	(universal role quantification)
		$\exists_{\geq n} S.C$	|	(at-least number restriction)
		$\exists_{\leq n} S.C$	|	(at-most number restriction)

Concept negation, intersection and union are the concept equivalents for the respective, well-known set operators. This will become clear also in our discussion of semantics.

Strictly speaking, after the inclusion of $\neg$, $\sqcap$ and $\sqcup$, there is no need to include top and bottom in the language as they have become definable using these constructors. For instance, $\top \equiv C \sqcup \neg C$ for any concept C. They are retained here because they form part of any DL, and because they are very handy shorthands.

We postpone the discussion of the other concept description possibilities – the final four of Table 3.1 — until we have introduced roles. But the reader should be aware that they denote concepts too; although they look like logic formulas with a truth value, they are not: they too denote sets of individuals.

Roles

Where a concept (description) denotes a collection of individuals, a role (description) denotes a binary relation between collections of individuals. This allows to define associations between concepts, and to build a semantic mesh. It helps to think of a role description as a denotation of a population of pairs (a, b) of individuals a and b; when the role name is R, we will also write $a\ R\ b$ for such a pair.

Atomic roles are roles postulated by the user. S/he just provides a name, and does (at this stage) not even indicate which concepts are involved in the role. In

$\mathcal{SHIQ}$, roles can be declared to be transitive — by intention. Obviously, a role R declared to be transitive allows transitivity inferences such as

$$x\ R\ y \wedge y\ R\ z \Rightarrow x\ R\ z.$$

Simple roles, as used in Table 3.1, are a special subset of the atomic roles, namely those that are non-transitive and that also do not contain a transitive role [35, which see for details]. This restriction was enforced to ensure that the logic does not allow undecidable theorems.[1]

Example atomic roles are hasVisited, hasVisitors, hasCrashedInto, three non-transitive roles, which might be intended to denote, respectively, relations between Croatian/Cinema, Cinema/Croatian and Car/ Church. The latter does, however, not follow from the role postulates.

Additional roles (role descriptions) can be syntactically constructed from existing concept and role descriptions: we call them *constructed roles*. In $\mathcal{SHIQ}$, there is only one additional role constructor, the role *inverse* operator $\_^-$. When R is a role with pairs $a\ R\ b$, by R^- we denote the role obtained from R with pairs $b\ R^-\ a$.

Other DLs provide further role constructors, some of which can be emulated in $\mathcal{SHIQ}$. Further constructors sometimes include role intersection, role union, role complement, and role composition — i.e., role chaining.

After this discussion of roles, let us revisit the last four concept description forms of Table 3.1, since roles are used there. With existential role quantification ($\exists R.C$), we denote the concept (population) of individuals that are related via role R with individuals of concept (population) C. Universal role quantification ($\forall R.C$) denotes the concept (population) of individuals that are related via role R *only* with individuals of concept (population) C.

At least number restriction ($\exists_{\geq n} S.C$) represents the concept (population) of individuals that enjoy at least n relations via simple role S with individuals in C. Similarly, at most number restriction ($\exists_{\leq n} S.C$) does so with an upper bound n on the number of relations. These number restrictions are commonly used in defining what we would call cardinality constraints on attribute values in a database context.

Axioms and assertions

The above syntaxis allows us to write up (advanced) concept descriptions. For modelling a UoD, however, next to postulating and defining concepts, we need to be able to define axioms about them. Two fundamental axiom formats are *concept inclusion* and *role inclusion*. They are written, respectively, as $C \sqsubseteq D$ (for concepts) and $R_1 \sqsubseteq R_2$ (for roles). With an axiom of this form, the concept C (the role R_1) becomes more strictly defined, as its population is declared to be a subset of that of concept D (role R_2).

[1] A language that allows to express undecidable theorems is itself called *undecidable*. An *undecidable theorem* is one that cannot be proven to hold (or not to hold) by an inference engine in a finite number of steps. Often such is shown through a correspondence with the *halting problem*. This problem, which attempts to prove that two computer programs have identical computational behaviour, is known to be undecidable.

A *concept hierarchy* (or *role hierarchy*) is a finite list of concept inclusion axioms (and role inclusion axioms). Another word for concept hierarchy is *terminology box*, or *TBox*. In $\mathcal{SHIQ}$, a TBox may also contain declarations of the type transitive R, which states that R is a transitive role.

With concept and role descriptions, we have obtained a framework that allows to describe populations — 'at the type level' – but not individuals of those populations — 'at the instance level'. Just like one can postulate the existence of atomic concepts and roles, one can also postulate individuals as concept members, and pairs of individuals as role members. Such postulates are known as *assertions*. Together, they form what is called the *assertion box*, or *ABox*.

Assertions come in two shapes. Individuals can be assigned to belong to a concept, in the notation $C(a)$. Pairs can be asserted to belong to a role, in the notation $R(a, b)$. For instance, we might write:

Cinema(ALHAMBRA)
Croatian(GORAN)
hasVisited(GORAN, ALHAMBRA)

Since DLs are especially used for writing up descriptions of concepts and roles, not individuals, the convention is to write the latter with all capitals. Differently named individuals are implicitly assumed to denote different individuals [35, as in], [39, but contra]. (But see issues on individual identity and references below.)

3.2.2 Semantics of $\mathcal{SHIQ}$

Before continuing our discussion of how to use $\mathcal{SHIQ}$ in describing a UoD, and then to exploit such descriptions, for completeness sake we provide here the semantics of the syntactic constructs introduced above. The source for this is [35].

The semantics of $\mathcal{SHIQ}$ are founded on a semantic universe $\mathcal{U}$, which can be understood as the set of all individuals, and associations between them, about which we may want to make statements. A formal semantics for the language is a mapping $\mathcal{I}$ between the syntactic constructs and the semantic domain. The rules of semantics are laid out in Table 3.2.

3.2.3 Additional syntax

Various DLs, and the reasoners built for them, allow additional syntactic constructs. There may be various reasons for this: some constructs may be just very handy shorthands for more elaborate expressions that are already allowed; others may similarly not extend the language's expressiveness, but may have a positive effect on the performance of reasoning.

As a simple example, the RACER system [36] allows to directly express concept equality in syntax a la $C \doteq D$, providing a shorthand for the two concept inclusion statements $C \sqsubseteq D$ and $D \sqsubseteq C$.

Table 3.2: Semantic rules for the description logic $\mathcal{SHIQ}$. Terms used as in Table 3.1; $\mathcal{U}$ is the semantic universe, $P^{\mathcal{I}}$ is the interpretation of P in $\mathcal{U}$, and $\#$ is the count function on sets

A	$A^{\mathcal{I}} \subseteq \mathcal{U}$
C	$C^{\mathcal{I}} \subseteq \mathcal{U}$
$\top, \quad \bot$	$\top^{\mathcal{I}} = \mathcal{U}, \quad \bot^{\mathcal{I}} = \emptyset$
$\neg C$	$(\neg C)^{\mathcal{I}} = \mathcal{U} \setminus C^{\mathcal{I}}$
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$R, \quad S$	$R^{\mathcal{I}} \subseteq \mathcal{U} \times \mathcal{U}, \quad S^{\mathcal{I}} \subseteq \mathcal{U} \times \mathcal{U}$
$\exists R.C$	$\{\ a \in \mathcal{U} \mid \exists b : (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\ \}$
$\forall R.C$	$\{\ a \in \mathcal{U} \mid \forall b : (a,b) \in R^{\mathcal{I}} \Rightarrow b \in C^{\mathcal{I}}\ \}$
$\exists_{\geq n} S.C$	$\{\ a \in \mathcal{U} \mid \#\{\ (a,b) \in S^{\mathcal{I}} \mid b \in C^{\mathcal{I}}\ \} \geq n\ \}$
$\exists_{\leq n} S.C$	$\{\ a \in \mathcal{U} \mid \#\{\ (a,b) \in S^{\mathcal{I}} \mid b \in C^{\mathcal{I}}\ \} \leq n\ \}$

Another example is the use of roles as attributes for individuals belonging to a concept population. Such attributes are partial or total functions: they assign just one value to the individual. The attribute hasFirstName, for instance, can be combined with the atomic concept Croatian as follows

$$\text{Croatian} \sqcap \exists_{\leq 1} \text{hasFirstName.String}\,,$$

constructing a concept description for Croatians who have at most a single first name, which has a string value. This states that hasFirstName is a *partial* function; if we want to make it total, we could have added $\exists_{\geq 1}$hasFirstName.String to the concept description, using another concept union:

$$\text{Croatian} \sqcap \exists_{\leq 1} \text{hasFirstName.String} \sqcap \exists_{\geq 1} \text{hasFirstName.String}\,.$$

One important, more fundamental extension of the logic is in the provision of so-called *concrete domains*. A concrete domain is best viewed as a value set to be used for the range of roles that represent simple attributes. In this way, integers/reals, some of their arithmetics, strings, and for all three, some of their predicates can be embedded in the language. Such inclusion allows to express integrity constraints of a certain class. The use of the String name above was already a precursor of this.

We will discuss RACER's support for concrete domains below.

Knowledge bases

When using a DL, one models the UoD by defining a *knowledge base*. A knowledge base is a combination of a TBox with an ABox. The knowledge (statements) about concepts and roles is captured in the TBox, whereas the knowledge about specific individuals is represented in the ABox. In database termi-

nology, we may compare the TBox to a database schema, and the ABox to the database contents, i.e., the collection of records stored.

Typically, a TBox starts with a declaration of the atomic concepts and roles that are of importance in the UoD. From these, more complex concept descriptions can be obtained, using the syntax discussed above. Then, concept inclusion axioms are used to denote concept subsumption. Depending on the DL in use, the same can be done for roles, to indicate role inclusions. We will see examples below

An important issue to be aware of is the assumption for interpreting the combination of TBox and ABox. Where standard databases — and their query languages that usually provide negation in their syntax — make the *closed world assumption*, DLs do not do so. They are built on the *open world assumption*. The difference between the two can have far-reaching consequences. Essentially, if the inference engine cannot prove that an individual is not a member of a concept, such does not constitute a proof that it is a member of the concept's negation. There is, we could say, 'middle ground'. Some DLs provide an additional modal operator $\mathbf{K}$ for this situation, standing for "the knowledge base knows," and allowing to make the distinction between what is and what is not in, or derivable from, the knowledge base.

A simple example may further illustrate this. Suppose by declaration we indicate that GORAN is an individual of the constructed Croatian concept on page 33, then, even when we do not provide a concrete value, the DL reasoner will be able to infer that GORAN has a name, which is a string value. It merely happens to be unknown at present.

The assumption of an open world makes sense for DL: it is after all based on descriptions, not on individuals so much. But it is also true to state that for users with a database background, where a closed world assumption is the norm, it may pose intuitive challenges.

3.2.4 Pragmatics of using DLs

In this section, we devote some space to issues related to how the tools provided by a DL can be used to write up a description of a UoD. This cannot possibly be a complete discussion as such pragmatic issues depend much of the context in which the DL is used. Nevertheless, some general rules of thumb can be provided, and are useful to be aware of. Some of the heuristics discussed below originate from [7], others were derived whule conducting this project.

Throughout this part of the text, $\sqsubseteq$ stands for subsumption (also known as inclusion), and $\doteq$ stands for definitional equality, both for concepts and for roles.

Atomic concepts Populations of individuals perceived in the UoD can be provided with a name; this especially makes sense if the population does not need to be defined by construction, i.e., if its definition requires no structure, and postulation is good enough. Hence the various atomic concepts illustrated above.

Some concepts can best be first postulated and then still be defined, for instance as in:

$$\mathsf{VisitingCroatian} \doteq \mathsf{Croatian} \sqcap \mathsf{CinemaVisitor}\,.$$

Whether atomic concepts have a subsumption relation is another matter of modelling importance, which can be stated with an inclusion axiom like

$$\mathsf{VisitingCroatian} \sqsubseteq \mathsf{Croatian}\,,$$

which actually, after the previous definition, is a superfluous statement.

Assigning roles In principle, any postulated or defined role (or feature, of course) can be used to relate two arbitrary individuals. Roles are not, by construction, restricted to relate individuals of two *indicated* concepts. But this can be achieved, if there is a modelling need, through inclusion axioms. We therefore first look at range restrictions.

For instance, the following axiom states that CinemaVisitors can have visited only Cinemas. There is nothing else that they can have visited. (Observe that they may not have visited anything.)

$$\mathsf{CinemaVisitor} \sqsubseteq \forall \mathsf{hasVisited}.\mathsf{Cinema}\,.$$

In direct terms, this can be read as "any cinema visitor is a member of the population of individuals that have only visited cinemas." It can indirectly be read as a statement saying that a cinema visitor has a role with values restricted to cinemas in hasVisited. It is a *concept-specific range restriction* on the role, in this case for the concept CinemaVisitor.

Suppose we wanted to state that the hasVisited role, applied to whatever individual, always provides Cinema individuals, and nothing else. This would be a *universal range restriction* for the role. It can be stated as follows:

$$\top \sqsubseteq \forall \mathsf{hasVisited}.\mathsf{Cinema}\,.$$

It can be read as "any individual (belonging to the top population) can only have visited cinemas."

Regarding the range of a role, we might also want to state *cardinality restrictions* on the number of individuals assigned to an individual from the domain. A classical example in this is when the role denotes a total function from domain to range. Suppose we only want to know the last cinema visited, then we can make do with a total function hasVisited:

$$\begin{aligned}\mathsf{CinemaVisitor} \sqsubseteq\ & \forall \mathsf{hasVisited}.\mathsf{Cinema} \sqcap \\ & \exists_{\geq 1} \mathsf{hasVisited}.\mathsf{Cinema} \sqcap \\ & \exists_{\leq 1} \mathsf{hasVisited}.\mathsf{Cinema}\,.\end{aligned}$$

Each cinema visitor must have visited at least one and at most one cinema. Other *range cardinality constraints* could obviously have been imposed. The above triple conjunction — defining, so to say, a non-null feature — is so typical that some DLs have a shorthand for it: **the** hasVisited Cinema.

We can model similar constraints on *domains* of roles. Observe first that since the domain of a role is the range of the inverse role, we can simply apply the above trickery to the role's inverse. If we want to state that the role

hasVisited, in the context of cinemas, applies only to CinemaVisitor, and not to individuals outside of that population, we could state this as

$$\mathsf{Cinema} \sqsubseteq \forall \mathsf{hasVisited}^{-}.\mathsf{CinemaVisitor}\,,$$

cinemas can only have been visited by cinema visitors. We might even replace Cinema with $\top$ to phrase the universal domain restriction that anything can only have been visited by cinema visitors, no matter which context. But this can be stated differently as well:

$$\exists \mathsf{hasVisited}.\top \sqsubseteq \mathsf{CinemaVisitor}\,.$$

Read this as "any individual that has visited something ($\top$) must be a cinema visitor."

Cardinality restrictions on the domain can, and should, be handled just like those on the range, making use again of the role's inverse. For instance, a statement saying that at most 200 people can have visited a cinema, is written as

$$\mathsf{Cinema} \sqsubseteq \exists_{\leq 200} \mathsf{hasVisited}^{-}.\mathsf{CinemaVisitor}\,.$$

Concept descriptions and definitions Providing a concept characterization through a description is different from providing its definition. The first we do via an inclusion axiom ($\sqsubseteq$), the second via a definition ($\doteq$). An inclusion axiom provides *necessary* conditions, as can be illustrated with a typical (partial) concept characterization for CinemaVisitor:

$$\begin{aligned} \mathsf{CinemaVisitor} \sqsubseteq\ & \mathsf{Human} \sqcap \\ & \forall \mathsf{hasVisited}.\mathsf{Cinema} \sqcap \\ & \ldots \end{aligned}$$

Such a characterization, during inferencing, allows to decide that some individual cannot be a cinema visitor, because the individual is not human, or has visited something else than a cinema. The concept characterization leaves a certain level of vagueness around the boundary of its population.

With a definition, *sufficient and necessary* conditions are provided for membership test of the population. For instance,

$$\begin{aligned} \mathsf{CinemaVisitor} \doteq\ & \mathsf{Human} \sqcap \\ & \forall \mathsf{hasVisited}.\mathsf{Cinema} \sqcap \\ & \exists_{\geq 1} \mathsf{hasVisited}.\top \end{aligned}$$

states that any human who has visited only cinemas, and at least one, is *by definition* a cinema visitor. This gives us a true membership test, and thus a way to prove that some individual indeed is an individual of this class.

Modelling subtleties arise when one wants to distinguish between essential and incidental properties of individuals. The difference between them is a matter of proper sensitivity to the modelling context. Generally, one should avoid including incidental properties in definitions, and instead declare them through an inclusion axiom.

For instance, we may be aware of the fact that all the cinemas under study only air movies with a viewer characteristic of "above age of 16," from which we could incidentally infer that all cinema visitors will/must be above that age. It would be unwise to include this in the definition of CinemaVisitor.

Concept hierarchies Where one wants to make use of subclassifications of concepts — using of course inclusion axioms — one typically needs to indicate whether subconcepts of the same superconcept will be *disjoint* in their populations or not, and also whether the populations of all subconcepts together *span* the population of the superconcept.

Disjointness of concepts can be expressed using concept negation. Suppose that FilmHouse, CineStar and OpenAir are three mutually disjoint subconcepts of Cinema, then such can be stated in the following style.

$$\begin{aligned} \mathsf{CineStar} \sqsubseteq\ & \mathsf{Cinema} \sqcap \\ & \textbf{not}\ \mathsf{FilmHouse} \sqcap \\ & \textbf{not}\ \mathsf{OpenAir} \sqcap \\ & \ldots \end{aligned}$$

In case one wants to state that all subconcepts together span the superconcept, such can be achieved by a definitional statement for that superconcept. This is independent of stating disjointness:

$$\begin{aligned} \mathsf{Cinema} \doteq\ & \mathsf{FilmHouse} \sqcup \\ & \mathsf{CineStar} \sqcup \\ & \mathsf{OpenAir}\ . \end{aligned}$$

More can be said about constructing hierarchies of concepts (and roles), but a fuller discussion is beyond the purpose of this text. Details can be found in [32, 66].

N-ary roles As we know from Entity-Relationship data modelling, there exists some need for ternary, and sometimes even quaternary, relationship types. Most DLs do not provide roles for such types, but they can be had relatively easily by considering the thus wanted relationship type to be a concept, instead of a role. This mechanism is known as *reification* [7]. A ternary relationship type (between three concepts) is considered itself to be a concept, flanked with an additional three (binary) roles with the original concepts. There is nothing magical about such modelling schemes, and all follows the techniques discussed above.

Materialized concepts A final important issue in conceptual modelling, and certainly so when using DLs, arises when in the same UoD any individual of a concept's population can also be viewed as itself representing a population. That is, each individual somehow stands for a set of individuals again (of a

subtly different kind). In practice, such situations often lead to confused modelling and the dilemma of whether to recognize a notion as a concept or as an individual.

The prototypical example in database modelling has always been the notion of committees inside an organization. If there are many committees, and they have dynamic life cycles, one will certainly tend to consider them as individuals. At the same time, a committee constitutes a collection of staff members, themselves typically also individuals in the DL sense.

Another example, adapted from [7] for our cinema UoD, is formed by different views on movies. Most of us will know the movie "One flew over the Cuckoo's Nest," a 1975 movie by Milos Forman. One movie, so one individual. It may have been aired for multiple weeks in various seasons, each season airing at a fixed ticket price. One season airing, one individual. Every evening it was aired, we may have registered the number of visitors. One evening airing, one individual. Still, Forman's movie, any season airing, any evening airing: they have various characteristics in common, and one might be fooled to believe that for that reason they belong in the same taxonomy, for attribute inheritance purposes, for instance.

The relationship between these three individuals, and their concepts behind, however, is not one of concept inclusion. Rather, there is a notion of *materialization* between them. The first notion being 'less material' than the last. Such distinction should be carefully noted, and properly represented in one's model of a UoD, for instance, using a properly defined materialization role, or even multiple roles, possibly using subroles.

3.2.5 The RACER system

RACER is a knowledge representation system; its acronym stands for Renamed ABox and Concept Expression Reasoner. It provides reasoning services over multiple TBoxes and multiple ABoxes. Intuitively, this means that one pose queries about the concepts as well as about the individuals. One can also perform manipulations on TBoxes and ABoxes. RACER implements the $\mathcal{SHIQ}$ logic; it even extends it a little further, by supporting reasoning over some concrete value domains with the following:

- constraints on mimimum/maximum integer values,
- linear polynomials used in equations with reals or cardinals,
- nonlinear polynomials used in equations with complex numbers, and
- (in)equality tests of strings.

RACER as such is a reasoning server; it doesn't have a nice user interface, but supports various input interfaces that can be used from other user interfaces, or from one's own developed application. One input interface is XML-based and supports the use of RDF, RDFS, DAML and OWL. It also has an HTML-server interface based on a knowledge representation standard known

as DIG. Furthermore, RACER can read various knowledge representation file formats directly.

RACER is, however, most easily addressed from a specially designed user interface, which is what was done in this project most of the time. Specifically, we used Protégé for building ontologies in OWL (and other formats), and we used RICE for (meta)data manipulations on the knowledge base. Both of these systems are capable of connecting to the RACER system. The first allows to perform certain satisfiability checks, the latter does that too but provides a command line interface to RACER that can be used for arbitrary function calls and queries. We discuss both systems more thoroughly in Section 3.2.6 below. In Section 3.2.7 we come back to various knowledge manipulations that are possible with RACER.

3.2.6 User interfaces to RACER

In this project, I have made use mostly of five types of software:

Relational DBMS Specifically, MS Access.

GIS software Specifically, ESRI Arc/Info.

Ontology & knowledge base editor Specifically, Protégé 2000.

Knowledge base reasoner Specifically, RACER.

GUI to the knowledge base reasoner Specifically, RICE.

Some of these software packages were fully developed, mature, commercial packages, viz. the DBMS and GIS. Others are public domain, and are currently being developed, mostly in academia, and have not fully matured yet. The last three in the list above qualify as such. Consequently, it is with these software packages that sometimes unexpected events and errors occurred, which could not always be explained. For instance, over the execution period of this project, RACER went from version 1.7.7. to version 1.7.18, undergoing substantial updates, and somewhat improved stability. Just one week before submission deadline of the thesis, RACER became equipped with a query language (RQL), which we had missed during our earlier work. Then, when we found out that some type of queries could still not be posed, the implementers, when so asked, extended RQL with a primitive that did allow our queries. (These queries allowed us to discriminate between cases where an individual can be proven to be related via a role, from cases where actually a role filler exists in the ABox. See our discussion on the open world assumption below.)

We provide here a concise overview of why these software packages were used; a fuller discussion of actual data management work is provided in Chapter 4.

The DBMS was used to collect the species data from the AMD data set, analyse the species' ecological preference settings, and derive the rules that had been applied in determining these values. This allowed us to compact the largish data files for each species, into a small rule set — partly associated with

species, partly with the model metadata — included in the knowledge base that was built later.

The GIS software was used to emulate the outcome of the script configuration, so that we could run the actual spatial data operators on the GIS platform.

The knowledge base editor was used to create or adapt a number of ontologies needed for the project: the species ontology, the spatial data set ontology, and the model (script configuration) ontology. From this editor, an OWL-DL file was generated for each ontology.

RACER was used to import these knowledge base files, to convert them to a simpler format (the racer file format), perform various checks on the knowledge, and to run various queries at TBox and ABox levels against.

The RACER GUI, finally, made it easier to carry out all of these knowledge base manipulations.

Protégé and RICE were used to access the functions of RACER. Their use is fairly widely accepted, and although they are still under development, they have been put to use for multiple types of application. Other GUIs to RACER do exist, such as OilEd, however, we have not put much effort in making a comparison between all the systems.

Protégé

Protégé is an ontology and knowledge base editor, developed by Holger Knublauch at the Stanford Medical Informatics group at Stanford University. It provides a flexible and fairly easy to use environment for building up knowledge bases, also those founded on DLs. Its more recent versions provide support for OWL files.

This system has a highly extensible set-up, allowing others to add to the GUI other graphical widgets for specialized purposes.

Protégé connects to the RACER system primarily for classification purposes and consistency checks. Its extensible interface is, in principle, open to more uses of RACER as the reasoner behind the system.

The Protégé GUI uses six main tabs: Classes, Properties, Individuals, Forms, Queries and the Ontology Metadata tab. We briefly describe each tab and its constituent panes.

Classes Tab This Tab is a single window in which one can view, create, and edit concepts in an ontology. It contains four panes: the *Inheritance pane* displays the subconcept/superconcept relationship of the knowledge base as a tree. From this pane, one can create and delete subconcepts and superconcepts, and access extra windows to view and edit concepts and superconcepts; the *Class metadata pane* allows to label concepts in different languages and store comments about them; the *Properties pane* displays the direct and inherited roles for the selected concept. From this pane one can view, create, delete and add new roles, and the *Logical definition pane* allows to view and create/combine restrictions (necessary and necessary and sufficient conditions on concepts) that must hold for individuals of a concept. When the ontology makes use of subclassification of concepts, from this pane one can also express disjointness of concepts.

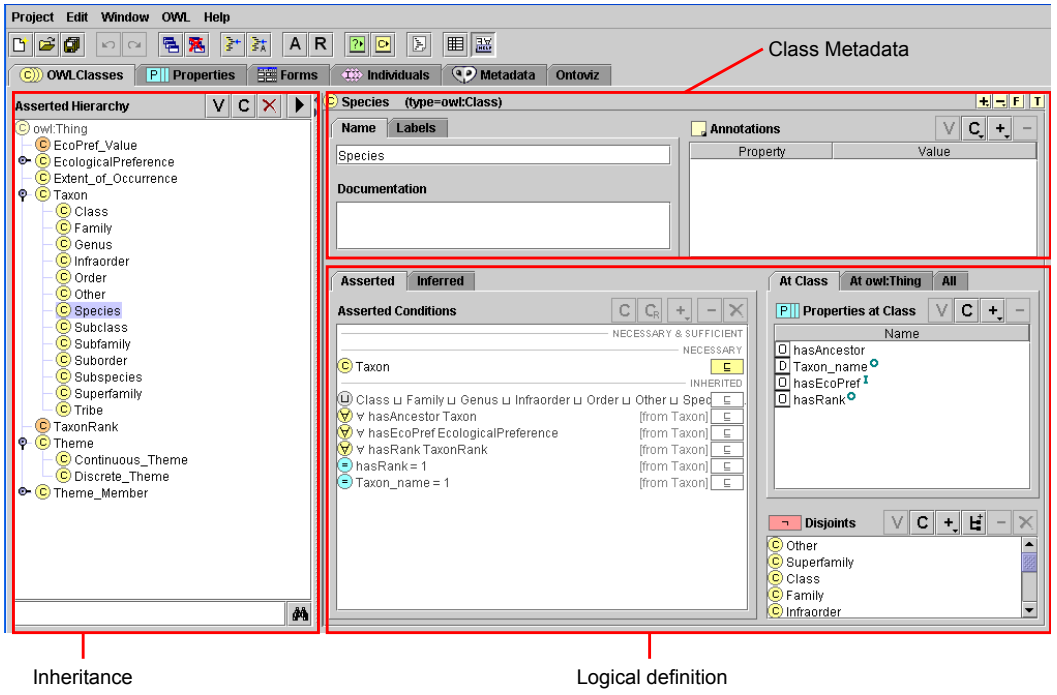

Figure 3.1: Graphical User Interface of Protégé displaying the *Class Tab*. The tab contains three panes: the *Inheritance, Class metadata* and *Logical definition* pane.

Properties Tab Provides a single window in which one can view, create and edit roles. These can be be defined independently, or attached to a concept. One can define the domain and the range of roles, their characteristics (e.g., transitive and functional roles) and the restrictions on them.

Individuals Tab From this window one can view, create, and edit individuals belonging to a concept. When certain individual is selected, a form is displayed to view and edit the roles which apply to that individual.

Forms Tab Provides a window from which one can create an edit the layout of forms that appear in the Individuals Tab.

Queries Tab Provides a window in which one can locate individuals from the knowledge base based on the values of one or more roles. One can construct queries by selecting a concept, role, a criterion and a value (based on the role's domain and range type). One can also combine and save the resulting queries.

Ontology Metadata Tab Provides information regarding the Namespace Prefixes used in the ontology as well as a set of panes where individuals of the knowledge base may be stated to be mutually distinct (this is important as OWL does not assume that individuals have one and only one name).

In this project we also used the Tab *Ontoviz*. This plug-in allows to generate

graphs from the knowledge base, displaying elements such as concepts, subconcepts, roles and individuals. This tab has been used to visualize parts of the ontology and to generate some of the illustrations included in the theses work.

RICE

RICE is the RACER Interactive Client Environment, developed by Ronald Cornet, department of Medical Informatics, Academic Medical Center, University of Amsterdam. It provides a simple and straightforward interface to RACER. Its GUI is depicted in Figure 3.2.

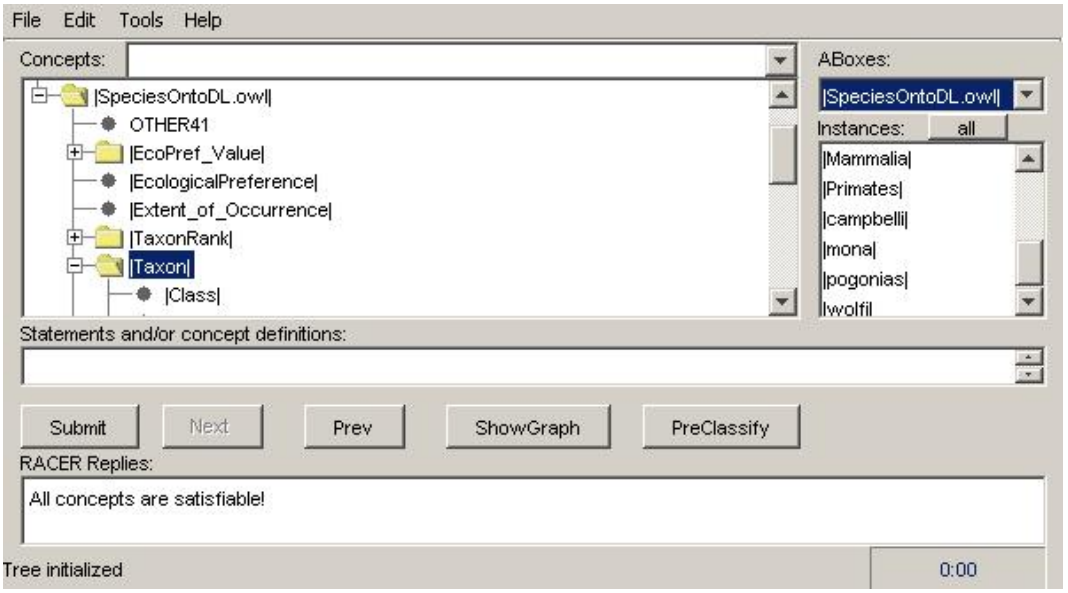

Figure 3.2: Graphical User Interface of RICE. Top left is a tree representation of the TBox; top right is a list representation of the ABox; the middle window is the command line; the bottom window is RACER's reply window.

With RICE, one can load one or more TBoxes, and one or more ABoxes. RICE ensures, obviously, that these data sources are loaded in RACER, such that it can subsequently manipulate and query the (meta)data through its command line interface. Upon loading a TBox, the satisfiability of its concepts is automatically verified and commented upon. A fully satisfiable TBox can subsequently be used for further knowledge manipulations. For that purpose, RICE offers a command line window and a command feedback window. An overview of available commands is provided in Section 3.2.7.

RICE itself is a plain Java archive, and a good start for study for anyone wanting to develop a project for building a specialized application that makes use of RACER as a reasoning engine in the background.

3.2.7 Reasoning with data and metadata

The most important advantage of using a DL-based reasoner is the seamless integration of data, metadata and schema information. We refuse to provide clear-cut definitions of these, but add that — in the approach taken here — data and metadata are typically covered in the ABox(es), while the schema information is captured in the TBox(es).

In the sequel of this section, we first discuss two important considerations for reasoning with DL-based descriptions, after which we look at various functions available in RACER in support of reasoning.

The open world assumption

All reasoning systems for DLs are built on, and make use of, the open world assumption (OWA). This has a number of consequences that may surprise the unaware. The reason that these systems take this approach is because of their focus on reasoning over description terms, i.e., over intentionally defined sets, not over explicitly enumerated sets of individuals. In such a context, th use of a closed world assumption is not very productive.

The OWA represents a reasoning mode that can be summarized as "statements in the knowledge base are true, statements derivable from those in the knowledge base are true, and all other statements may or may not hold." I.e., the latter statements are not considered not to hold. We assume here that the knowledge base is consistent: it does not contain or allow deductions that are falsehoods. A consistent knowledge base allows one or more non-empty worlds in which all its statements are true. (An inconsistent knowledge base does not.)

The OWA applies to both simple facts, such as concept and role membership, as to more complicated logical statements, such as attributions of characteristics to concepts and roles. Provability is an important notion in the implementation of the OWA.

For instance, is we would ask the system to enumerate the members of some concept, it would list those individuals for which it can prove their membership. Such a proof may be simple, as it follows from a concept membership statement "i is in C," or it may the result of a deduction: "since i has a child k, it is a member of Parent."

Additions to the knowledge base can only be made if they are consistent with the statements already in there. For instance, we cannot state that i is not a member of Parent if that membership for i can already be inferred from the knowledge base.

Likewise, querying the knowledge base can have its surprises, especially for those with a database background, where the closed world assumption is the prevailing model. A query asking for individuals that are member of a concept results in a list for which their membership van be proven; a query that asks for individuals not member of a concept results in a list of those individuals for which one can prove they are not a member. Such a proof could for instance come from the deduction that an individual is (provably) a member of some other concept, which can be shown to be always disjoint from the first concept.

The unique name assumption

Individuals with different names are different individuals, by this assumption (UNA). Also, in the query language, where variables are used, different variables cannot concurrently be assigned the same fillers: they too denote different individuals.

Especially when using the query language, one should always be aware of cases where the possibility may arise that different variables may have identical fillers. This may lead to a need of rewriting the original query.

Knowledge base manipulations

The various functions that RACER offers for knowledge base manipulation fall into four broad categories:

Loading and unloading of a knowledge base

Making declarations

Setting the reasoning mode

Evaluating and querying

Retrieval

We discuss each of these below. Upfront, it is important to understand that RACER allows to operate on multiple TBoxes and ABoxes, and many functions provide a parameter to indicate on which box the function should be carried out. There is a default box notion for both TBoxes and ABoxes.

RACER also offers support for concurrent access by multiple users; we do not pay attention to this feature.

(Un)loading a knowledge base Under this heading fall a number of functions that allow to load into RACER an external file defining a knowledge base (i.e., TBox and ABox). Various file formats are supported: racer, daml, owl. One can also perform some functions in support of the use of ontologies available elsewhere on the internet. A knowledge base can also be saved to a file in one of various formats.

Under the term TBox management fall a number of functions that one can use to operate on the TBox object specifically. They include: setting and saving the current TBox; reading a TBox from a file into memory; setting its signature, discarding it from memory; cloning it; finding a TBox in memory; and asking for the list of ABoxes associated with the current TBox.

Under the term ABox management fall an equivalent set of functions, now on ABoxes.

Making declarations Another batch of functions allows to interactively make additional declarations; these will be added to the current TBox or ABox. Concept subsumption, concept equivalence and disjointness can be stated; primitive concepts can be postulated, and concept terms be defined. Similarly, various role issues can be declared: primitive roles and attributes, additional characteristics of existing roles such as symmetry, reflexivity, transitivity their inverse, their domain and range, and the role by which they are subsumed. Functions also exist to make the knowledge base forget again about earlier made declarations. Testing also exist to determine whether some role has such characteristics. Finally, concrete domain roles can be declared.

All of the above functions operate on the TBox, but ABox functions also exist for declarations. One can assert or forget an instance of a class, and one can assert or forget a role instance. Concrete domain assertions are also possible.

Setting the reasoning mode One implementation of RACER, the LISP version, allows to set the reasoning mode to either lazy or eager evaluation. In lazy evaluation, any, possibly recursive, function used in reasoning is only evaluated as far as it needs to be for the purpose of the reasoning process. This results typically in faster reasoning processes, which, however, do use up larger amounts of main memory (as partial function results are kept much longer in main memory).

Evaluations and queries The true reasoning is performed with this batch of functions. One can test whether a concept is satisfiable, i.e., whether it allows a non-trivial (non-empty) model. One can test whether one concept (term) subsumes another, whether two concept (terms) are equivalent, or denote disjoint populations.

Similarly, tests exist for role subsumption and equivalence, role transitivity and reflexivity, and whether a role displays functional behaviour — i.e., whether a role is a feature. Such tests, both for concepts and roles, are always tests on the intention of the concept or role description, not on the actual individuals currently forming their population.

The TBox evaluation functions allow to classify the TBox, i.e., to compute the hierarchies of concepts and roles, test for satisfiability of all concepts, and test whether cyclic inclusion axioms have been used.

ABox consistency can also be verified, and implicit role fillers can be made explicit.

Some available ABox queries are the test for its consistency, whether some individual is an concept or role instance, whether two individuals are related by a role, and whether a constraint is entailed by the ABox. The term ‘query’ is here more used as a request about characteristics of the ABox, whereas ‘retrieval’ typically is a request for a set of individuals in the ABox meeting some condition.

Retrievals Again, we discriminate between TBox and ABox retrievals. Functions of the first kind (TBox retrievals) typically ask about structure of the con-

cept hierarchy: equivalence, ancestors, children for both concepts and roles. TBox retrievals always are about concepts and roles only, not about individuals.

ABox retrievals are much more about the relationship between concepts/roles on the one hand, and individuals on the other hand. Thus, one can ask for an individual's direct concepts, its most specific atomic concepts, all of a concept's or role's instances, et cetera.

On the basis of this set of TBox and ABox retrieval functions, the RACER Query Language (RQL) was defined and implemented, and became available early April 2004. A full discussion of this language is therefore somewhat difficult to provide, as the language's definition seems still to undergo important changes, even some suggested by us. The fundamental query notation is:

```
(retrieve (variable-list) (query-body))
```

The variable list mentions zero or more variables that should be matched against individuals in the ABox. The query body defines the condition of matching. It allows the use of concepts, roles and concept and role terms. It also provides various logical connectives like `and`, `or` and `not`. The query body allows introduction of further variables, and this gives rise to an important issue of use of the OWA. When these extra variables are introduced together with a `not` operator, that operator may be in front of, or after the variable introduction:

```
a:  (variable-list) (not P)
b:  (not (variable-list) P)
```

These two notations have drastically different semantics. The first asks for those individual fillers for the provided variables for which `(not P)` can be proven. The second asks for those individual fillers for the provided variables for which `P` cannot be proven. The latter set is a much larger set, normally.

This concludes our discussion of the reasoning capabilities of the RACER system.

3.3 Summary

In this chapter, we have taken a tour through the group of formalisms known as Description Logics. We specifically focused on $\mathcal{SHIQ}$, a rather powerful such language, that forms that theoretic foundation for various well-known web languages.

We have also looked at how DLs can be used for describing (modelling) UoDs, both syntactically, and pragmatically.

Finally, we took a more in-depth look at RACER, a reasoning system that allows to reason over descriptions in $\mathcal{SHIQ}$.

Chapter 4

SMP knowledge representation

In our earlier discussion, we saw that Description Logics (and particularly the $\mathcal{SHIQ}$ formalism) allow to describe a Universe of Discourse in terms of concepts, which are viewed as populations of individuals, and roles, which represent binary relationships between individuals. This knowledge representation can then be used by reasoners for machine interpretation and automatic inference [21].

In this chapter, we look at the application of Description Logics in our UoD: the automation or semi-automation of species mapping procedures. Any species distribution mapping procedure is, in the most general and simplistic way, determined by a combination of

Data (D)–Method (M)–Output (O).

The D component comprises the available or required data sources for obtaining a specified output O (e.g., the potential presence or absence of a species within a region), while M represents the method used to generate output O with data D (e.g., a complete script made of computational steps, fuelled by elements of the D component).

Using the $\mathcal{SHIQ}$ formal semantics and the language OWL (Web Ontology language) we build ontologies for the D component of the SMP. The M and O components are explained in Chapter 5. We start by describing the base data used in this project (from the African Mammals Databank [9]), followed by the ontologies we built for the species that we work on, as well as for the types of spatial data set that we expect to work with. We use the editor Protégé, together with an OWL plug-in for this purpose.

4.1 SMP data

4.1.1 Base data

Somewhat arbitrarily, we picked a group of mammals as our target experimentation group. This group was the subfamily of the *Cercopithecinae*, compris-

ing 46 monkey species. The AMD project [41], through its published materials (book, CDs and website) provided data files for each species with ecological preference data. Essentially, these data files provided a suitability score for each combination of vegetation class and land cover class occurring in the area of extent of the species. Originally, there were 98 and 123 classes for these ecological themes, respectively. Since not all combinations of vegetation and land use typically occur, the data files contained on average 400 suitability scores for each species, out of a theoretical number of 12,054. Moreover, the AMD method identified some species as being water-dependent. In the method, these species, for certain vegetation or land cover classes, were assigned bonus suitability scores.

For all 46 species, we built up a systematic taxonomy. Since the AMD method appeared to be so methodical, we were not willing to continue having the average 400 scores as base data. Rather, we decided to analyse the given data and retrace the formulas that were used to obtain it.

In [41], these formulas are discussed to a fair level of detail, however, for the outsider, after study of the provided species data, there remains some degree of freedom on the formulas applied. From the description given it is clear that for any species suitability scores are assigned separately for the themes. A fairly simple formula then allows to compute the resulting suitability score. But that all data seemed to follow such a pattern.

4.1.2 Data processing

We first discuss how we dealt with land cover data. Scores on land cover classes were analysed for all species in our experimentation set. We worked with the assumption that within a taxonomically homogenous group, many classes of whatever ecological theme would have identical suitability scores for many of its species. For instance, since the *Cercopithecinae* are the 'swamp monkeys', desert-like habitats all are unsuitable habitats. Such knowledge could be attributed once and for all to the group, i.e., the higher taxonomic level, and need not be repeated for all its individual species.

Moreover, we felt that the original 123 land cover classes in the actual data set had many pairwise look-alikes, allowing to work out a more abstract classification with fewer classes without loosing thematic resolution. Such an abstract classification also, and perhaps more importantly, allows to build up the species-related part of the knowledge base in a fashion more independent of actual data sets, in this case more independent of the land cover data actually used. Clearly, when such an abstraction is made we need to keep track of the translation scheme between abstract and actual theme legends.

In summary, we attempted to simplify a large two-dimensional data grid (species vs. land cover classes) by grouping both species and land cover classes. This approach worked out well for the land cover theme. We identified 18 abstract classes, from which 14 classes could be assigned suitability scores at the subfamily level. That is, just 14 suitability scores covering the majority of land cover classes for all species involved. For three of the remaining four classes, we had to identify two species groups as they displayed different suitability for these classes. The final abstract class was so ill-behaved that it needed treat-

ment at the species level, so for each species individually.

The reader should understand that we are attributing suitability scores for certain thematic classes at different levels in the species taxonomy. Obviously, to retrieve the complete overview of a species behaviour with respect to classes in a theme, all suitability scores (higher up the taxonomy) need to be collected. This is reminiscent of, but not identical to, attribute inheritance along a class hierarchy. Below, we provide a way of how to achieve this collecting of preferences with RACER.

The above story was a bit less ideal than we are here depicting it. Working on the base data, we found out certain data irregularities that we could not methodically explain or account for. In plain wording, we either missed some subtle step in the method or the base data had a certain percentage of error. We opted for the latter explanation. For instance, many species behaved totally proper, i.e., functional, meaning that the same class always showed the same score. A few species, however, displayed different suitability scores for the same land cover class, so they were non-functional on that land cover class. Quite often, in such cases, out of 20 or so combinations (with vegetation classes) using one and the same land cover class, two or so would give a score different from the other 18. We considered those two cases data errors. The overall figures on land cover suitability scores told us that approx. 92% behaved functional, meaning proper. The other 8% behaved non-functional. That percentage is only an upper bound on the error rate for land cover suitability because all 20 combinations above would count as non-functional. We believe that the actual error rate would be in the range of 0.8–1.5%.

We took a largely similar approach with vegetation data and the suitability scores for species. Luckily, the original vegetation data set came with two legends, one being an abstraction of the other. Data analysis showed that this abstract legend was a good candidate also to be taken up in our species ontology. However, it was a little bit too abstract as certain concrete vegetation classes still needed to be present to discriminate between cases. Due to lack of time, we worked out the vegetation data only for four closely related monkey species, the so called *mona monkeys* [45]. They form a superspecies taxon within the *Cercopithecinae*.

The identification of groups of species with similar ecological preferences for certain theme classes is accommodated in the species ontology by the special subclass Other. An individual of this class represents a group of species taxonomically positioned between the species and genus levels. The system is set up in such a way that arbitrary groups can be defined, violating the rule that a taxonomy takes a tree shape, but only at this level of the Other concept. This means that a single species may be represented by multiple individuals of the Other concept, and thus that they would need to 'inherit' their ecological preferences from all these individuals.

For the *mona* monkeys we associated preferences at the level of the *mona* superspecies as well as to two more species groups, contained within the superspecies, at the Other level. For each species individually, we still needed to define preferences at that level.

As to non-traceable data errors on vegetation scores, we saw a picture somewhat similar to the land cover case. These 'errors' were repaired by putting our methodical rules in place.

The above two ecological themes are discrete by nature, meaning that a finite number of different classes is identified for each of them. The species ontology discriminates between such discrete themes and continuous themes. A continuous theme is one in which the underlying geographical field behaves as a continuous function. The reason for this distinction is that their operational treatment in GIS context is fundamentally different. (This is a well-known fact.)

As indicated before, the AMD model also identifies certain species as being dependent on permanent water bodies. Where these exist, these species find more suitable habitats in the vicinity. The distance to water is the predominant factor in this ecological dependence. Clearly, distance is a continuous measure and thus dependence on water is better modelled as a continuous theme. This is just one example of a continuous theme having ecological relevance; other obvious examples are elevation and various meteorological parameters such as precipitation and various sorts of temperature measures.

We have chosen to represent ecological preferences for continuous themes through value ranges leading to a suitability score. In the case of water distance in the AMD model, this leads to the assignation of a single suitability score for the distance range 0 to max_distance, where max_distance depends on the species. For other continuous themes, multiple ranges may be required. Such themes were not present in this study, but could have been accommodated.

4.2 Species knowledge representation

As mentioned in Chapter 2, there exist two approaches in modelling species distribution. The two approaches differ in the sense that the species' ecological preferences in the first are known a priori and in the second, they need to be found from the characterization of locations where the species is known to occur. In this work, we focus on the first approach and in this section we define and provide a species ontology to store and model species ecological preferences. The information about the ecological preference for a species is used in the deductive approach to generate a distribution map (possibly) using a GIS.

We describe the ontology in terms of its concepts, individuals and roles. We specifically discuss two important concepts. The full ontology can be found in Appendix 6.

4.2.1 Taxon

The Taxon concept represents the set of biological taxa at different levels. It is a 'hybrid' representation of the 'pure' systematic/taxonomic hierarchy for species, together with the hierarchy we use to assign ecological preferences. The Taxon concept has thirteen specialized, disjoint subconcepts: Class, Subclass, Order, Suborder, Infraorder, Superfamily, Family, Subfamily, Tribe, Genus, Species,

Subspecies and Other. These concepts have been defined as general concept inclusions and together fully span the population of the class Taxon. The first twelve can be seen as representing individuals such as the species *wolfi*, the genus *Cercophitecus*, the subfamily Cercopithecinae, the family Cercopithecidae, the order Primates, the subclass Eutheria and the class Mammalia. The concept Other was already discussed.

Taxon individuals have relationships. For instance, we are interested to trace the taxonomic tree for any taxon at any level. We modelled this relationship using the role *hasAncestor*, and use it, for instance, to indicate the taxonomic link between a species and its genus. The role is defined as a *transitive* role, with its domain and range being the concept Taxon. The transitivity allows us (or better still: RACER) to infer that the species *wolfi* belongs to the order of the Primates.

The hasAncestor role not only allows to model 'pure' taxonomic relationships. It is also used also to create intermediate taxa (with little bio-systematic relevance) that were useful for attributing ecological preferences to species groups. We discussed this issue already above.

Each Taxon individual is associated with a certain rank. This allows to determine that *wolfi* is ranked at the species level, and not at the genus level. The different rank values that are associated to taxa are individuals of the enumeration list of the concept TaxonRank.

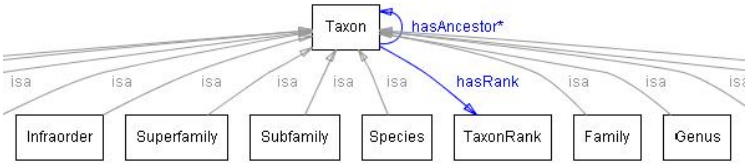

Figure 4.1: Subconcepts and roles of Taxon

We also defined the concrete domain attribute Taxon_name of type string, as we wanted to pose questions to the knowledge base like "provide the list of ecological preferences for taxon *wolfi*". To formulate such a query, as described in Section 3.2.7, we made use of *concrete domain concepts* (concrete predicate restrictions for attribute fillers [36]), which require internal, unique identifiers to discriminate amongst individuals.

Qualified and number restrictions have also been imposed for the concept Taxon. For instance, one of the range restrictions states that a taxon has a role with values restricted to taxon ranks in hasRank:

$$\mathsf{Taxon} \sqsubseteq \forall \mathsf{hasRank}.\mathsf{TaxonRank}\,.$$

An example of a cardinality restriction on the number of individuals assigned to an individual of the domain can be found in the role hasRank. It states that each Taxon individual must have been assigned exactly one taxon rank.

$$\mathsf{Taxon} \sqsubseteq \exists_{\geq 1}\mathsf{hasRank}.\mathsf{TaxonRank} \sqcap \exists_{\leq 1}\mathsf{hasRank}.\mathsf{TaxonRank}\,.$$

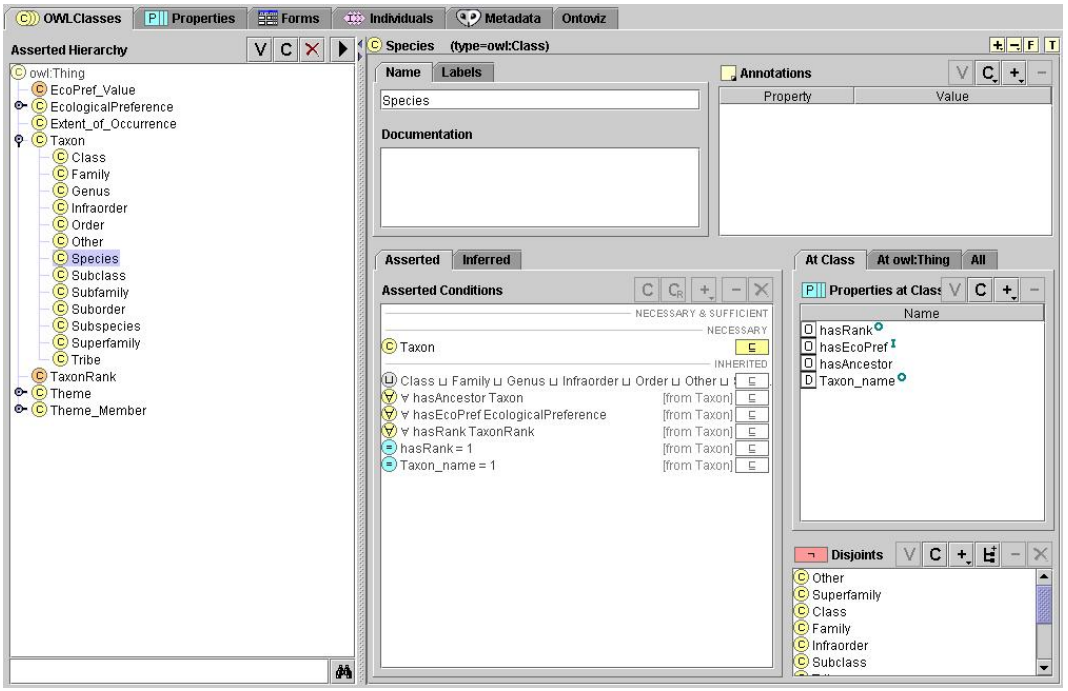

Figure 4.2: Graphical User Interface of Protégé showing the representation of the Taxon concept

4.2.2 Ecological preferences

The EcologicalPreference concept represents the population of ecological preferences associated to Taxon individuals. Ecological preferences are based on suitability scores assigned to themes separately (e.g., vegetation or land cover) and a fixed rule that determines the final score when suitability scores are combined. We essentially, for each theme involved, define a classification of the theme into sufficient members to be used in the model. An ecological preference indication like "Vegetation types such as forest are considered suitable for the species. Woodlands, and woodland mosaics and transitions are considered moderately suitable whereas grassland is considered unsuitable" leads to four individuals of the EcologicalPreference concept.

The Theme concept represents environmental variables in ecological preferences. As mentioned in section 4.1.2, the nature of the theme involved in an ecological preference has been accounted for in our modelling. The Theme concept has two disjoint specializations, Discrete_Theme and Continuous_Theme. Individuals of the first concept are vegetation and land cover whereas individuals of the second concept include elevation and distance_to_water. Following the same principle, we have specialized the EcologicalPreference concept into two disjoint subconcepts: EcologicalPreference_Discrete and EcologicalPreference_Continuous.

The EcologicalPreference_Discrete concept is described as a quadruple made of individuals of the Taxon, Theme_Discrete, Discrete_Theme_Member and

EcoPref_Value concepts. This allows us to express ecological preferences like "vegetation types such as forest are considered suitable for species *wolfi*."

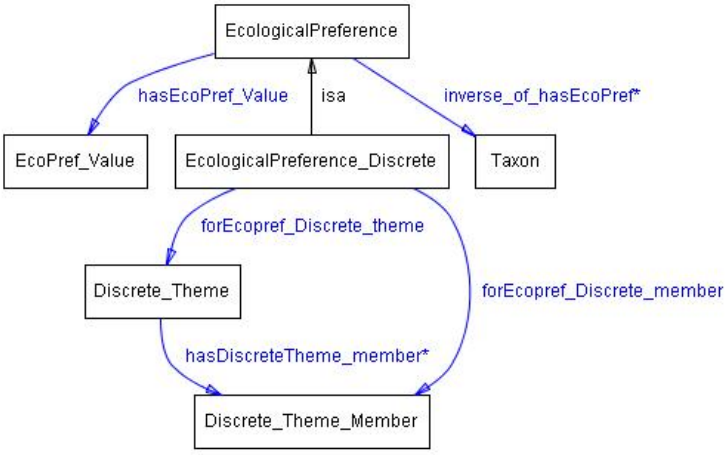

Figure 4.3: Main concepts and roles describing ecological preferences related to discrete themes

The Discrete_Theme_Member concept represents abstract classes of discrete themes (refer to Section 4.1.2). It is a specialization of the superconcept Theme and some of its individuals include forest and cropland. Properties assigned to the superclass are inherited by the Discrete_Theme_Member concept. This is the case for Theme_Member_Code, a concrete domain attribute of type integer that serves as an identifier for its individuals.

The different suitability scores that can be associated to an ecological preference are individuals of the enumeration list of the concept EcoPref_Value. Examples of these individuals are Suitable, Moderately Suitable and Unsuitable. We have assigned an identifier to these individuals using the concrete domain attribute Value.

Individuals of the EcologicalPreference_Discrete concept are related to individuals of the concepts Taxon, Discrete_Theme, Discrete_Theme_Member and EcoPref_Value. We have described the concepts and individuals involved in expressing such preferences but not how their individuals are related.

The relationship between a taxon and its associated ecological preferences is modelled using the role hasEcoPref. This role has as its range the concept EcologicalPreference, as either individuals of the concepts EcologicalPreference_Discrete and EcologicalPreference_Continuous may be related to a taxon. As each taxon may have arbitrarily many fillers for this role, no cardinality restrictions have been imposed. This is not true for the inverse role inverse_of_hasEcoPref, since individuals of the domain EcologicalPreference are related with at most with one individual of the range Taxon.

The relation of an ecological preference with a discrete theme is modelled using the role forEcopref_Discrete_theme. This role is subsumed by the role forTheme (with range EcologicalPreference and range Theme) as every instance of the first role must be an instance of the second role. We have ap-

plied the same principle of role hierarchy to relate individuals of the concept EcologicalPreference_Discrete_member with individuals of the concept Discrete_Theme_Member, using the subrole forEcopref_Discrete_member.

The role hasEcoPref_Value, which fillers map into the concept EcoPref_Value, completes our discussion on ecological preferences related to discrete themes.

We would like to impose the constraint that the discrete theme member indicated in an ecological preference is a valid member of the theme indicated in that preference. In $\mathcal{SHIQ}$, as far as we know this is not possible. It would be possible if we could use role chains as role descriptions, but $\mathcal{SHIQ}$ does not allow this. (Some other DLs do.) It is however fairly trivial to write a retrieval query in RQL that finds violations of this constraint.

As discussed in Section 4.1.2, we model ecological preferences for continuous themes through value ranges leading to a suitability score. The concepts involved in such preferences are individuals of Taxon, Continuous_Theme and Ecopref_Value. To express value ranges, we created two concrete domain attributes, namely forRange_maximum and forRange_minimum of type double. This allows to describe a preference like "species *pogonias* occurs within a distance of 1000 meters of permanent water."

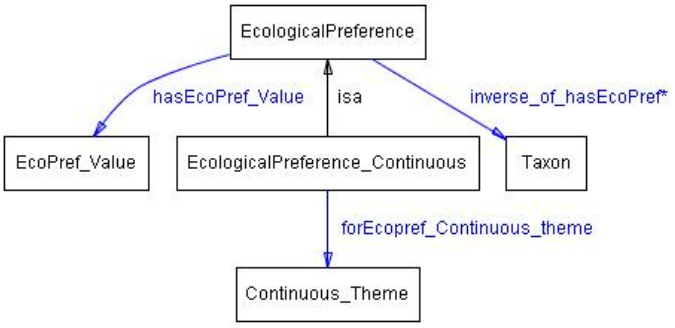

Figure 4.4: Main concepts and roles describing ecological preferences related to continuous themes

4.3 Spatial data set representation

In a deductive approach to SMP, once the environmental characteristics for the species have been identified, available spatio-ecological data sets have to be located for building the species distribution map. This section presents a modified and simplified version of the Geographic Information — Metadata (ISO 19115) ontology [42]. We briefly describe the concepts, individuals and roles that were used in our case study.

4.3.1 Metadata

This is the main concept in the ontology. It represents the population of individuals describing the characteristics of spatial data sets. It has several roles of im-

portance for SMPs, which mainly map into other concepts represented in the ontology. For instance, the role has_referenceSystem maps into the concept Reference_System to provide a description of the spatial reference system used in the data set. There exist two types of reference systems. Therefore, this concept is specialized in two disjoint subconcepts, Geographic_Coordinate_System and Projected_Coordinate_System, spanning over the superconcept. Both subconcepts require information regarding the ellipsoid for their definition. This information is described by individuals of Ellipsoid and Ellipsoid_Parameters. An individual of Ellipsoid_Parameters is described by characteristics such as the radius of the equatorial axis of the ellipsoid.

The subconcept Projected_Coordinate_System, besides the ellipsoid information we just described, has two more properties. The role Projection_Parameters has as fillers individuals of ProjectionParameters. This concept describes the parameters used by the projection. It has several roles that allow to set characteristics such as the scale factor at the projection origin, false northing and false easting, standard parallel and latitude and longitude of the projection centre.

The information of the spatial representation of the data set is described in the concept SpatialRepresentation. This concept has two disjoint specializations, VectorRepresentation and GridRepresentation. For instance, individuals in the first concept include TINs and data sets with attributes, and they describe properties such as the type of geometry used to represent spatial features. Examples of individuals of the second concept include images and raster files.

The geographic area covered by the data set is described in the concept Geographic_Extent. This concept has four properties that allow to set the bounding coordinates of its spatial extent. This allows to locate data sets when interested in a specific area.

To locate spatial data sets relevant for the taxon under study, we need to look for information related to environmental variables (themes). For this purpose, we created the concept Legend, with specializations Legend_Discrete and Legend_Continuous.

The concept Legend has the role has_Legend_Theme mapping into concept Legend_Theme. This concept, as described in the previous section, represents environmental variables that are (possibly) used to describe ecological preferences. It has two disjoint specializations, Legend_Discrete_Theme and Legend_Continuous_Theme. Examples of individuals of the first concept are vegetation and land cover. Elevation is one example of an individual of the second subconcept.

The subconcept Legend_Discrete has two roles, has_Legend_Discrete_Values and has_Classification_system. The first role has as fillers the values that are used in the discrete theme described by the data set. Individuals of this concept include transitional rain forest and dry evergreen forest - Malagasy for the discrete theme vegetation. The role has_Classification_system maps into individuals of the concept Classification_system. This concept represents the set of classification systems that has been used for the values in the data set. Individuals of this concept include White's Vegetation Map and Seasonal Land Cover Map, two of the actual classification systems that we use in this work.

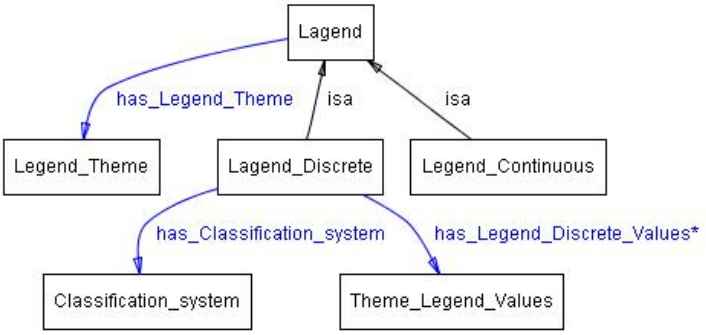

Figure 4.5: Main concepts and roles describing legends related to discrete themes

We would like to impose the constraint that the theme legend value member indicated in a legend is a valid member of the classification system indicated in that legend. Again, this has not been done as we can not use role chains as role descriptions (refer to our early discussion in Section 4.2.2).

4.4 Reasoning over the species ontology

In this section, we explain a number of queries that we run against the species ontology. Essentially, they are used to retrieve ecological preference information for taxon individuals. We posed these queries to the RACER system using the extended query language RQL (Racer Query Language) [37].

Determining the ecological preferences of a taxon This query requires retrieving the ecological preferences directly assigned to the specified taxon (*mona* used here as an example) and the ones assigned to its ancestors. Below we provide such a query together with a description of the objects involved.

```
(retrieve (?e ?t ?m ?s) (or
                         (and
                          (|mona| ?e |hasEcoPref|)
                          (?e ?m |forTheme_member|)
                          (?e ?s |hasEcoPref_Value|)
                          (?m ?t |inverse_of_hasTheme_member|))
                         (and
                          (|mona| ?x |hasAncestor|)
                          (?x ?e |hasEcoPref|)
                          (?e ?m |forTheme_member|)
                          (?e ?s |hasEcoPref_Value|)
                          (?m ?t |inverse_of_hasTheme_member|)))
```

This query determines the union of two sets (A and B). For set A, we retrieve the ecological preferences, themes, theme members and suitability scores assigned to taxon *mona*. We can look at it as the query:

```
(retrieve (?e ?t ?m ?s) (and
                         (|mona| ?e |hasEcoPref|)
                         (?e ?m |forTheme_member|)
                         (?e ?s |hasEcoPref_Value|)
                         (?m ?t |inverse_of_hasTheme_member|)))
```

Within the call of `retrieve`, `(?x ?t ?m s?)` represents the list of result objects (each list in the result will list four variable-value-pairs). The rest of the expression is what RQL calls the *query body*. In this case, it is a compound expression made of four conjuncts. In the first expression, the variable `?e` is bound to the ecological preferences assigned to the individual *mona*. It asks the system for fillers of role `hasEcoPref` on individual `mona`. The second expression uses the values of variable `?e` and binds variable `?m` to each of the Theme_Member individuals in each ecological preference. The third expression, binds variable `?s` to the suitability score (individuals of EcoPref_Value) assigned to each ecological preference. The last expression, binds variable `?t` to individuals of Theme which these ecological preferences refer to.

After running this query, RACER returns the following list:

```
(((?E |EcoPrefG1|) (?T |Vegetation|)
   (?M |woodland|)(?S |Moderately_suitable|))
 ((?E |EcoPrefG2|) (?T |Vegetation|)
   (?M |woodland_mosaics_and_transitions|) (?S |Moderately_suitable|))
 ((?E |EcoprefS1|) (?T |LandCover|)
   (?M |mangroves/swamps-tropical_forest|) (?S |Suitable|)))
```

The second part of the union query, set B, retrieves the same information but for the ancestors of taxon *mona*.

```
(retrieve (?e ?t ?m ?s) (and
                         (|mona| ?x |hasAncestor|)
                         (?x ?e |hasEcoPref|)
                         (?e ?m |forTheme_member|)
                         (?e ?s |hasEcoPref_Value|)
                         (?m ?t |inverse_of_hasTheme_member|)))
```

This query is very similar to the one described above. It only contains one more query expression which binds variable `x?` to all the ancestors of taxon *mona*. The system retrieves the ancestors for the taxon because the role hasAncestor has been declared as a transitive role in the TBox. The values of this variable are then used to retrieve the ecological preferences assigned to each of the ancestors of taxon *mona*, as well as the themes, theme members and suitability scores related to them.

The or operator in the first query is simply used to generate the union of the results obtained in set A and set B separately. The final result is a list of quadruples listing all the ecological preferences, themes, theme members and suitability scores for taxon *mona*.

Determine the ecological preferences of a taxon related to a theme When we are interested in querying a single theme, we have to slightly modify the above query. We remove the variable that we previously associated to individuals of the concept Theme, and we replace the variable in the query body by the constant that represents the theme that we are interested in. The following query retrieves the complete list of ecological preferences, theme members and suitability scores associated to taxon *mona* and related to the theme vegetation.

```
(retrieve (?e ?m ?s)
      (or
       (and
        (|mona| ?x |hasAncestor|)
        (?x ?e |hasEcoPref|)
        (?e ?m |forTheme_member|)
        (?e ?s |hasEcoPref_Value|)
        (?m |Vegetation| |inverse_of_hasTheme_member|))
       (and
        (|mona| ?x |hasAncestor|)
        (?x ?e |hasEcoPref|)
        (?e ?m |forTheme_member|)
        (?e ?s |hasEcoPref_Value|)
        (?m |Vegetation| |inverse_of_hasTheme_member|))))
```

4.5 Summary

In this chapter, we have looked at Description Logics applied to species mapping procedures. We started by describing the base data for this project.

We modelled and described two ontologies, the species ontology and the spatial data set ontology. Then, using RACER we run some queries against the knowledge base. We have described two of these queries.

In the next chapter, we look at an example of a mapping exercise. We describe its main computational steps and explain how configuration techniques may help in automating or semi-automating this process.

Chapter 5

Scripting as a configuration problem

We are interested in a (semi-)automatic system for species mapping procedures. A mapping procedure, as described in Chapter 4, can be thought of as a (possibly abstract) computing script that combines parts of the D, M and O components fitting together for generating a final distribution map. In the previous chapter, we described the D component, explaining two of the ontologies behind it (the species and spatial data set ontology). In this chapter, we describe the remaining components, but especially the M component. We can view the M component as a complete script for generating a distribution map. This script is a nested script, with computational steps at different levels. In the first part of the chapter, we describe three high level computational steps that can be seen as parts plugged into the main script: *species data selection*, *spatial data set selection* and *mapping potential species distribution*.

The second part of the chapter looks at the species mapping procedure from a *configuration problem* perspective. We follow the technique proposed in [29] and we apply Description Logics in the SMP domain.

It is worth mentioning that the second part of this chapter is not settled as much as previous chapters. Time constraints have not allowed us to look at the configuration problem domain in the detail we had wanted. We provide a description of our understanding on configuration problems and on how this could be applied to species mapping procedures.

5.1 High level components in SMPs

The first model we developed consists of three high level components, i.e., rather abstract computational steps. Each of these itself consists of smaller components, which need to be filled out. A component can be seen as a part plugged into the script. This is true for components at all levels.

5.1.1 Species data selection

The overall goal of this step is to determine the species of interest to the user, and optionally the area of interest to the user. Also, once these have been identified, any knowledge regarding the species' ecological preferences will be traced in the knowledge base. It consists of several smaller components:

Processing of user request This first function of the system reads the user's request consisting of the species of interest plus (optionally) the area of interest for the mapping exercise. We can think of the input for this component as the species scientific name and the specification of the area by means of the bounding coordinates of the area of extent, including the spatial reference system parameters used to specify those coordinates. This component, after processing the request, delivers as output the same information in a suitable format to be read by the next component.

Tracing knowledge on the species' ecological preference This component reads the output parameter regarding the species of interest generated by the previous component and traces in the knowledge base any knowledge related the species' ecological preference. In the deductive approach, as depicted in this work, the species' ecological preference information is available within the system. The operation within this component may well be a constructed query against the species ontology. The query should allow for treatment of specific ecological preferences like water dependence. For the case of a non-water-dependent species, the outcome of the query would consist of a list of triplets $\langle theme, thememember, suitabilityscore \rangle$.

Extracting relevant themes for the SMP This component reads the information regarding the species ecological preference, and splits it according to the main environmental variables (themes). Again, the splitting operation should take into account the type of ecological preference (water or non-water related) when extracting the themes represented in the ecological preference. In our example, we look at non-water related species and at two themes, vegetation and land cover.

Generating tables for themes The model here depicted, requires separate tables with attributes providing the relationship between theme members and suitability scores assigned to them. This component, takes as input triplets of the form $\langle theme, thememember, suitabilityscore \rangle$ and generates two separate tables, each related to a single theme, with theme members and suitability scores as attribute values.

5.1.2 Spatial data set selection

Since our model is a deductive one, this step identifies whether an Extent of Occurrence (EO) for the species is available, and whether spatio-ecological data sets for the themes identified above are available. Also, any necessary data conversions are determined as preparatory steps. This high level component consists of the following smaller components:

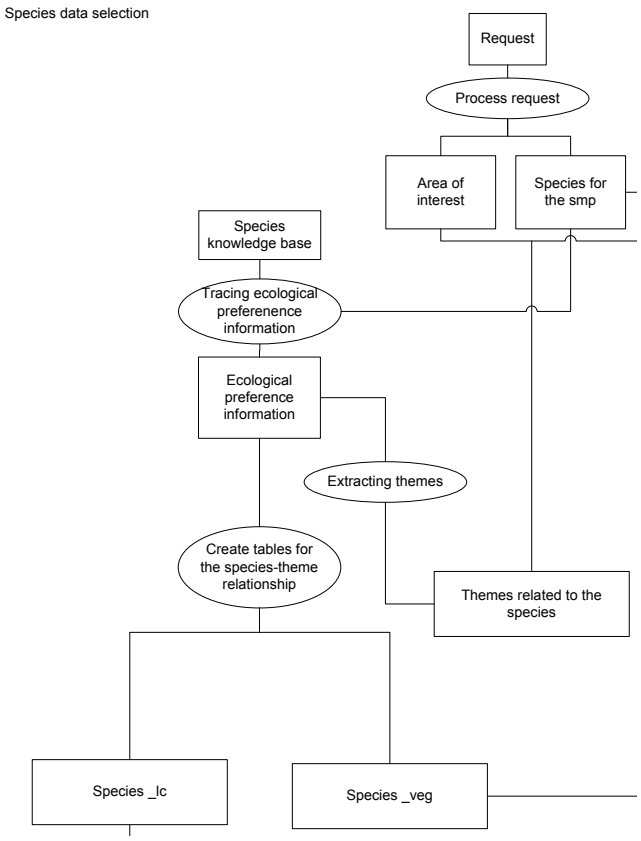

Figure 5.1: Main components of the 'Species data selection' component

Locating the species' EO file This component has the task to locate the extent of occurrence file using the species name as input. It makes use of the data set ontology and creates an instantiation of the spatial data set.

Import Aware of the EO file format, this component is responsible to bring the data into the system in an appropriate format for use in the SMP. The function considers the spatio-ecological (theme) data set format that will also be used. This component, using the spatial data set ontology, creates an instantiation of the spatial data set.

Locating spatio-ecological data sets Once the environmental themes important for the species have been identified, this component traces the available spatio-ecological data sets, taking into account the user's requested area of interest. This is a two step process: (1) identification of data sets reflecting the main theme of preference (e.g., vegetation), and (2) analysis of theme values within the ecological preference to identify if there is a match with the ones in the data sets. This last step may require different functions when these data sets do not come with metadata that defines the theme members in a standardized way.

Choice amongst data sets When multiple data sets are available for a single theme, this component is in charge of making a choice between them

(based on metadata values). Depending on the theme, the criteria used may reflect different priorities. For instance, temporal resolution may be more important than spatial accuracy for the vegetation theme than it is for the elevation theme. This component delivers the spatial data sets that are useful for the mapping exercise.

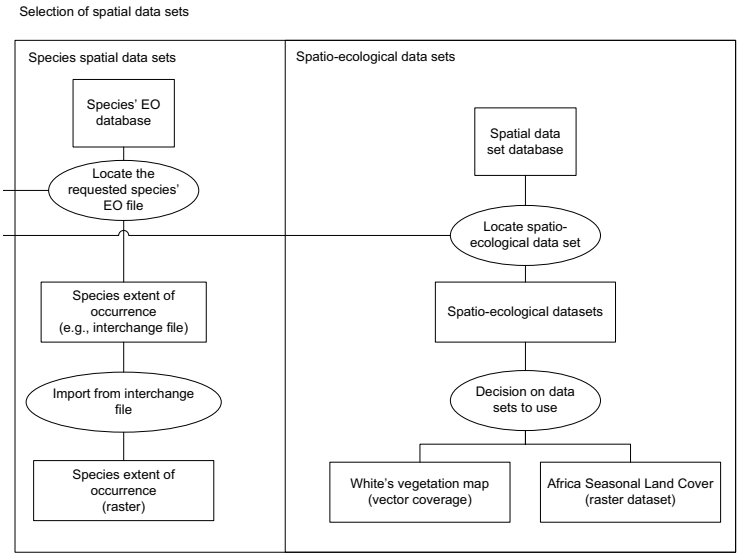

Figure 5.2: Main components of the 'Spatial data set selection' component

5.2 Mapping potential species distribution

In this third high level step of the model, the data sets found earlier are combined in a spatially meaningful way, using a GIS, to obtain the potential distribution map. We describe the main second-level component in the paragraph below.

Overlay The final step in the species mapping procedure is to combine the spatial data sets related to species' ecological preferences with a fixed rule that determines the final score when suitability scores are combined. This component, therefore, requires spatial data sets that match the elements described by the logical rule and other metadata characteristics that make them suitable for being processed by the component. In our example, this rule is defined for suitability scores related to vegetation and land cover. The overlay function is such that it requires the suitability scores as values in a specific attribute. In this case, other functions have to be applied to construct an input data set that meets these requirements. Moreover, to use the rule as such, the component requires the spatial data sets to be in 'raster' format, and with several metadata values that should be shared by all the data sets involved (e.g., the same spatial reference system and the same cell size). This means that, when these conditions are not met,

the component may request other functions (such as format conversions and resampling functions) to first prepare the data for further use.

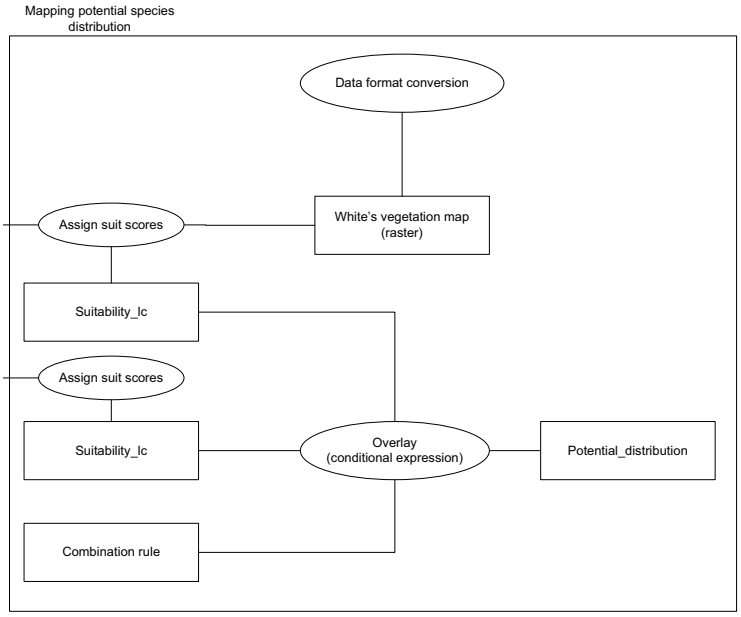

Figure 5.3: Main components of the 'Mapping potential species distribution' component

After completion of this whole procedure, we obtained the map depicted in Figure 5.4.

5.3 Constructing SMPs as a configuration problem

In this work, we attempt to contribute to methodical consistency, especially in that of repeatable, instantaneous computer-aided distribution mapping, in scenarios where new data sets become frequently available. This requires a proper combination of spatial data sets, knowledge of species' ecological preferences and of mapping methods to generate distribution maps upon request.

In the previous section, we looked at the components combining such information. The choice between components, was guided by metadata characteristics of the different types of input and output. For instance, when we had to combine spatial data sets and these were in a raster format, we used a component that had 'built-in knowledge' on how to perform this operation. We can say that each SMP needs to find the proper components that work together to generate a final map.

Configuration techniques, as described in in [29, 51] aim at finding a suitable composition of parts to construct a whole. For instance, they have been used in the the AI domain in configuring computer systems. Moreover, configuration problems, seem to be well-suited to description-logic solutions [51]. We therefore believe that these techniques can be applied to our SMP domain problem.

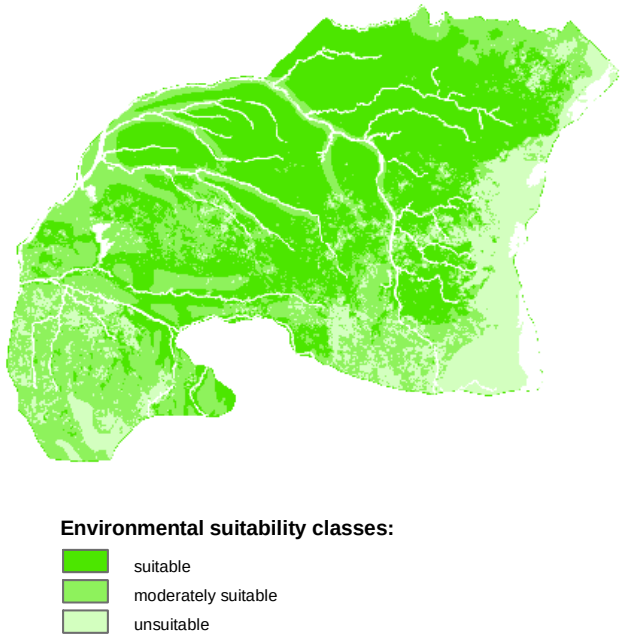

Figure 5.4: The potential distribution of Wolf's Monkey (*Cercopithecus (mona) wolfi*). Area depicted is part of the Southern Zaïre Basin, Democratic Republic of Congo, south and east of Zaïre river, including the Kasai river watershed.

We based our work on the techniques proposed in [29]. We first provide a general description of how these authors view a configuration problem, followed by the application of this particular technique to a small part of the SMP depicted in the previous section.

5.3.1 Configuration problems

Configuration problem techniques attempt to construct a consistent whole out of a range of possible *parts*. These may well be parts of a computer system or parts of a high quality stereo system. The main idea behind configuration problem-solving is to describe the possible configurations of these parts in a certain domain.

The work in [29] provides an example configuration problem applied to computer systems. A computer system comprises several parts (such as CPUs and motherboards), which the final product may consist of. The idea behind their configuration technique is to define, beforehand, which are the possible configuration products built from these parts (e.g., which are possible, allowed or restricted combinations of these parts). Then, when a customer may have specific requirements for a variant of the final product, the system can determine if the request can be satisfied.

A *configuration problem* is therefore described as a set of parts, a set of constraints, and a set of user requirements that may (or may not) satisfy a valid configuration solution.

The authors describe a configuration problem as the triple D, S, and C. Part

D is the domain description of valid configurable products. Part S describes the system requirements specification, while C denotes the set of concept and role names that the system can use as a language for describing a solution to a configurable product. This set is a subset of the names used in the ontology/-ies of D, chosen in such a way that the solution can only identify true (physical) parts, not abstractions of these.

A valid *configuration solution* (or actual configuration), on the other hand, is described by (*CONF*, *COMPS*, and *ROLES*). *CONF* is the description of a configuration solution, made of the set of actual components (*COMPS*) and the relation between these components (*ROLES*) in an actual configuration solution. A valid configuration solution is one that does not violate the domain description D.

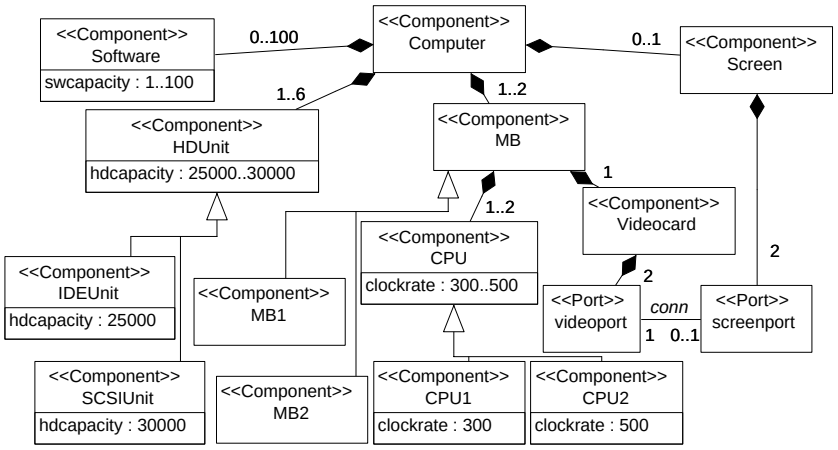

Figure 5.5: A computer system configuration ontology, after (29)

Let us look in more detail at the description of a configuration problem. Figure 5.5 depicts the product structure of a computer system, showing the parts (component types), subparts of parts (indicated by part-whole relationships), and physical connections between parts (so called port connections). What is not depicted in the figure are further constraints on component connections and those on allowed values of certain attributes/features of components. Using Description Logics, a configuration problem is described as follows:

Component types are the parts of a configuration problem. They can be organized in a hierarchy — with abstract component classes higher up, and concrete component classes at the leafs — and they may or (may not) have attributes. Hierarchies allow to describe the 'architecture' of the system in general terms, before adding detail in characteristics and constraints.

Part-whole relationships allow to describe parts and subparts using two roles hasPart and partOf. The description allows to define constraints on these roles stating, for instance, that a subpart can only belong to one part. Often, additional roles are identified to specialize the two part-whole relationships, and these are then defined as subrole of either hasPart or

partOf, retaining the original semantics of these roles, but allowing special domain and/or range for them.

Port connections describe how components are connected with each other. For instance, they may model the fact that a Videocard component must be connected via its videoport with a Screen component via its screenport. For this purpose, a new concept is introduced, namely that of Port, which is characterized by three roles. The role compnt indicates the component of which it is part, the role portname identifies the name of the port in the connection, and the role conn determines the Port individual of the other component that it connects to. Connections have to be unique and, therefore, the roles are defined using the inverse role constructor. The constraints over these connections are modelled by role compositions (navigation paths), a sequence of existential role quantifications, for the two concepts involved in the connection.

Existence dependency Sometimes the existence of one component (type) may be required for another component (type), (e.g., CPU2 requires MB2). This takes the shape of an implication $A \Rightarrow B$ and in Description Logics, it is defined by using this logical rule applied to the roles in which fillers are the required components.

Incompatibility In cases where there exists potential incompatibility between components (e.g., CPU1 is incompatible with motherboard MB2), we can express this constraint in Description Logics using the related roles for which fillers are the incompatible components in the way $not(A \wedge B)$.

5.3.2 Our configuration problem

Following from what we described in the previous section, we make an attempt in defining part of the SMP from a configuration problem perspective. The part here depicted, relates to the procedure in which spatial data sets are combined in an overlay to produce the final distribution map.

In our case, we are interested in computational steps, rather than in physical components such as CPUs and MBs. We believe that the general principles we described above, can also be applied in our domain. In this example, the computational step is a script that performs such an overlay operation. The Overlay component can be thought of as an abstract component, specialized as one of two subcomponents. Types of overlay operators include Raster-Raster overlay and Vector-Vector overlay. These contain the actual code to perform the overlay. The difference between these two components is the type of data they require as input. This can be modelled as a property of the component, of which the values are concepts of the spatial data set ontology.

Other components in our set-up include receivers. A receiver component can be thought of a 'box' in which another computational step drops the data it generates. Therefore, we also have different types of receivers, depending on the type of data they can hold. These are organized in a hierarchy, with subcomponents such as Raster receiver and Vector receiver. The important

difference amongst them is in the value for their data type property. These values point, once more, to the concepts of the spatial data set ontology. For instance, one computational step may generate a raster data set. This means that we have to ensure that the receiver where the data is dropped is of type Raster.

In our work, part-whole relationships exist between the different small scripts and the one covering the whole spatial mapping procedure, for instance.

Port connections is what we call InputOutput channels. They define how different components are connected to each other. We can think of them as consisting of two roles and a property, which type indicates the (spatial) data set that can pass the channel.

Constraints of dependency between components can be found in our example between, for instance, the type of overlay operator and the type of receiver that produces its input. This can be modelled as a constraint, as illustrated in the previous section.

5.4 Summary

In the first part of this chapter, we have described one example of a mapping exercise. Essentially, it is consisting of three main computational blocks: selection of the species data, selection of geo-ecological data sets and the operations, in a GIS context, to generate a species distribution map. We describe these first-level components as well as smaller components at different levels.

We have also looked at configuration problems and how these techniques may be applied in the domain of SMPs.

Chapter 6

Conclusions and our future work

This thesis work addresses a research effort made in the context of GIS-supported mapping of the distribution of biological life forms. Such mappings are commonly obtained to study and improve the understanding of living creatures on our planet. Many techniques have been proposed in the past, and are in day-to-day use. We have not attempted to add to the already large collection of such techniques.

Instead the work aimed at improving the automatization of these mapping procedures, so as to make them more methodically consistent, and to let them allow repeated, instantaneous computer-aided species distribution mapping, in scenarios where new data sets become available regularly.

Species mapping is often conducted in one-off projects, in which the data handling steps are usually never revisited. In such situations, steps inappropriate for the data at hand are easily made, due the infrequency of actions taken and subsequent inexperience of the experts involved. Automated support in such cases is wanting. When species mapping has become a more regular activity, as in large mapping initiatives of many biological taxa, or as in repeated monitoring of fewer taxa, automated support also has obvious advantages.

Thus, we did not attempt to answer ecological problems, but rather wanted to work on the provision of flexible methods supporting ecologists in their mapping procedures, in the hope of deriving a procedural understanding that could eventually be (better) automated.

Specifically, we addressed the issue of automatically constructing a reliable method for determining (anew) a species distribution map, using a GIS, from spatial foundation data, species knowledge, mapping method knowledge and map purpose. We worked under the assumption that any of the latter four inputs might change overnight, possibly resulting in redetermination of the output, the species map.

The question that this chapter should answer then is whether we succeeded in meeting the goals of the work. In summary, we feel it is too early to tell. (In other words: no, we did not.) But we also believe that there is substantial promise in the approach we took; we explore the reasons for this belief below.

First of all, under the very assumption of this work that data sets relevant to distribution mapping will in the future become available at increasing frequency, there is certainly a need for the implementation of methods that can accommodate such new data quickly. We must expect that the primary source for such data is the internet, and that the data's 'author' is not us but someone 'strange' to our own projects. The data thus obtained will need to be self-descriptive: indicate what exactly it contains, and how it should be interpreted. Our techniques should be aware of such systems of self-description, and be capable of interpreting them. This is one reason why the use of DLs here is promising: they are a cutting-edge technique of self-description.

The reader will have unsuccessfully searched for a definition that makes a solid distinction between such terms as data, information, metadata, and (even) knowledge. One may argue that information and knowledge only live in (or at least only survive in) the human brain. This would mean that these terms have little use in a purely technical work like this one. (It also shines a light on terms such as 'information system' and 'knowledge representation' but we leave it at that.) The distinction between data and metadata seems at best a relative one: what is metadata in one situation, may be data in the next. We thus find it fitting that we have worked with a formalism in which the two can almost seamlessly be combined. Again, the application of DLs to the domain seems a useful choice, with potential for the future.

A formalism that allows to combine data and metadata is certainly needed in cases where one wants to automate a reasoning mechanism that deals with available data and choices of data manipulations on that data. One in which flexibility must be provided to deal with changes in user requirements, data availability, support for different procedures. All these require reasoning at some level, thus they requires knowledge representation in a format that can be operated on by a machine. Description Logics are so applicable because they define concepts and their relations, and they have built-in support for typical inferences such as subclassing an subsumption.

Finally, we started to work on the use of DL-based definitions of a Universe of Discourse for the purpose of constructing well-understood artifacts. There is a fairly extensive build-up of proof that configuration problems can be nicely addressed with DL-based formalisms. We were optimistic in using the techniques that have been developed for the composition of technical (hardware) systems using configuration algorithmics, and we still are. We have been able to describe our configuration problem, the construction of a (software) script to generate distribution maps, at least in part, as a satisfiability problem against our accumulated otologic knowledge.

We must confess also that the actual generation of such scripts appears to be more complicated than we anticipated, at least given our current understanding. SO far, we have not identified in the literature proposals to approaches of generating these solutions. They are known to exist, but so far elude us, and seem only present in proprietary software packages with high price tags.

Much time during the execution of this project was invested to understand the various domains: ecological mapping, description logics, configuration problems. Not all that time was equally well-spent. Quite a bit of time was also

spent in understanding the data that we used for the project, as well as their open and hidden regularities and irregularities. These all provided us with useful lessons, although the going was sometimes slow.

We will continue to work on this problem domain, because it is an exciting field, with lots of new techniques that are just begging to be explored and applied. Ecological mapping is a really nice application domain for it, with obvious relevance. But all the techniques have a much wider application as well, definitely also in the geo-information field.

For the short term, we have in mind exploring the issue of configuration solutions further. These are needed for an eventual complete first version of our system. It will put the designed ontologies to a wider and much needed test. Equally important in this domain is the further unfolding of an ontology of GIS operations. Many people are active in this field, but the crucial characteristics that help to represent them best have yet to be discovered. In the end, we hope to redo part of the AMD project, just to see whether our approach would have resulted in equally good final products, or perhaps even better ones ...

Species ontology

The code below provides the full ontology for the species part of the knowledge base. It uses the native racer format, which is actually the most readable format.

```
(IN-TBOX |SpeciesOntoDLFinal_DL.owl| :SIZE 372 :ROLE-SIZE 25)

(IMPLIES |Taxon| TOP)
(IMPLIES |Taxon| (ALL |hasAncestor| |Taxon|))
(IMPLIES |Taxon| (ALL |hasEcoPref| |EcologicalPreference|))
(IMPLIES |Taxon| (ALL |hasRank| |TaxonRank|))
(IMPLIES |Taxon| (AND (AT-LEAST 1 |hasRank|) (AT-MOST 1 |hasRank|)))
(IMPLIES |Taxon| (AND (AT-LEAST 1 |Taxon_name|) (AT-MOST 1 |Taxon_name|)))
(IMPLIES |Taxon| (OR |Class| |Family| |Genus| |Infraorder| |Order| |Other| |Species| |Subclass| |Subfamily|
     |Suborder| |Subspecies| |Superfamily| |Tribe|))
(IMPLIES |Class| |Taxon|)
(IMPLIES |Family| |Taxon|)
(IMPLIES |Genus| |Taxon|)
(IMPLIES |Infraorder| |Taxon|)
(IMPLIES |Order| |Taxon|)
(IMPLIES |Other| |Taxon|)
(IMPLIES |Species| |Taxon|)
(IMPLIES |Subclass| |Taxon|))
(IMPLIES |Subfamily| |Taxon|)
(IMPLIES |Suborder| |Taxon|)
(IMPLIES |Subspecies| |Taxon|)
(IMPLIES |Superfamily| |Taxon|)
(IMPLIES |Tribe| |Taxon|)
(DISJOINT |Class| |Genus| |Family| |Infraorder| |Order| |Other| |Species| |Subclass| |Subfamily|
     |Suborder| |Subspecies| |Superfamily| |Tribe|)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |Taxon_name| :DOMAIN |Taxon| :TYPE STRING)
(DEFINE-PRIMITIVE-ATTRIBUTE |hasRank| :DOMAIN |Taxon| :RANGE |TaxonRank|)
(DEFINE-PRIMITIVE-ROLE |hasAncestor| :TRANSITIVE T :DOMAIN |Taxon| :RANGE |Taxon|)
(DEFINE-PRIMITIVE-ROLE |hasEcoPref| :INVERSE |inverse_of_hasEcoPref| :DOMAIN |Taxon|
     :RANGE |EcologicalPreference|)

(DEFINE-CONCEPT |TaxonRank| (OR |class| |family| |genus| |infraorder| |order| |other| |species|
     |subclass| |subfamily| |suborder| |subspecies| |superfamily| |tribe|))
(IMPLIES |TaxonRank| TOP)

(IMPLIES |Theme| TOP)
(IMPLIES |Theme| (AND (AT-LEAST 1 |Theme_name|) (AT-MOST 1 |Theme_name|)))
(IMPLIES |Theme| (OR |Discrete_Theme| |Continuous_Theme|))
(IMPLIES |Theme| (ALL |hasTheme_member| |Theme_Member|))
(IMPLIES |Continuous_Theme| |Theme|)
(IMPLIES |Continuous_Theme| (ALL |hasContinuousTheme_member| |Continuous_Theme_Member|))
(IMPLIES |Discrete_Theme| |Theme|)
(IMPLIES |Discrete_Theme| (ALL |hasDiscreteTheme_member| |Discrete_Theme_Member|))
(DISJOINT |Discrete_Theme| |Continuous_Theme|)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |Theme_name| :DOMAIN |Theme| :TYPE STRING)
(DEFINE-PRIMITIVE-ROLE |hasTheme_member| :INVERSE |inverse_of_hasTheme_member| :DOMAIN |Theme|
     :RANGE |Theme_Member|)
(DEFINE-PRIMITIVE-ROLE |hasContinuousTheme_member| :PARENTS |hasTheme_member|
     :INVERSE |inverse_of_hasContinuousTheme_member| :DOMAIN |Continuous_Theme|
     :RANGE |Continuous_Theme_Member|) (DEFINE-PRIMITIVE-ROLE
|hasDiscreteTheme_member| :PARENTS |hasTheme_member|
     :INVERSE |inverse_of_hasDiscreteTheme_member| :DOMAIN |Discrete_Theme|
     :RANGE |Discrete_Theme_Member|)

(IMPLIES |Theme_Member| TOP)
(IMPLIES |Theme_Member| (ALL |inverse_of_hasTheme_member| |Theme|))
(IMPLIES |Theme_Member| (OR |Continuous_Theme_Member| |Discrete_Theme_Member|))
(IMPLIES |Continuous_Theme_Member| |Theme_Member|)
(IMPLIES |Continuous_Theme_Member| (AND (NOT |Discrete_Theme_Member|)))
(IMPLIES |Discrete_Theme_Member| |Theme_Member|)
(IMPLIES |Discrete_Theme_Member| (AND (AT-LEAST 1 |Theme_Member_Code|) (AT-MOST 1 |Theme_Member_Code|)))
(DISJOINT |Discrete_Theme_Member| |Continuous_Theme_Member|)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |Theme_Member_Code| :DOMAIN |Theme_Member| :TYPE INTEGER)
(DEFINE-PRIMITIVE-ATTRIBUTE |inverse_of_hasTheme_member| :INVERSE |hasTheme_member|
     :DOMAIN |Theme_Member| :RANGE |Theme|)
```

```
(DEFINE-PRIMITIVE-ATTRIBUTE |inverse_of_hasContinuousTheme_member| :PARENTS |inverse_of_hasTheme_member|
     :INVERSE |hasContinuousTheme_member| :DOMAIN |Continuous_Theme_Member| :RANGE |Continuous_Theme|)
(DEFINE-PRIMITIVE-ROLE |inverse_of_hasDiscreteTheme_member| :PARENTS |inverse_of_hasTheme_member|
     :INVERSE |hasDiscreteTheme_member| :DOMAIN |Discrete_Theme_Member| :RANGE |Discrete_Theme|)

(IMPLIES |EcologicalPreference| TOP)
(IMPLIES |EcologicalPreference| (ALL |hasEcoPref_Value| |EcoPref_Value|))
(IMPLIES |EcologicalPreference| (ALL |inverse_of_hasEcoPref| |Taxon|))
(IMPLIES |EcologicalPreference| (AND (AT-LEAST 1 |forTheme_member|) (AT-MOST 1 |forTheme_member|)))
(IMPLIES |EcologicalPreference| (AND (AT-LEAST 1 |forTheme|) (AT-MOST 1 |forTheme|)))
(IMPLIES |EcologicalPreference| (AND (AT-LEAST 1 |hasEcoPref_Value|) (AT-MOST 1 |hasEcoPref_Value|)))
(IMPLIES |EcologicalPreference| (OR |EcologicalPreference_Continuous| |EcologicalPreference_Discrete|))
(IMPLIES |EcologicalPreference_Continuous| |EcologicalPreference|)
(IMPLIES |EcologicalPreference_Continuous| (ALL |forEcopref_Continuous_member| |Continuous_Theme_Member|))
(IMPLIES |EcologicalPreference_Continuous| (ALL |forEcopref_Continuous_theme| |Continuous_Theme|))
(IMPLIES |EcologicalPreference_Continuous| (AND (AT-LEAST 1 |forEcopref_Continuous_theme|)
     (AT-MOST 1 |forEcopref_Continuous_theme|)))
(IMPLIES |EcologicalPreference_Discrete| |EcologicalPreference|)
(IMPLIES |EcologicalPreference_Discrete| (AND (AT-LEAST 1 |forEcopref_Discrete_member|)
     (AT-MOST 1 |forEcopref_Discrete_member|)))
(IMPLIES |EcologicalPreference_Discrete| (ALL |forEcopref_Discrete_member| |Discrete_Theme_Member|))
(IMPLIES |EcologicalPreference_Discrete| (AND (AT-LEAST 1 |forEcopref_Discrete_theme|)
     (AT-MOST 1 |forEcopref_Discrete_theme|)))
(IMPLIES |EcologicalPreference_Discrete| (ALL |forEcopref_Discrete_theme| |Discrete_Theme|))
(DISJOINT |EcologicalPreference_Continuous| |EcologicalPreference_Discrete|)
(DEFINE-PRIMITIVE-ROLE |inverse_of_hasEcoPref| :INVERSE |hasEcoPref| :DOMAIN |EcologicalPreference|
     :RANGE |Taxon|)
(DEFINE-PRIMITIVE-ATTRIBUTE |forTheme| :DOMAIN |EcologicalPreference| :RANGE |Theme|)
(DEFINE-PRIMITIVE-ATTRIBUTE |forTheme_member| :DOMAIN |EcologicalPreference| :RANGE |Theme_Member|)
(DEFINE-PRIMITIVE-ROLE |hasEcoPref_Value| :DOMAIN |EcologicalPreference| :RANGE |EcoPref_Value|)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |units| :DOMAIN |EcologicalPreference_Continuous| :TYPE STRING)
(DEFINE-PRIMITIVE-ATTRIBUTE |forEcopref_Continuous_theme| :PARENTS |forTheme|
     :DOMAIN |EcologicalPreference_Continuous| :RANGE |Continuous_Theme|)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |forRange_minimum| :DOMAIN |EcologicalPreference_Continuous| :TYPE DOUBLE)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |forRange_maximum| :DOMAIN |EcologicalPreference_Continuous| :TYPE DOUBLE)
(DEFINE-PRIMITIVE-ATTRIBUTE |forEcopref_Continuous_member| :PARENTS |forTheme_member|
     :DOMAIN |EcologicalPreference_Continuous| :RANGE |Continuous_Theme_Member|)
(DEFINE-PRIMITIVE-ATTRIBUTE |forEcopref_Discrete_theme| :PARENTS |forTheme|
     :DOMAIN |EcologicalPreference_Discrete| :RANGE |Discrete_Theme|)
(DEFINE-PRIMITIVE-ATTRIBUTE |forEcopref_Discrete_member| :PARENTS |forTheme_member|
     :DOMAIN |EcologicalPreference_Discrete| :RANGE |Discrete_Theme_Member|)

(DEFINE-CONCEPT |EcoPref_Value| (OR |Suitable| |Moderately_suitable| |Unsuitable| |Undefined|
     |Environmental_classes_not_found_in_EO| |Water|))
(IMPLIES |EcoPref_Value| TOP)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |Value| :DOMAIN |EcoPref_Value| :TYPE INTEGER)

(IMPLIES |Extent_of_Occurrence| TOP)
(DEFINE-CONCRETE-DOMAIN-ATTRIBUTE |Greater_geographic_area| :DOMAIN |Extent_of_Occurrence| :TYPE STRING)
```

The above statements provide the TBox of our species ontology. Below, we provide an example ABox that makes use of it, for our case of 'swamp monkeys'. Observe that this ABox could easily be generated from a simple relational database, using just a few queries.

```
(IN-ABOX |SpeciesOntoDLFinal_DL.owl| |SpeciesOntoDLFinal_DL.owl|)
(INSTANCE |Cercopithecidae| |Family|)
(INSTANCE |Cercopithecinae| |Subfamily|)
(INSTANCE |Cercopithecus| |Genus|)
(INSTANCE |Distance_to_water_areas| |Continuous_Theme|)
(INSTANCE |EcoPrefE1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoPrefE2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoPrefG1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoPrefG2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefA1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefA2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefB1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefB2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC3| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC4| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC5| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC6| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC7| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC8| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefC9| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefD1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF10| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF11| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF12| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF13| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF14| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF1| |EcologicalPreference_Discrete|)
```

```
(INSTANCE |EcoprefF2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF3| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF4| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF5| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF6| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF7| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF8| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefF9| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefH1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefH2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefH3| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefI1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefI2| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefI3| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefS1| |EcologicalPreference_Discrete|)
(INSTANCE |EcoprefS2| |EcologicalPreference_Discrete|)
(INSTANCE |Elevation| |Continuous_Theme|)
(INSTANCE |Environmental_classes_not_found_in_EO| |EcoPref_Value|)
(INSTANCE |Eutheria| |Subclass|)
(INSTANCE |GroupA| |Other|)
(INSTANCE |GroupB| |Other|)
(INSTANCE |GroupC| |Other|)
(INSTANCE |GroupD| |Other|)
(INSTANCE |GroupH| |Other|)
(INSTANCE |GroupI| |Other|)
(INSTANCE |LandCover| |Discrete_Theme|)
(INSTANCE |Mammalia| |Class|)
(INSTANCE |Moderately_suitable| |EcoPref_Value|)
(INSTANCE |Primates| |Order|)
(INSTANCE |Suitable| |EcoPref_Value|)
(INSTANCE |Swamp_forest| |Discrete_Theme_Member|)
(INSTANCE |Undefined| |EcoPref_Value|)
(INSTANCE |Unsuitable| |EcoPref_Value|)
(INSTANCE |Vegetation| |Discrete_Theme|)
(INSTANCE |Water| |EcoPref_Value|)
(INSTANCE |altimontane_vegetation| |Discrete_Theme_Member|)
(INSTANCE |anthropic_landscapes| |Discrete_Theme_Member|)
(INSTANCE |azonal_vegetation| |Discrete_Theme_Member|)
(INSTANCE |barren| |Discrete_Theme_Member|)
(INSTANCE |bushland_and_thicket_mosaics| |Discrete_Theme_Member|)
(INSTANCE |bushland_and_thicket| |Discrete_Theme_Member|)
(INSTANCE |campbelli| |Species|)
(INSTANCE |cape_shrubland| |Discrete_Theme_Member|)
(INSTANCE |class| |TaxonRank|)
(INSTANCE |cropland| |Discrete_Theme_Member|)
(INSTANCE |desert| |Discrete_Theme_Member|)
(INSTANCE |edaphic_grassland_mosaics| |Discrete_Theme_Member|)
(INSTANCE |family| |TaxonRank|)
(INSTANCE |forest_transitions_and_mosaics| |Discrete_Theme_Member|)
(INSTANCE |forest_with_grass-/woodland| |Discrete_Theme_Member|)
(INSTANCE |forest_with_savanna| |Discrete_Theme_Member|)
(INSTANCE |forest| |Discrete_Theme_Member|)
(INSTANCE |fragmented_tropical_forest| |Discrete_Theme_Member|)
(INSTANCE |genus| |TaxonRank|)
(INSTANCE |grass-_and_shrubland| |Discrete_Theme_Member|)
(INSTANCE |grassland| |Discrete_Theme_Member|)
(INSTANCE |grassy_shrubland| |Discrete_Theme_Member|)
(INSTANCE |infraorder| |TaxonRank|)
(INSTANCE |mangroves/swamps-tropical_forest| |Discrete_Theme_Member|)
(INSTANCE |mangroves| |Discrete_Theme_Member|)
(INSTANCE |mona| |Species|)
(INSTANCE |montane_dry_forest| |Discrete_Theme_Member|)
(INSTANCE |montane_evergreen_forest| |Discrete_Theme_Member|)
(INSTANCE |order| |TaxonRank|)
(INSTANCE |other| |TaxonRank|)
(INSTANCE |outside_area| |Discrete_Theme_Member|)
(INSTANCE |pogonias| |Species|)
(INSTANCE |savanna| |Discrete_Theme_Member|)
(INSTANCE |secondary_tropical_forest_with_crops| |Discrete_Theme_Member|)
(INSTANCE |secondary_tropical_forest| |Discrete_Theme_Member|)
(INSTANCE |secondary_wooded_grassland| |Discrete_Theme_Member|)
(INSTANCE |semi-desert_vegetation| |Discrete_Theme_Member|)
(INSTANCE |species| |TaxonRank|)
(INSTANCE |subclass| |TaxonRank|)
(INSTANCE |subfamily| |TaxonRank|)
(INSTANCE |suborder| |TaxonRank|)
(INSTANCE |subspecies| |TaxonRank|)
(INSTANCE |superfamily| |TaxonRank|)
(INSTANCE |transitional_scrubland| |Discrete_Theme_Member|)
(INSTANCE |tribe| |TaxonRank|)
(INSTANCE |tropical_forest_with_crops| |Discrete_Theme_Member|)
(INSTANCE |tropical_rainforest_with_savanna| |Discrete_Theme_Member|)
(INSTANCE |tropical_rainforest| |Discrete_Theme_Member|)
(INSTANCE |undifferentiated_montane_vegetation-Afromontane| |Discrete_Theme_Member|)
(INSTANCE |waters| |Discrete_Theme_Member|)
(INSTANCE |water| |Discrete_Theme_Member|)
(INSTANCE |wolfi| |Species|)
(INSTANCE |woodland_mosaics_and_transitions| |Discrete_Theme_Member|)
(INSTANCE |woodlands| |Discrete_Theme_Member|)
(INSTANCE |woodland| |Discrete_Theme_Member|)
```

```
(RELATED |Cercopithecidae| |Primates| |hasAncestor|)
(RELATED |Cercopithecidae| |family| |hasRank|)
(RELATED |Cercopithecinae| |Cercopithecidae| |hasAncestor|)
(RELATED |Cercopithecinae| |EcoprefF10| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF11| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF12| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF13| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF14| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF1| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF2| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF3| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF4| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF5| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF6| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF7| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF8| |hasEcoPref|)
(RELATED |Cercopithecinae| |EcoprefF9| |hasEcoPref|)
(RELATED |Cercopithecinae| |subfamily| |hasRank|)
(RELATED |Cercopithecus| |Cercopithecinae| |hasAncestor|)
(RELATED |Cercopithecus| |genus| |hasRank|)
(RELATED |EcoPrefE1| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoPrefE1| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoPrefE1| |woodland| |forEcopref_Discrete_member|)
(RELATED |EcoPrefE2| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoPrefE2| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoPrefE2| |azonal_vegetation| |forEcopref_Discrete_member|)
(RELATED |EcoPrefG1| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoPrefG1| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoPrefG1| |woodland| |forEcopref_Discrete_member|)
(RELATED |EcoPrefG2| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoPrefG2| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoPrefG2| |woodland_mosaics_and_transitions| |forEcopref_Discrete_member|)
(RELATED |EcoprefA1| |Environmental_classes_not_found_in_EO| |hasEcoPref_Value|)
(RELATED |EcoprefA1| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefA1| |outside_area| |forEcopref_Discrete_member|)
(RELATED |EcoprefA2| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefA2| |Water| |hasEcoPref_Value|)
(RELATED |EcoprefA2| |water| |forEcopref_Discrete_member|)
(RELATED |EcoprefB1| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefB1| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefB1| |forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefB2| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefB2| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefB2| |forest_transitions_and_mosaics| |forEcopref_Discrete_member|)
(RELATED |EcoprefC1| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefC1| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC1| |undifferentiated_montane_vegetation-Afromontane| |forEcopref_Discrete_member|)
(RELATED |EcoprefC2| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefC2| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC2| |forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefC3| |Swamp_forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefC3| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefC3| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC4| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefC4| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC4| |forest_transitions_and_mosaics| |forEcopref_Discrete_member|)
(RELATED |EcoprefC5| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefC5| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC5| |woodland| |forEcopref_Discrete_member|)
(RELATED |EcoprefC6| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefC6| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC6| |woodland_mosaics_and_transitions| |forEcopref_Discrete_member|)
(RELATED |EcoprefC7| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefC7| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC7| |bushland_and_thicket_mosaics| |forEcopref_Discrete_member|)
(RELATED |EcoprefC8| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefC8| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC8| |grassland| |forEcopref_Discrete_member|)
(RELATED |EcoprefC9| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefC9| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefC9| |altimontane_vegetation| |forEcopref_Discrete_member|)
(RELATED |EcoprefD1| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefD1| |Vegetation| |forEcopref_Discrete_theme|)
(RELATED |EcoprefD1| |azonal_vegetation| |forEcopref_Discrete_member|)
(RELATED |EcoprefF10| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF10| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefF10| |mangroves| |forEcopref_Discrete_member|)
(RELATED |EcoprefF10| |mangroves| |forTheme_member|)
(RELATED |EcoprefF11| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF11| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefF11| |montane_dry_forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefF12| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF12| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefF12| |tropical_forest_with_crops| |forEcopref_Discrete_member|)
(RELATED |EcoprefF13| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF13| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefF13| |forest_with_savanna| |forEcopref_Discrete_member|)
(RELATED |EcoprefF14| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF14| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefF14| |forest_with_grass-/woodland| |forEcopref_Discrete_member|)
(RELATED |EcoprefF1| |LandCover| |forEcopref_Discrete_theme|)
```

```
(RELATED |EcoprefF1| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefF1| |cropland| |forEcopref_Discrete_member|)
(RELATED |EcoprefF2| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF2| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefF2| |grass-_and_shrubland| |forEcopref_Discrete_member|)
(RELATED |EcoprefF3| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF3| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefF3| |savanna| |forEcopref_Discrete_member|)
(RELATED |EcoprefF4| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF4| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefF4| |woodlands| |forEcopref_Discrete_member|)
(RELATED |EcoprefF5| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF5| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefF5| |montane_evergreen_forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefF6| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF6| |Unsuitable| |hasEcoPref_Value|)
(RELATED |EcoprefF6| |barren| |forEcopref_Discrete_member|)
(RELATED |EcoprefF7| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF7| |Water| |hasEcoPref_Value|)
(RELATED |EcoprefF7| |waters| |forEcopref_Discrete_member|)
(RELATED |EcoprefF8| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF8| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefF8| |tropical_rainforest| |forEcopref_Discrete_member|)
(RELATED |EcoprefF9| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefF9| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefF9| |fragmented_tropical_forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefH1| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefH1| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefH1| |secondary_tropical_forest_with_crops| |forEcopref_Discrete_member|)
(RELATED |EcoprefH2| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefH2| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefH2| |secondary_tropical_forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefH3| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefH3| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefH3| |tropical_rainforest_with_savanna| |forEcopref_Discrete_member|)
(RELATED |EcoprefI1| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefI1| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefI1| |secondary_tropical_forest_with_crops| |forEcopref_Discrete_member|)
(RELATED |EcoprefI2| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefI2| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefI2| |secondary_tropical_forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefI3| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefI3| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefI3| |tropical_rainforest_with_savanna| |forEcopref_Discrete_member|)
(RELATED |EcoprefS1| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefS1| |Suitable| |hasEcoPref_Value|)
(RELATED |EcoprefS1| |mangroves/swamps-tropical_forest| |forEcopref_Discrete_member|)
(RELATED |EcoprefS2| |LandCover| |forEcopref_Discrete_theme|)
(RELATED |EcoprefS2| |Moderately_suitable| |hasEcoPref_Value|)
(RELATED |EcoprefS2| |mangroves/swamps-tropical_forest| |forEcopref_Discrete_member|)
(RELATED |Eutheria| |Mammalia| |hasAncestor|)
(RELATED |Eutheria| |subclass| |hasRank|)
(RELATED |GroupA| |Cercopithecus| |hasAncestor|)
(RELATED |GroupA| |EcoprefA1| |hasEcoPref|)
(RELATED |GroupA| |EcoprefA2| |hasEcoPref|)
(RELATED |GroupA| |other| |hasRank|)
(RELATED |GroupB| |EcoprefB1| |hasEcoPref|)
(RELATED |GroupB| |EcoprefB2| |hasEcoPref|)
(RELATED |GroupB| |GroupA| |hasAncestor|)
(RELATED |GroupB| |other| |hasRank|)
(RELATED |GroupC| |EcoprefC1| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC2| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC3| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC4| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC5| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC6| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC7| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC8| |hasEcoPref|)
(RELATED |GroupC| |EcoprefC9| |hasEcoPref|)
(RELATED |GroupC| |GroupA| |hasAncestor|)
(RELATED |GroupC| |other| |hasRank|)
(RELATED |GroupD| |EcoprefD1| |hasEcoPref|)
(RELATED |GroupD| |GroupB| |hasAncestor|)
(RELATED |GroupD| |other| |hasRank|)
(RELATED |GroupH| |EcoprefH1| |hasEcoPref|)
(RELATED |GroupH| |EcoprefH2| |hasEcoPref|)
(RELATED |GroupH| |EcoprefH3| |hasEcoPref|)
(RELATED |GroupH| |GroupA| |hasAncestor|)
(RELATED |GroupH| |other| |hasRank|)
(RELATED |GroupI| |EcoprefI1| |hasEcoPref|)
(RELATED |GroupI| |EcoprefI2| |hasEcoPref|)
(RELATED |GroupI| |EcoprefI3| |hasEcoPref|)
(RELATED |GroupI| |GroupA| |hasAncestor|)
(RELATED |GroupI| |other| |hasRank|)
(RELATED |LandCover| |barren| |hasDiscreteTheme_member|)
(RELATED |LandCover| |cropland| |hasDiscreteTheme_member|)
(RELATED |LandCover| |forest_with_grass-/woodland| |hasDiscreteTheme_member|)
(RELATED |LandCover| |forest_with_savanna| |hasDiscreteTheme_member|)
(RELATED |LandCover| |fragmented_tropical_forest| |hasDiscreteTheme_member|)
(RELATED |LandCover| |grass-_and_shrubland| |hasDiscreteTheme_member|)
(RELATED |LandCover| |mangroves/swamps-tropical_forest| |hasDiscreteTheme_member|)
```

```
(RELATED |LandCover| |mangroves| |hasDiscreteTheme_member|)
(RELATED |LandCover| |montane_dry_forest| |hasDiscreteTheme_member|)
(RELATED |LandCover| |montane_evergreen_forest| |hasDiscreteTheme_member|)
(RELATED |LandCover| |savanna| |hasDiscreteTheme_member|)
(RELATED |LandCover| |secondary_tropical_forest_with_crops| |hasDiscreteTheme_member|)
(RELATED |LandCover| |secondary_tropical_forest| |hasDiscreteTheme_member|)
(RELATED |LandCover| |tropical_forest_with_crops| |hasDiscreteTheme_member|)
(RELATED |LandCover| |tropical_rainforest_with_savanna| |hasDiscreteTheme_member|)
(RELATED |LandCover| |tropical_rainforest| |hasDiscreteTheme_member|)
(RELATED |LandCover| |waters| |hasDiscreteTheme_member|)
(RELATED |LandCover| |woodlands| |hasDiscreteTheme_member|)
(RELATED |Mammalia| |class| |hasRank|)
(RELATED |Primates| |Eutheria| |hasAncestor|)
(RELATED |Primates| |order| |hasRank|)
(RELATED |Vegetation| |Swamp_forest| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |altimontane_vegetation| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |anthropic_landscapes| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |azonal_vegetation| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |bushland_and_thicket_mosaics| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |bushland_and_thicket| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |cape_shrubland| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |desert| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |edaphic_grassland_mosaics| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |forest_transitions_and_mosaics| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |forest| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |grassland| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |grassy_shrubland| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |outside_area| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |secondary_wooded_grassland| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |semi-desert_vegetation| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |transitional_scrubland| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |undifferentiated_montane_vegetation-Afromontane| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |woodland_mosaics_and_transitions| |hasDiscreteTheme_member|)
(RELATED |Vegetation| |woodland| |hasDiscreteTheme_member|)
(RELATED |campbelli| |EcoprefS1| |hasEcoPref|)
(RELATED |campbelli| |GroupD| |hasAncestor|)
(RELATED |campbelli| |GroupH| |hasAncestor|)
(RELATED |campbelli| |species| |hasRank|)
(RELATED |mona| |EcoPrefG1| |hasEcoPref|)
(RELATED |mona| |EcoPrefG2| |hasEcoPref|)
(RELATED |mona| |EcoprefS1| |hasEcoPref|)
(RELATED |mona| |GroupD| |hasAncestor|)
(RELATED |mona| |GroupH| |hasAncestor|)
(RELATED |mona| |species| |hasRank|)
(RELATED |pogonias| |EcoPrefE1| |hasEcoPref|)
(RELATED |pogonias| |EcoPrefE2| |hasEcoPref|)
(RELATED |pogonias| |EcoprefS2| |hasEcoPref|)
(RELATED |pogonias| |GroupB| |hasAncestor|)
(RELATED |pogonias| |GroupI| |hasAncestor|)
(RELATED |pogonias| |species| |hasRank|)
(RELATED |wolfi| |EcoprefS1| |hasEcoPref|)
(RELATED |wolfi| |GroupC| |hasAncestor|)
(RELATED |wolfi| |species| |hasRank|)

(CONSTRAINED |Cercopithecidae| O37 |Taxon_name|)
(CONSTRAINED |Cercopithecinae| O38 |Taxon_name|)
(CONSTRAINED |Cercopithecus| O33 |Taxon_name|)
(CONSTRAINED |Distance_to_water_areas| O3 |Theme_name|)
(CONSTRAINED |Elevation| O2 |Theme_name|)
(CONSTRAINED |Environmental_classes_not_found_in_EO| O60 |Value|)
(CONSTRAINED |Eutheria| O34 |Taxon_name|)
(CONSTRAINED |GroupA| O9 |Taxon_name|)
(CONSTRAINED |GroupB| O5 |Taxon_name|)
(CONSTRAINED |GroupC| O11 |Taxon_name|)
(CONSTRAINED |GroupD| O6 |Taxon_name|)
(CONSTRAINED |GroupH| O8 |Taxon_name|)
(CONSTRAINED |GroupI| O1 |Taxon_name|)
(CONSTRAINED |LandCover| O41 |Theme_name|)
(CONSTRAINED |Mammalia| O35 |Taxon_name|)
(CONSTRAINED |Moderately_suitable| O63 |Value|)
(CONSTRAINED |Primates| O36 |Taxon_name|)
(CONSTRAINED |Suitable| O64 |Value|)
(CONSTRAINED |Swamp_forest| O15 |Theme_Member_Code|)
(CONSTRAINED |Undefined| O61 |Value|)
(CONSTRAINED |Unsuitable| O62 |Value|)
(CONSTRAINED |Vegetation| O22 |Theme_name|)
(CONSTRAINED |Water| O59 |Value|)
(CONSTRAINED |altimontane_vegetation| O28 |Theme_Member_Code|)
(CONSTRAINED |anthropic_landscapes| O18 |Theme_Member_Code|)
(CONSTRAINED |azonal_vegetation| O31 |Theme_Member_Code|)
(CONSTRAINED |barren| O39 |Theme_Member_Code|)
(CONSTRAINED |bushland_and_thicket_mosaics| O58 |Theme_Member_Code|)
(CONSTRAINED |bushland_and_thicket| O13 |Theme_Member_Code|)
(CONSTRAINED |campbelli| O4 |Taxon_name|)
(CONSTRAINED |cape_shrubland| O14 |Theme_Member_Code|)
(CONSTRAINED |cropland| O55 |Theme_Member_Code|)
(CONSTRAINED |desert| O21 |Theme_Member_Code|)
(CONSTRAINED |edaphic_grassland_mosaics| O30 |Theme_Member_Code|)
(CONSTRAINED |forest_transitions_and_mosaics| O16 |Theme_Member_Code|)
(CONSTRAINED |forest_with_grass-/woodland| O52 |Theme_Member_Code|)
(CONSTRAINED |forest_with_savanna| O50 |Theme_Member_Code|)
```

```
(CONSTRAINED |forest| O17 |Theme_Member_Code|)
(CONSTRAINED |fragmented_tropical_forest| O56 |Theme_Member_Code|)
(CONSTRAINED |grass-_and_shrubland| O54 |Theme_Member_Code|)
(CONSTRAINED |grassland| O23 |Theme_Member_Code|)
(CONSTRAINED |grassy_shrubland| O27 |Theme_Member_Code|)
(CONSTRAINED |mangroves/swamps-tropical_forest| O49 |Theme_Member_Code|)
(CONSTRAINED |mangroves| O47 |Theme_Member_Code|)
(CONSTRAINED |mona| O7 |Taxon_name|)
(CONSTRAINED |montane_dry_forest| O43 |Theme_Member_Code|)
(CONSTRAINED |montane_evergreen_forest| O57 |Theme_Member_Code|)
(CONSTRAINED |outside_area| O32 |Theme_Member_Code|)
(CONSTRAINED |pogonias| O0 |Taxon_name|)
(CONSTRAINED |savanna| O40 |Theme_Member_Code|)
(CONSTRAINED |secondary_tropical_forest_with_crops| O42 |Theme_Member_Code|)
(CONSTRAINED |secondary_tropical_forest| O51 |Theme_Member_Code|)
(CONSTRAINED |secondary_wooded_grassland| O29 |Theme_Member_Code|)
(CONSTRAINED |semi-desert_vegetation| O20 |Theme_Member_Code|)
(CONSTRAINED |transitional_scrubland| O19 |Theme_Member_Code|)
(CONSTRAINED |tropical_forest_with_crops| O46 |Theme_Member_Code|)
(CONSTRAINED |tropical_rainforest_with_savanna| O48 |Theme_Member_Code|)
(CONSTRAINED |tropical_rainforest| O53 |Theme_Member_Code|)
(CONSTRAINED |undifferentiated_montane_vegetation-Afromontane| O25 |Theme_Member_Code|)
(CONSTRAINED |waters| O45 |Theme_Member_Code|)
(CONSTRAINED |water| O10 |Theme_Member_Code|)
(CONSTRAINED |wolfi| O12 |Taxon_name|)
(CONSTRAINED |woodland_mosaics_and_transitions| O24 |Theme_Member_Code|)
(CONSTRAINED |woodlands| O44 |Theme_Member_Code|)
(CONSTRAINED |woodland| O26 |Theme_Member_Code|)

(CONSTRAINTS (EQUAL O64 1) (EQUAL O63 2) (EQUAL O62 3) (EQUAL O61 4) (EQUAL O60 5) (EQUAL O59 8) (EQUAL O58 700) (EQUAL O57 5)
     (EQUAL  O56 9) (EQUAL  O55 1) (EQUAL  O54 2) (EQUAL  O53 8) (EQUAL  O52 14) (EQUAL  O51 128) (EQUAL  O50 13)
     (EQUAL  O49 187) (EQUAL  O48 129) (EQUAL  O47 10) (EQUAL  O46 12) (EQUAL  O45 7) (EQUAL  O44 4) (EQUAL  O43 11)
     (EQUAL  O42 30) (EQUAL  O40 3) (EQUAL  O39 6) (EQUAL  O32 0) (EQUAL  O31 1600) (EQUAL  O30 1300) (EQUAL  O29 500)
     (EQUAL  O28 1400) (EQUAL  O27 1100) (EQUAL  O26 300) (EQUAL  O25 150) (EQUAL  O24 400) (EQUAL  O23 1200)
     (EQUAL  O21 1500) (EQUAL  O20 1000) (EQUAL  O19 800) (EQUAL  O18 1700) (EQUAL  O17 100) (EQUAL  O16 200) (EQUAL  O15 160)
     (EQUAL  O14 900) (EQUAL  O13 600) (EQUAL  O10 9999)
     (=CONSTANT |Theme_name| O41 "LandCover") (=CONSTANT |Taxon_name| O38 "Cercopithecinae")
     (=CONSTANT |Taxon_name| O37 "Cercopithecidae") (=CONSTANT |Taxon_name| O36 "Primates")
     (=CONSTANT |Taxon_name| O35 "mammalia") (=CONSTANT |Taxon_name| O34 "Eutheria")
     (=CONSTANT |Taxon_name| O33 "Cercopithecus") (=CONSTANT |Theme_name| O22 "Vegetation")
     (=CONSTANT |Taxon_name| O12 "wolfi") (=CONSTANT |Taxon_name| O11 "GroupC") (=CONSTANT |Taxon_name| O9 "GroupA")
     (=CONSTANT |Taxon_name| O8 "GroupH") (=CONSTANT |Taxon_name| O7 "mona") (=CONSTANT |Taxon_name| O6 "GroupD")
     (=CONSTANT |Taxon_name| O5 "GroupB") (=CONSTANT |Taxon_name| O4 "campbelli")
     (=CONSTANT |Theme_name| O3 "Water_areas") (=CONSTANT |Theme_name| O2 "Elevation")
     (=CONSTANT |Taxon_name| O1 "GroupI") (=CONSTANT |Taxon_name| O0 "pogonias"))
```

Bibliography

[1] R. A. Aspinall and N. Veitch. Habitat mapping from satellite imagery and wildlife survey data using a bayesian modelling procedure in a GIS. *Photographic Engineering and Remote Sensing*, 59(4):537–543, 1993. 17

[2] R. C. Balling, Jr. *The Heated Debate: Greenhouse Predictions versus Climate Reality*. Pacific Research Institute for the Public, 1993. 1

[3] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. OWL Web Ontology Language reference. Technical report, World Wide Web Consortium (W3C), 2004. www.w3.org/TR/2004/REC-owl-ref-20040210/. 28

[4] J. Bekhuis et al., editors. *Atlas van de Nederlandse Vogels*. SOVON, 1987. 15

[5] F. A. Bink. *Ecologische Atlas van de Dagvlinders van Noordwest-Europa*. Schuyt & Co, Haarlem, 1992. 15

[6] M. R. Blaha, W. J. Premerlani, and J. E. Rumbaugh. Relational database design using an object-oriented methodology. *Communications of the ACM*, 31(4):414–427, 1988. 28

[7] A. Borgida and R. J. Brachman. Conceptual modeling with Description Logic. In F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors, *The Description Logic Handbook*, pages 349–372. Cambridge University Press, Cambridge, UK, 2003. 34, 37, 38

[8] A. H. Borning and D. H. H. Ingalls. A type declaration and inference system for Smalltalk. In *Proceedings of the 9th ACM SIGPLAN-SIGACT Symposium on Principles of programming languages*, pages 133–141. ACM Press, 1982. 28

[9] L. Botani, F. Corsi, A. De Biase, I. D. Carranza, M. Ravagli, G. Reggiani, I. Sinibaldi, and P. Trapanese. A databank for the conservation and management of the African mammals. Technical report, IEA—Istituto di Ecologia Applicata, Rome, Italy, 1999. 4, 9, 22, 47

[10] R. J. Brachman and J. G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2):171–216, 1985. 27

[11] T. Bray, J. Paoli, and C. M. Sperberg-McQueen. Extensible markup language (XML) 1.0. Technical report, World Wide Web Consortium (W3C), 1998. W3C Recommendation 10-February-1998. 28

[12] V. Brilhante and D. Robertson. Metadata-supported automated ecological modelling. citeseer.nj.nec.com/305083.html (accessed 19.11.2003). 4

[13] D. Bryant, D. Nielsen, and L. Tangley. *Last Frontier Forests: Ecosystems and Economies on the Edge*. World Resources Institute, 1997. 1

[14] R. Butterfield. Numbers of species in the major groups of organisms. Internet, 2002. www.btinternet.com/~rb.freelance/courses/inh/handouts/relativeabundance.htm (accessed 12.01.2004). 11

[15] P. P.-S. S. Chen. The Entity-Relationship model: Toward a unified view of data. *ACM Transactions on Database Systems*, 1(1):9–36, 1976. 28

[16] D. Connolly, F. van Harmelen, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. DAML+OIL (march 2001) reference description. Technical report, World Wide Web Consortium (W3C), 2001. W3C Note 18 December 2001. 28

[17] F. Corsi. Mapping animal species distributions for conservation: How to get the most from limited data., 2003. Presentation to the ITC-NRS program Research day. 4

[18] F. Corsi, J. de Leeuw, and A. Skidmore. Modeling species distribution with GIS. In L. Botani and T. K. Fuller, editors, *Research Techniques in Animal Ecology*, pages 389–434. Columbia University Press, New York, 2000. 4, 16, 17, 18, 19

[19] B. Csuti and P. Crist. Methods for developing terrestrial vertebrate distribution maps for GAP analysis. Technical report, Idaho Cooperative Fish and Wildlife Research Unit, Univerity of Idaho, Moscow, ID, 1998. www.gap.uidaho.edu/handbook/VertebrateDistributionModeling (accessed 22.03.2003). 4

[20] A. Cuthbert, B. O'Rourke, E. Keighan, S. Margoulies, J. Sharma, P. Daisey, R. Lake, and S. Johnson. Geography markup language (GML) v1.0. Technical report, Open GIS Consortium (OGC), 2000. OGC Document Number: 00–029. 28

[21] M. C. Daconta, L. J. Orbst, and K. T. Smith. *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*. Wiley Publishing, Inc., Indianapolis, Indiana, 2003. 47

[22] O.-J. Dahl and K. Nygaard. Simula: an algol-based simulation language. *Communications of the ACM*, 9(9):671–678, 1966. 28

[23] K. de Queiroz. Systematics and the Darwinian revolution. *Philosophy of Science*, 55:238–259, 1988. 20

[24] K. de Queiroz and J. Gauthier. Phylogeny as a central principle in taxonomy: Phylogenetic definitions of taxon names. *Systematic Zoology*, 39(4):307–322, 1990. 20

[25] Department of Geography, University at Buffalo. GIS for environmental modeling, 2004. www.geog.buffalo.edu/~lbian/envdelphi.html (accessed 10.02.2004). 17

[26] Division of Ecological Services. Standards for the developement of habitat suitability models. Technical report, U.S. Fish and Wildlife Service, 1981. 13

[27] T. A. Dreisbach, J. E. Smith, and R. Molina. Challenges of modeling fungal habitat: When and where do you find chanterelles? Internet, 1999. www.cnr.uidaho.edu/coop/Symposium%20abstracts.htm (accessed 18.04.2003). 12

[28] S. M. Embury, A. C. Jones, I. Sutherland, W. A. Gray, R. J. White, J. S. Robinson, F. A. Bisby, and S. M. Brandt. Conflict detection for integration of taxonomic data sources. In *Statistical and Scientific Database Management*, pages 204–213, 1999. 20

[29] A. Felfernig, G. Friedrich, D. Jannach, and M. Zanker. A joint foundation for configuration in the Semantic Web. In *Proceedings 15th European Conference on Artificial Intelligence — Configuration Workshop*, pages 89–94. ECAI, 2002. 59, 63, 64, 65

[30] D. L. Garshelis. Delusions in habitat evaluation: Measuring use, selection, and importance. In L. Botani and T. K. Fuller, editors, *Research Techniques in Animal Ecology*, pages 111–164. Columbia University Press, New York, 2000. 12, 13

[31] A. Goldberg and D. Robson. *SMALLTALK-80: The Language and its Implementation*. Addison-Wesley Longman Publishing Co., Inc., 1983. 28

[32] N. Guarino and C. Welty. Ontological analysis of taxonomic relationships. In A. H. F. Laender, S. W. Liddle, and V. C. Storey, editors, *Proceedings International Conference on Conceptual Modeling / the Entity Relationship Approach (ER2000)*, volume 1920 of *Lecture Notes in Computer Science*, pages 210–224. Springer-Verlag, October 2000. 37

[33] A. Guisan, J. Edwards, Thomas C., and T. Hastie. Generalized linear and generalized additive models in studies of species distributions: Setting the scene. *Ecological Modelling*, 157(2–3):89–100, 2002. 4, 18

[34] A. Guisan and N. E. Zimmermann. Predicted habitat distribution models in ecology. *Ecological Modelling*, 135:147–186, 2000. 2, 4, 16, 17, 18, 19, 23, 24

[35] V. Haarslev and R. Möller. Description of the RACER system and its applications. In C. Goble, D. L. McGuinness, R. Möller, and P. F. Patel-Schneider, editors, *Working Notes of the 2001 International Description*

Logics Workshop (DL-2001), pages 132–141, Stanford, Ca, USA, 2001. ceur-ws.org/Vol-49. 29, 31, 32

[36] V. Haarslev and R. Möller. RACER user's guide and reference manual, version 1.7.7. Technical report, Concordia University, Montreal, Canada & University of Applied Sciences, Wedel, Germany, November 2003. www.fh-wedel.de/~mo/racer/. 32, 51

[37] V. Haarslev and R. Möller. The Racer Query Language–RQL. Technical report, Concordia University, Montreal, Canada & University of Applied Sciences, Wedel, Germany, April 2004. ?? 56

[38] A. M. Herr and L. P. Queen. Crane habitat evaluation using GIS and Remote Sensing. *Photographic Engineering and Remote Sensing*, 59:1531–1538, 1993. 17

[39] I. Horrocks, U. Sattler, and S. Tobies. Reasoning with individuals for the description logic $\mathcal{SHIQ}$. In D. MacAllester, editor, *Proc. of the 17th Int. Conf. on Automated Deduction (CADE 2000)*, number 1831 in Lecture Notes in Artificial Intelligence, pages 482–496. Springer-Verlag, 2000. download/2000/CADE17.ps.gz. 32

[40] R. Hull and R. King. Semantic database modeling: Survey, applications, and research issues. *ACM Computing Surveys*, 19(3):201–260, 1987. 28

[41] IEA (Institute of Applied Ecology). *African Mammals Databank — A Databank for the Conservation and Management of the African Mammals*, volume 1 and 2. European Commission Directorate, Bruxelles, 1998. Report to the Directorate-General for Development (DGVIII/A/1) of the European Commission. Project No. B7-6200/94-15/VIII/ENV. 20, 48

[42] A. S. Islam, L. Bermudez, M. Piasecki, and S. Fellah. Ontology for geographic information - metadata (iso 19115). Internet, February 2004. http://loki.cae.drexel.edu/~wbs/ontology/. 54

[43] M. L. Isler. A sector-based ornithological geographical information system for the Neotropics. In J. J. V. Remsen, editor, *Ornithological Monographs*, volume 48, pages 345–354. American Ornithologists' Union, Washington, D.C., 1997. 2

[44] P. R. Kemp. Ecology & the biosphere. Internet, ? home.sandiego.edu/~pkemp/Ecology.html (accessed 01.12.2003). 12

[45] J. Kingdon. *The Kingdon Field Guide to African Mammals*. Academic Press, 1997. 20, 49

[46] A. D. Kliskey, E. C. Lofroth, W. A. Thompson, S. Brown, and H. Schreier. Simulating and evaluating alternative resource-use strategies using GIS-based habitat suitability indices. *Landscape and Urban Planning*, 45(4):163–175, 1999. TY-JOUR. 13

[47] B. Loiselle. Modeling species distributions: Inductive approaches including important environmental parameters—a case study from the Atlantic forests of Brazil. Internet. cabs.kms.conservation.org//WOMBAT/DynamicFile/Object4_Indexed/0xd18d5727cbe43f49b361da0fd67c3904.pdf. 4

[48] M. E. S. Loomis, A. V. Shah, and J. E. Rumbaugh. An object modelling technique for conceptual design. In *European conference on object-oriented programming on ECOOP '87*, pages 192–202. Springer-Verlag, 1987. 28

[49] B. G. Mackey. The role of GIS and environmental modelling in the conservation of biodiversity. Third annual conference/workshop on GIS and environmental modelling, National Centre for Geographical Information Anallysis, Santa Fe, USA, January 1996, Santa Barbara., 1996. www.sbg.ac.at/geo/idrisi/gis_environmental_modeling/sf_papers/brendan_mackey/mackey_paper.html (accessed 16.02.2004). 11

[50] S. Manel, J.-M. Dias, and S. J. Ormerod. Comparing discriminant analysis, neural networks and logistic regression for predicting species distributions: A case study with a Himalayan river bird. *Ecological Modelling*, 120(2–3):337–347, 1999. 4

[51] D. L. McGuinness. *The Description Logic Handbook*, chapter Configuration, pages 388–405. Cambridge University Press, Cambridge, UK, 2003. 63

[52] D. H. Meadows, D. L. Meadows, J. Randers, and I. William W. Behrens, editors. *The Limits to Growth: A Report for The Club of Rome's Project on the Predicament of Mankind*. Universe Books, New York, 1972. The Club of Rome report. 1

[53] G. Nelson. Why, after all, must it? *Cladistics*, 8:139–146, 1992. 20

[54] R. G. Ortigosa, G. A. de Leo, and M. Gatto. VVF: Integrating modelling and GIS in a software tool for habitat suitability assessment. *Environmental Modelling and Software*, 15(1):1–12, 2002. vii, 4, 21, 22

[55] A. T. Peterson, A. G. Navarro-Sigüenza, and H. Benítez-Díaz. The need for continued scientific collecting: A geographic analysis of Mexican bird specimens. *Ibis*, 140:288–294, 1998. Reference not found yet. 5

[56] E. J. Proctor, S. D. Blum, and G. Chaplin. A software tool for retrospectively georeferencing specimen localities using arcview. Internet, 2001. www.calacademy.org/research/informatics/georef/Main_Pages/2_Background.html (accessed 27.02.2004). 20

[57] M. R. Quillian. Semantic memory. In M. Minsky, editor, *Semantic Information Processing*, pages 227–270. MIT Press, Cambridge, MA, 1968. 27

[58] J. Rumbaugh, I. Jacobson, and G. Booch. *The Unified Modeling Language Reference Manual*. Addison-Wesley Longman Ltd., 1999. 28

[59] A. K. Skidmore. Taxonomy of environmental models in the spatial sciences. In A. K. Skidmore, editor, *Environmental Modelling with GIS and Remote Sensing*, pages 8–25. Taylor & Francis, London, 2002. 4

[60] D. R. B. Stockwell. SEEK overview, 2002. seek.ecoinformatics.org/overview.html (accessed 24.03.2003). 4

[61] M. M. University. GIS and conservation. Internet, 2004. 149.170.199.144/new_gis/unitintr.htm (accessed 11.01.2004). 11

[62] USGS. Traditional approaches to mapping species distributions. Internet, 2004. www.gap.uidaho.edu/About/Overview/wildlifemonographs/PADSR.htm##TraditionalApproachestoMappingSpeciesDistributions (accessed 21.04.2003). 14

[63] A. B. van den Berg and C. A. W. Bosman. *Rare Birds of the Netherlands*. GMB Uitgeverij, Haarlem, 1999. 15

[64] U. Visser, H. Stuckenschmidt, G. Schuster, and T. Vogele. Ontologies for geographic information processing. *Computers & Geosciences*, 28(1):103–117, 2002. TY - JOUR. 23

[65] I. N. Vogiatzakis. GIS-based modelling and ecology: A review of tools and methods. Technical report, The University of Reading, UK, 2003. www.geog.rdg.ac.uk/Research/Papers/GP170.pdf. 4, 16

[66] C. Welty and N. Guarino. Supporting ontological analysis of taxonomic relationships. *Data & Knowledge Engineering*, 39(1):51–74, October 2001. 37