# Building Shared Ontologies for Terminology Integration

Gerhard Schuster and Heiner Stuckenschmidt

Center for Computing Technologies
University of Bremen, Germany

**Abstract.** We present an approach for developing shared ontologies which can be used to define terms from different vocabularies and to automatically translate them from one vocabulary into another. We first motivate the use of shared ontologies as a means for identifying semantic correspondences between terms and underline the need for a shared ontology. Then, the general steps of a specialized methodology for building such shared ontologies are described. We further illustrate the application of the methodology using a real-life example from the domain of geo-informatics.

## 1 Introduction

The exchange on information has become a crucial factor in today's economy. Many processes in business and administration involve different organizations that have to work together in order to reach a common goal. Further, the collection and maintenance of information is a time-consuming and costly effort that leads to the need of using existing information whenever possible. The most prominent example of an information repository accessible in principle is the World Wide Web with its millions of pages. However, the World Wide Web is also the best example of the problems that are connected with the exchange of information. Most of these problems are related to the heterogeneity of information sources. On the World Wide Web syntactic heterogeneity in terms of different encoding languages and formats is more or less solved by the standardization efforts of the World Wide Web Consortium W3C. However, the heterogeneity in the information itself is even more striking: using XML users are able to build their own data structures and terminologies. While the structures can be adopted almost without loss of information using schema information provided by document type definitions or XML schemas the use of different terminologies may lead to serious problems when the intuitive meaning of the elements from the terminology is not clear and the understanding of the information provider differs from that of the user.

We propose an approach for finding mappings between classes from different ontologies that combines the ideas of comparing class structures and definitions with the one of using common upper-level ontologies. Our approach relies on

the definition of a common terminology similar to a upper-level ontology. The elements of this common terminology are used to describe classes from different ontologies in terms of formal concept expressions. This formal model can be used to derive correspondences between classes on a well-founded logical level. The approach will be described in section 2. Our approach relies on the existence of an explicitly shared terminology that is expressive enough to form the basis of concept expressions for the classes of all ontologies to be compared. On the other hand, the terminology should be as small as possible in order to ease comparison and reduce the effort of building it. Therefore a specialized method is required to build such shared terminologies. Such a method will be presented in section 3 of this paper. It has to be more specific than existing proposals for ontology engineering methodologies (e.g. [11],[3],[4]), because it has to take the specific needs of the integration approach into account. We present this method as well as the general approach for relating classes using a real-life example that is still small enough to be comprehensive.

## 2 Ontology-Based Terminology Integration

Our approach to the semantic integration problem is based on the view that each information source serves as a context for the interpretation of the information contained therein. This view implies that an information entity can only be completely understood within its source unless we find ways to preserve the contextual information in the translation process. This claim has two implications:

1. We have to represent the context of an information entity given by its source
2. We have to use this contextual information to integrate an entity into the new context given by the target of the translation

We argued that contextual knowledge of an information entity can be represented by necessary and sufficient conditions for deciding whether an entity belongs to a certain class of objects [9]. Using these conditions the integration of an entity in a new context is equivalent with a classification that is based on its contextual knowledge. Details of this approach are given below.

### 2.1 Specifying Information Context

In information sources contextual knowledge is often hidden in type information. Most information sources are based on a data model describing classes, attributes and relations. Each entity within the information source is assigned to one of these categories we will refer to as 'concepts' in the sequel. Depending on the intended use of the information source each concept is assumed to serve a special function and to show special properties necessary for that function. Some of these properties will explicitly be contained in the information source other properties remain implicit because there is a silent agreement that

a property always holds. In order to support semantic translation we have to explicate these hidden assumptions by defining necessary and sufficient conditions an information entity has to fulfill in order to belong to that concept.

*Necessary Conditions:* Concepts are described by a set of necessary conditions in terms of values of some properties $p_i$. We write $p_i^X$ to denote that the entity $X$ shows property $p_i$. We claim that there are properties that are characteristic for a concept and can therefore always be observed for instances of that class. We write $\mathcal{N}_C = \{p_1, \cdots, p_m\}$ to denote that the concept $c$ has necessary conditions $p_1, \cdots, p_m$. Assuming that class and property definitions always refer to the same entity $X$ we get the following equation:

$$N^c \equiv c(X) \Rightarrow p_i^X \wedge \cdots \wedge p_m^X \tag{1}$$

*Sufficient Conditions:* On the other hand, we assume that an entity automatically belongs to the concept $c$ if it shows sufficient characteristic properties. We write $\mathcal{S}_C = \{p_1, \cdots, p_n\}$ to denote that $p_1 \cdots, p_n$ are sufficient conditions indicating that X belongs to the concept c. We characterize the class $c$ by the following equation:

$$S^c \equiv p_1^X \wedge \cdots \wedge p_n^X \Rightarrow c(X) \tag{2}$$

The distinction between necessary and sufficient conditions for concept membership enables us to identify entities that definitely belong to a concept because they show all sufficient conditions. On the other hand, we can identify entities that clearly do not belong to the concept, because they do not fulfill the necessary conditions.

### 2.2 Context Transformation

Concepts identify common properties of their members by defining necessary conditions for a membership. A classification problem is characterized by the determination of membership relations between an object under consideration and a set of predefined concepts. The identification process starts with data about the object that has to be classified. This data is provided by so-called observation. In the course of the classification the observed data is matched against the necessary conditions provided by the class definitions leading to one or more classes. The match between observations and membership conditions is performed using knowledge that associates properties of objects with their class. This view on classification can be formalized in the following way [6]:

- Let $C$ be a set of solution classes (in our case concept predicates $\{c_1, \ldots, c_m\}$)
- Let $O$ be a set of Observations (in our case the necessary conditions for concept membership $\{N^c | c \in C\}$)
- Let $R$ be a set of classification rules (in our case sufficient conditions for class membership $\{S^c | c \in C\}$)

Then in principle a classification task is to find a solution class $c_i \in C$ in such a way, that

$$O \wedge R \Rightarrow c_i(X) \tag{3}$$

In terms of the definitions given above, semantic translation is equivalent to a re-classification of entities already classified in one semantic structure $C^S = \{c_1^S, \cdots, c_n^S\}$ using another semantic structure $C^T = \{c_1^T, \cdots, c_m^T\}$. The process of re-classification can be based upon the semantic characterizations given by both structures. The source structure provides the observations ($O = \{N^c | c \in C^S\}$), while solution classes and classification rules are provided by the target structure ($C = C^T, R = \{S^c | c \in C^T\}$). Using these definitions, a single information entity can be translated from one context into the other by finding a concept definition $c_i^T$ in the target structure satisfying equation 3.

### 2.3 Support for the Integration Process

The considerations from the last section provide a theoretical foundation for semantic translation. However there are still many problems that have to be solved to put this approach to work. The most important question is how and what kind of context knowledge has to be considered in the translation process because the choice of the representation has major impacts on the classification method to choose and the expected results. Ontologies can play an important role in the translation process because their ability to explicate context knowledge can provide great support. In the following we analyze the roles different ontologies play in our translation approach and describe how they support the whole process of information integration.

**The Role of Ontologies** A closer look at the semantic translation approach described above reveals that different ontologies are used for different purposes within the approach. In order to get clear notions of these different roles we adopt the distinction made in [5]. Jasper and Uschold distinguish three roles an ontology can play in an application scenario, each associated with a level of application:

$L_0$**:** Operational data
$L_1$**:** Ontology
$L_2$**:** Ontology representation language

We will see that each of these roles occur within our framework. Each role is filled by a another kind of ontology with different extends of explication according to the specific requirements.

*Operational Information* that should be translated from one information source to another corresponds to $L_0$. We argued that the real task is to determine the concept an information entity belongs to in a new context. So we rather translate type annotations than the information entity itself. This type information

already is an ontology in the sense of an explicit specification of a conceptualization, because we have to describe the concepts we want to translate. As a consequence we are already concerned with an ontology on the level of operational data. However this ontology does not show a large extend of explication, because it consists of a set of concept terms arranged in a simple taxonomy.

*Specification of Contextual Knowledge* is the basis for the translation of information entities. We use necessary and sufficient conditions for concept membership to specify contextual knowledge. This kind of context explication is a typical application of an ontology. The descriptions of necessary and sufficient conditions therefore is an ontology corresponding to level $L_1$. It shows a larger extend of explication than the pure taxonomy of concept terms, because it explicates the intended meaning of these terms. Each information source to be integrated is supposed to be specified by such an ontology to enable us to use its contextual knowledge in the translation process.

*Properties of Concept* defining necessary and sufficient conditions serve as a common vocabulary used to build the ontologies of different information sources to be integrated. As such they can be seen as an ontology representation language corresponding to level $L_2$. They have to be shared across all information sources to enable a classifier to check whether conditions are fulfilled. They explicate a common understanding of a basic vocabulary that is necessary to explain and exchange specialized vocabulary from different information sources. The extend of explication required from an ontology specifying properties largely depends on the complexity of the information to be translated and requirements on the efficiency of the translation. If complex information has to be translated once more complex property definitions may be used than in the case of simple information that has to be translated in real-time.

**Process and Supporting Technologies** In order to clarify the use of different ontologies we will now discuss the process of intelligent information integration that is implied by our approach. The process sketched below describes actors, supporting tools and knowledge items (i.e. ontologies) involved. Notice that although the approach described above translates only between two sources at a time, it is not limited to bilateral integration, because we do not use a hard-coded translator but a general classifier that will be able to integrate every information source owning a suitable semantic annotation.

*Authoring of Shared Terminology* Our approach relies on the use of a shared terminology in terms of properties used to define different concepts. This shared terminology has to be general enough to be used across all information sources to be integrated but specific enough to make meaningful definitions possible. Therefore the shared terminology will normally be built by an independent domain expert who is familiar with typical tasks and problems in a domain, but who is not concerned with a specific information source. As building a domain

ontology is a challenging task sufficient tool support has to be provided to build that ontology. A growing number of ontology editors exist [1]. The choice of a tool has to be based on the special needs of the domain to be modeled and the knowledge of the expert.

*Annotation of Information Sources* Once a common vocabulary exists, it can be used to annotate different information sources. In this case annotation means that the inherent concept hierarchy of an information source is extracted and each concept is described by necessary and sufficient conditions using the terminology built in step one. The result of this annotation process is an ontology of the information source to be integrated. The annotation will normally be done by the owner of an information source who wants to provide better access to his information. In order to enable the information owner to annotate his information he has to know about the right vocabulary to use. It will be beneficial to provide tool support also for this step. We need an annotation tool with different repositories of vocabularies according to different domains of interest.

*Semantic Translation of Information Entities* The only purpose of the steps described above was to lay a base for the actual translation step. The existence of ontologies for all information sources to be integrated enables the translator to work on these ontologies instead of treating real data. This way of using ontologies as surrogates for information sources has already been investigated in the context of information retrieval [12]. In that paper we showed that the search for interesting information can be enhanced by ontologies. Concerning semantic translation the use of ontologies as surrogates for information sources enables us to restrict the translation on the transformation of type information attached to an information entity by manipulating concept terms indicating the type of the entity. The new concept term describing the type of an information entity in the target information source is determined automatically by a classifier that uses ontologies of source and target structures as classification knowledge. This is possible, because both ontologies are based on the same basic vocabulary that has been built in the first step of the integration approach.

## 3 Building Shared Ontologies - An Overview

The integration process sketched above relies on the existence of a shared ontology suitable to define concepts from all terminologies to be integrated in sufficient detail. This requirement is a challenge with respect to ontology building. In order to support this difficult task, we propose a development methodology that is tailored to the purpose of building shared ontologies. In this section we give an overview of the development process. After describing a concrete integration problem (section 4) we will present a trial run through the methodology in section 5.

### 3.1 The Process

The proposed methodology is based on stepwise-refinement. It consists of five steps executed in sequence resulting in a partial specification of the shared ontology. The last step of each run is an evaluation step that triggers one of the previous steps in order to extend and refine the ontology if necessary.

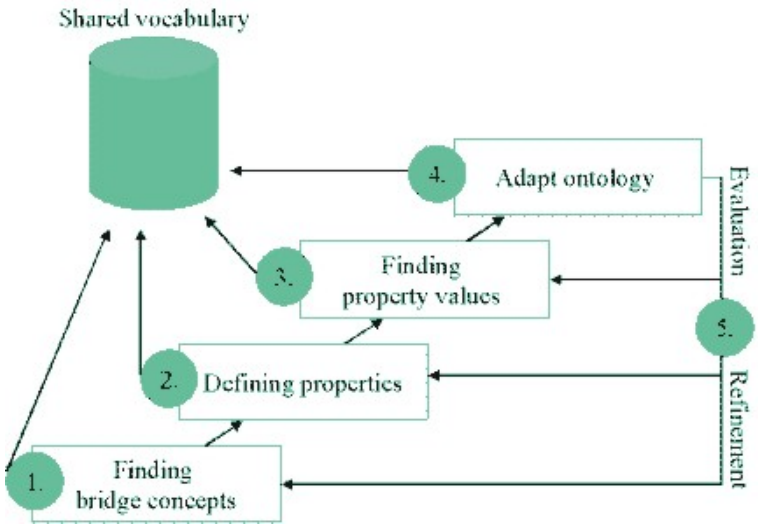


**Fig. 1.** Steps of the Development Process

Figure 1 illustrates the process model, the individual steps are briefly described below.

**Step 1: Finding Bridge Concepts** The first step is to examine the translation task. Asking the question "what do I want to translate?" leads to a concept that subsumes all classes from the source and destination systems. Because this concept makes a semantic translation from one source into another possible we call it bridge concept. While defining its properties and attribute values through the methodology we achieve the needed shared vocabulary. The most general bridge concept is "top", a concept that subsumes every other possible concept. For an exact classification it is recommended to choose the bridge concept as concrete as possible. If needed, more than one bridge concept can be defined to enable semantic translation.

**Step 2: Definition of Properties** The next step is defining properties that describe the chosen bridge concepts. A car, for instance, can be described through its color, its brand, its price, etc.

**Step 3: Finding property values** Once we have defined the properties, we search for values which can fill the attributes. These "fillers" are the main part of the shared vocabulary.

**Step 4: Adapt ontology** The strict usage of the methodology often shows some problems. During the first development cycles, the shared ontology will not be expressive enough to define all terms to be translated. So it may be necessary to adapt the ontology by building a special "support ontology" in order to revise the problem.

**Step 5: Refine Definitions** The introduced methodology follows the "evolving" life cycle. It allows the engineer to step back all the time to modify, add and remove ontology definitions, e.g. refining the bridge concept or integrate further taxonomies into the shared vocabulary.

Each of the steps modifies a different aspect of the shared ontology. While step 1 is concerned with the central concept definition, step 2 defines slots, step 3 integrates existing taxonomies, and step 4 generates application-specific taxonomies. This fact is useful in order to determine where to go back to if the evaluation step reveals the inability to describe a certain aspect of a terminology to be integrated.

### 3.2 Sources of Information

The use of the ontology to be built as a common basis for communication between systems makes it necessary to stay as closely as possible to a vocabulary and conceptualization of the domain that is widely accepted as a standard. In order to meet this requirement, we use several sources of information to build upon. These information sources are existing ontologies and thesauri as well as scientific classifications and data catalogues.

**Upper-Level Ontologies** are mainly used to find the bridge concept which acts as a template for the definition of all terms to be translated. In most cases, the bridge concept is obvious, however, the use of an upper level ontology provides us with a vocabulary which is partly standardized.

**Scientific Classifications** are another form of standards describing the conceptualization of a domain. Classifications like taxonomies of animals or plants are common knowledge which can be used to specify concepts from domain-specific ontologies.

**Domain Thesauri** contain typical terms used in an application domain, therefore they are a natural source for finding concept names for the shared ontology. Further, many thesauri contain at least free-text definitions of the terms included. These definitions provide guidance for the definition of

concepts.

**Linguistic Thesauri** are used to supplement information taken from domain-specific thesauri. In contrast to the specialized vocabulary defined in domain-specific thesauri, linguistic thesauri can be used to identify correspondences between terms found in different information sources. Especially, we use linguistic thesauri to expand the search for definitions of terms to their synonyms.

**Data Catalogues** finally contain the definitions of the terminology to be modeled. Therefore they define the concepts to be modeled and are the basis for evaluating the expressiveness of the shared ontology at a specific point in the modeling process.

In the course of the modeling process, we stick as closely as possible to the information from the sources mentioned above. Therefore the selection of these sources, though not discussed in this paper is already an important step when building a shared ontology.

## 4 An Example Problem

In order to illustrate our methodology for building shared ontologies we use a real-life example from the area of sharing geographic information. We define the integration task to be solved and describe the terminologies that are subject to integration. Both will be the basis for a detailed description of the ontology building process in the next section.

### 4.1 The Task to be Solved

The opening of geographical information systems (GIS) and the interoperability between these systems demands new requirements for the description of the underlying data [10]. GIS normally distinguish different types of spatial objects. Different standards exist specifying these object types. These standards are also called catalogues. Since there is more than one standard, these catalogues compete with each other. To date, no satisfactory solution has been found to integrate these catalogues.

In order to address the semantic translation problem we assume a scenario where the existing land administration database that is normally based on the ATKIS classification should be updated with new information extracted from satellite images of some area. Satellite images are normally analyzed using image processing techniques. These result in a segmentation of different areas which are classified according to the CORINE land-cover nomenclature, a standard for the segmentation and classification of satellite images. The process of updating the land administration system with this new data faces two main problems:

1. The boundaries of the objects in the database might differ from the boundaries determined on the satellite image.
2. The class information attached to areas on the satellite images and the type information in the land administration system do not match.

The first problem is clearly out of the scope of this report, but the second one is a perfect example of a terminology integration problem.

The use of ontologies for the representation of contextual knowledge gives us two options: (a) *integrated views* and (b) *verification*. An integrated view from the users perspective merges the data between the catalogues. This process can be seen as two layers which lay on top of each other. The second option gives user's the opportunity to verify ATKIS data with CORINE land cover data or vice versa.

A query interface – this could be an intelligent dialogue within a GIS system – sends its request to an inference engine. The inference engine builds up the actual knowledge base by using the ontologies of the concepts. The interesting part of the whole idea is that the inference engine can infer on the actual knowledge base and is therefore able to derive new knowledge which can be used for further questions.

### 4.2 The Information Sources

The ATKIS catalogue is an official information system in Germany. It is a project of the head surveying offices of all the German states. The working group offers digital landscape models with different scales from 1:25.000 up to 1:1.000.000 with a detailed documentation in corresponding object catalogues. We use the large scale catalogue ATKIS-OK-1000. This catalogue offers several types of objects including definitions of different types of areas. Figure 2 shows the different types of areas defined in the catalogue.

CORINE land cover is a part of the CORINE Programme the European Commission carried out from 1985 to 1990. The results are essentially of three types, corresponding to the three aims of the Programme: (a) an information system on the state of the environment in the European Community has been created (the CORINE system). It is composed of a series of data bases describing the environment in the European Community, as well as of data bases with background information. (b) Nomenclatures and methodologies were developed for carrying out the programs, which are now used as reference in the areas concerned at the Community level. (c) A systematic effort was made to concert activities with all the bodies involved in the production of environmental information especially at international level. The nomenclature developed in the CORINE Programme can be seen as another catalogue, because it also defines a taxonomy of area types (see figure 3) with a description of characteristic

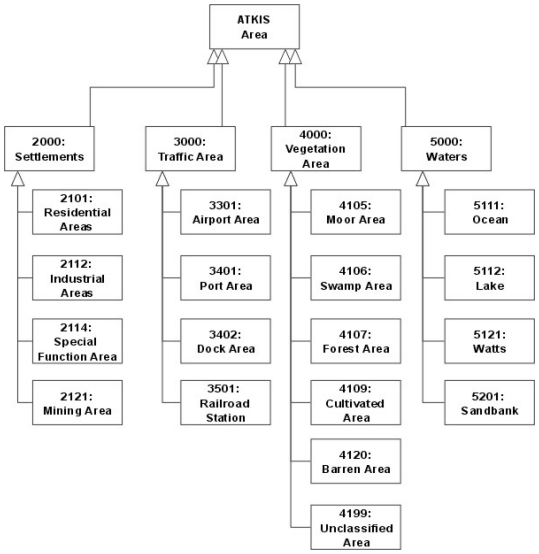


**Fig. 2.** Taxonomy of land-use types in the ATKIS-OK-1000 catalogue

properties of the different land types.

The taxonomies of land-use types in figures 2 and 3 illustrate the context problem mentioned in the introduction. The set of land types chosen for these catalogues are biased by their intended use: while the ATKIS catalogue is used to administrate human activities and their impact on land use in terms of buildings and other installations, the focus of the CORINE catalogues is on the state of the environment in terms of vegetation forms. Consequently, the ATKIS catalogue contains fine-grained distinctions between different types of areas used for human activities (i.e. different types of areas used for traffic and transportation) while natural areas are only distinguished very roughly. The CORINE taxonomy on the one hand contains many different kinds of natural areas (i.e. different types of cultivated areas) which are not further distinguished in the ATKIS catalogue. On the other hand, areas used for commerce and traffic are summarized in one type.

## 5 Building Terminologies - An Example

In this section we will show how a shared terminology and concept descriptions for the example problem can be developed using the methodology briefly described above.

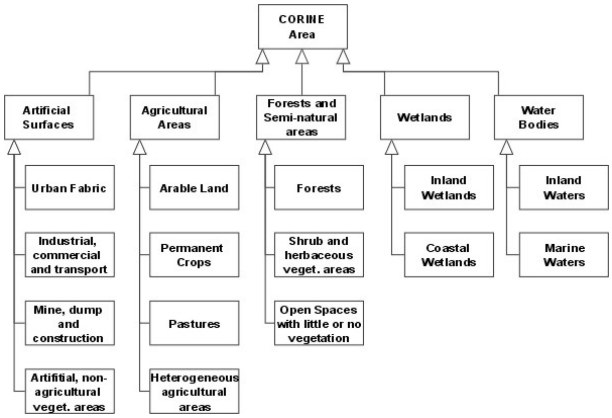


**Fig. 3.** A part of the taxonomy of land-use types in the CORINE land cover nomenclature

### 5.1 Information Sources

For this specific integration task we chose several sources of information to be used for guiding the development process. We briefly describe these sources in the following.

**UpperCyc Ontology** The UpperCyc, developed by Doug Lenat, CyCorp Inc. (http://www.cyc.com), is an upper-level ontology that captures approximately 3,000 terms of the most general concepts of human consensus reality. There is also a full Cyc knowledge base (KB) including a vast structure of more specific concepts descending below the UpperCyc, the so called top-level ontology. It contains millions of logical axioms – rules and other assertions – which specify constraints on the individual objects and classes found in the real world. Therefore the Upper Cyc ontology provides a sufficient common grounding for applications. It is possible to search the UpperCyc via internet: http://www.cyc.com/cyc-2-1/find-constant.html.

**GEMET** For the given scenario we choose the "GEneral Multilingual Environmental Thesaurus (GEMET)", a polyhierarchically structured thesaurus which covers approximately 5.400 terms and their definitions organized by groups, themes, and terms. GEMET has been created by merging different national and international thesauri. Analysis and evaluation work of numerous international experts and organizations let to a core terminology of generalized environmental terms and definitions. GEMET ensures validated indexing and cataloguing of environmental information all over Europe. Where available, synonyms or alternate terms can be found likewise. The visualization tool for the GEMET is ThesShow, supporting the navigation through the GEMET database. It features, amongst others, extensive search, retrieval, and indexing functions.

**Wordnet** WordNet, developed by the Cognitive Science Laboratory at Princeton University under the direction of Professor George A. Miller, is an on-line lexical reference system whose design is inspired by current psycholinguistic theories of human lexical memory. English nouns, verbs, adjectives, and adverbs are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets. You can query the WordNet through an online HTML form at http://www.cogsci.princeton.edu/cgi-bin/webwn.

**Standard Taxonomies** Scientific taxonomies can be found in many sources, like books or the internet. For this example we looked into the Google Web-directory (http://directory.google.com/Top/Science/Biology/Flora_and_Fauna) to obtain a classification of plant life. It is in no circumstances complete, but it satisfies our needs in this walkthrough.

### 5.2 Example Walkthrough

Based on the information described above we built up a first version of a shared ontology which should be used to solve the integration task mentioned in the last section. In this section we sketch the first development cycle of this ontology using the concrete modeling activities to illustrate the different steps of our methodology.

**Step 1: Finding Bridge Concepts** Looking at the given example scenario as described in section 4 it is quite obvious to choose a concept like "area" or "region", because all land-use classes are some kind of special "regions", or in other words, "region" subsums all land-use classes. We search for the term "region" in the "Upper-CYC" and get the following definition:

> *GeographicalRegion: A collection of spatial regions that include some piece of the surface of PlanetEarth. Each element of GeographicalRegion is a PartiallyTangible entity that may be represented on a map of the Earth. This includes both purely topographical regions like mountains and underwater spaces, and those defined by demographics, e.g., countries and cities* [· · · ]"

Figure 4 shows the hierarchical classification of the concept in the Upper-CYC. The definition fits very well, so finally we choose "Geographical Region" as our bridge concept. For further refinement we write it down in the OIL notation [2].

```
Class-def Geographical-Region
```

**Step 2: Definition of Properties** Now we have to find possible attributes for the bridge concept. We look for "Geographical Region" in the GEMET, but the search does not give any results. In that case the decomposition of the search phrase may give better results. For "Geography" and "Region" we get these definitions out of GEMET:

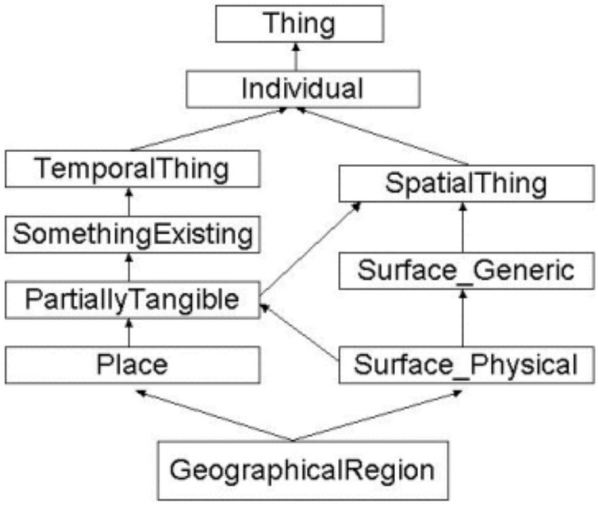


**Fig. 4.** Geographical Region in the Upper-CYC

*Geography: "The study of the natural features of the earth's surface, comprising topography, climate, soil, vegetation, etc. and man's response to them."*

*Region: "A designated area or an administrative division of a city, county or larger geographical territory that is formulated according to some biological, political, economic or demographic criteria."*

Here are some attributes clearly recognizable. For example "vegetation" is a biological criterion that defines a region, and it is also part of the scientific field geography. We update the bridge concept by defining a slot "vegetation" and adding it to the bridge concept.

```
Slot-def vegetation
    Domain Geographical-Region

Class-def Geographical-Region
```

**Step 3: Integration of Standard Taxonomies** To get possible "attribute values" or "filler" for the slot "vegetation", we take another look into GEMET. Vegetation is defined as:

*"The plants of an area considered in general or as communities* $[\cdots]$*; the total plant cover in a particular area or on the Earth as a whole."*

We also check the synonym "flora", found in WordNet:

*"The plant life characterizing a specific geographic region or environment."*

The attribute "vegetation", respectively "flora", can be filled with terms out of plant life like "tree" or "rose" for instance. A good top concept is "plants", because many scientific taxonomies of plants exists. The Swedish botanist Carlous Linaeus established 1753 a classification of plants. His work is considered the foundation of modern botanical nomenclature. In the Google Webdirectory we can access the plant kingdom with more than 10.000 entries online. We integrate this taxonomy into our vocabulary.

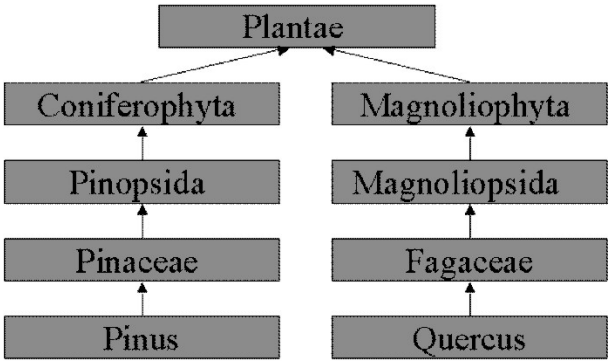


**Fig. 5.** Extract from scientific plant taxonomy

Now it is possible to describe classes from the land-use catalogues. The term "coniferous forest" in the CORINE context is defined as:

*"Vegetation formation composed principally of trees, including shrub and bush understories, where coniferous species predominate."*

In our vocabulary we find the term "Coniferophyta", comprising the conifers, which are trees or shrubs that bear their seeds in cones, without the protection of a fruit like angiosperms. This leads to the following OIL class:

```
class-def  Coniferous_Forest
  subclass-of  Geographical-Region
  slot-constraint vegetation value-type Coniferophyta
```

The division Magnoliophyta of the plant kingdom consists of those organisms commonly called the flowering plants or angiosperms. The flowering plants are the source of all agricultural crops, cereal grains and grasses, garden and roadside weeds, familiar broad-leaved shrubs and trees, and most ornamentals. So, it is easy to describe the next CORINE class "broad leaved forest":

```
class-def  Broad-leaved_Forest
  subclass-of  Geographical-Region
  slot-constraint vegetation value-type Magnoliophyta
```

A "mixed forest" in the CORINE nomenclature consists of conifers and broad-leaved trees.

```
class-def  Mixed_Forest
  subclass-of  Geographical-Region
  slot-constraint vegetation has-value Magnoliophyta
  slot-constraint vegetation has-value Coniferophyta
```

**Step 4: Adapt vocabulary** A closer look at the definition of the CORINE forest classes reveals that the classes are defined through the existence of trees and shrubs. Just using the term "Magnoliophyta" does not prevent the classification of a region covered with orchids as a broad-leaved forest (Orchidaceae is a subclass of Magnoliophyta). The mentioned taxonomy classifies plants by propagation, so there exists angiosperm and gymnosperm trees, shrubs and flowers. To handle this problem we need a more general distinction.

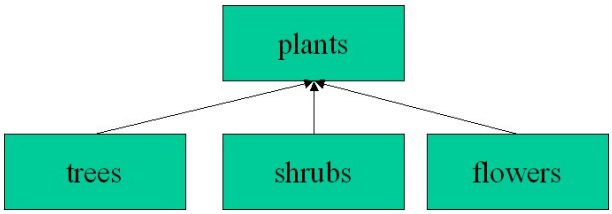


**Fig. 6.** Supplementary Plant Classification

Figure 6 shows a simple extension of the vocabulary that enables a more robust definition of the CORINE forest classes.

```
class-def  Coniferous_Forest
  subclass-of  Geographical-Region
  slot-constraint vegetation value-type Coniferophyta and (trees or shrubs)

class-def  Broad-leaved_Forest
  subclass-of  Geographical-Region
  slot-constraint vegetation value-type Magnoliophyta and (trees or shrubs)

class-def  Mixed_Forest
  subclass-of  Geographical-Region
  slot-constraint vegetation
```

```
    has-value Coniferophyta and (trees or shrubs)
    has-value Magnoliophyta and (trees or shrubs)
```

The shared vocabulary developed so far allows us to specify many different vegetation areas found in the land-use catalogues:

```
class-def  Pastures
  subclass-of  Geographical-Region
  slot-constraint vegetation value-type Poaceae

class-def  vineyards
  subclass-of  Geographical-Region
  slot-constraint vegetation value-type Vitis

class-def  Rice_fields
  subclass-of  Geographical-Region
  slot-constraint vegetation value-type Oryza
```

**Step 5: Evaluation and Revision** Not all CORINE landcover classes can be described after this first process cycle. "Mineral extraction sites", for instance, are defined as: "Areas with open-pit extraction of minerals (sandpits, quarries) or other minerals (opencast mines). Includes flooded gravel pits, except for river-bed extraction." No vegetation is mentioned, so the bridge concept must be refined. We go back to step 2 "defining properties" and search for another attribute. The definitions of "region" and "geography" show some anthropological aspects, like "man's response" or economic criteria. So we define a new slot "anthroposphere" and add it to our bridge concept:

```
slot-def anthroposphere
    Domain geographical-Region

slot-def vegetation
    Domain geographical-Region

class-def Geographical-Region
```

In GEMET exists the group "anthroposphere". One of its subclasses is "mining district", a district where mineral exploitation is performed. We integrate the partial taxonomy into the vocabulary (figure 4).

This special vocabulary can be used to simulate one-to-one mappings by using equality axioms. The CORINE class "mineral extraction sites" could be described as followed.

```
class-def  Mineral-extraction-sites
  subclass-of  Geographical-Region
  slot-constraint anthroposphere has-value mining-district
```

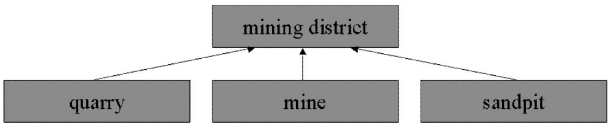


**Fig. 7.** Mining sites from the GEMET thesaurus

In a similar way we proceed iterating the process cycle until all terms from the two catalogue systems can be modeled as a specialization of the bridge concept. A further advantage of this strategy is the fact that the same process will be employed when additional terminologies are to be integrated as well. We cannot guarantee that the shared ontology also covers a new terminology, but we already provide guidance for the adaption of the ontology.

## 6 Discussion

We presented a method for building shared ontologies that can be used for terminology integration. The method was developed to support an approach to terminology integration by finding semantic correspondences we proposed in previous work [9, 7]. We first reviewed the integration approach and the role of shared ontologies in the approach. After giving an overview of the method, we introduced a real life problem and showed how the method can be used to successively develop a shared ontology for the problem. The example already shows that the method leads to better results than for example the hands-on approach described in [8]. However, there are still a number of open questions.

At the moment we only consider the case that the modifications to the ontology applied in each process cycle result in a strict refinement of the previous model. In our experiments this was always the case, however, in general it might be necessary to take back modeling decisions and revise parts of the ontology. In this case, we need adequate strategies for change management and versioning in order to avoid a loss of information. If parts of a previous model are removed, we have to track the impact on our possibility to define all terms that are subject to integration and we have to find ways to handle conflicts.

A second point concerns the evaluation of the results. At the moment we evaluate the ontology on the basis of its ability to be used to define concepts that represent terms from different terminologies to be integrated. As our goal is an automatic translation of terms, a decent evaluation should include translation trials. The use of a subsumption reasoner to prove semantic correspondences requires the model to be as complete as possible, a property that we do not check at the moment. Further, a measure is required that indicates, whether the model improves in the course of the process. Both topics, ontology revision and evaluation strategy will be addressed in future work.

## References


1. A. Duineveld, R. Studer, M. Weiden, B. Kenepa, and R. Benjamis. A comparative study of ontological engineering tools, 1999.
2. D. Fensel, I. Horrocks, F. Van Harmelen, S. Decker, M. Erdmann, and M. Klein. Oil in a nutshell. In *12th International Conference on Knowledge Engineering and Knowledge Management EKAW 2000*, Juan-les-Pins, France, 2000.
3. M. Fern'andez, A. G'omez-P'erez, and N. Juristo. Methontology: From ontological art towards ontological engineering, 1997.
4. M. Gruninger and M. Fox. Methodology for the design and evaluation of ontologies, 1995.
5. R. Jasper and M. Uschold. A framework for understanding and classifying ontology applications. In *12th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*. University of Calgary/Stanford University, 1999.
6. M. Stefik. *Introduction to Knowledge Systems*. Morgan Kaufmann Publishers, San Francisco (CA), USA, 1995.
7. H. Stuckenschmidt. Using oil for intelligent information integration. In *Proceedings of the Workshop on Applications of Ontologies and Problem-Solving Methods at ECAI 2000*, 2000.
8. H. Stuckenschmidt, F. van Harmelen, D. Fensel, M. Klein, and I. Horrocks. Catalogue integration: A case study in ontology-based semantic translation. Technical Report IR-474, Computer Science Department, Vrije Universiteit Amsterdam, 2000.
9. H. Stuckenschmidt and U. Visser. Semantic translation based on approximate reclassification. In *Workshop on Semantic Approximation, Granularity and Vagueness*, Breckenridge, Colorado, 2000.
10. H. Stuckenschmidt, U. Visser, G. Schuster, and Th. Voegele. Ontologies for geographic information integration. In *Intelligent Methods in Environmental Protection: Special Aspects of Processing in Space and Time.* Center for Computing Technologies, University of Bremen, 1999.
11. M. Uschold. Building ontologies: Towards a unified methodology. In *16th Annual Conf. of the British Computer Society Specialist Group on Expert Systems*, Cambridge, UK, 1996.
12. T. Voegele, H. Stuckenschmidt, and U. Visser. Towards intelligent brokering of geo-information. In *Proceedings of the Urban Data Management Symposium*, Delft, 2000.