Iterative Ontology Modelling for mapping land use and land cover catalogues

Peter Mandl and Mark Hall

Department of Geography and Regional Studies, University of Klagenfurt
Universitätsstrasse 65 - 67, 9020 Klagenfurt, Austria
peter.mandl@uni-klu.ac.at, mhall@edu.uni-klu.ac.at

Abstract. Spatial Data Infrastructures need harmonised thematic information. In the HarmonISA project an ontology based procedure for comparing different land use and land cover type catalogues was developed and used to harmonise cross border land use maps. This paper introduces an iterative ontology modelling approach for the knowledge engineering process. A modular ontology structure is proposed based on a common vocabulary for land use and land cover catalogues. Applications built on this ontology structures are presented including a web based user interface for the display of harmonised ontologies and maps.

1 Introduction

Spatial Data Infrastructures can only work in an interoperable environment, which means harmonised data formats, cartographic projections, data geometries, attribute formats and actuality as well as harmonised thematic information. Because of the many problems concerning these features and the additional problem of different languages in three European countries, the *ISAMAP Project*[1] which aims at the harmonisation of regional data resources for cross-border planning in Friuli - Venezia Giulia (Italy), Slovenia and Carinthia (Austria) was started two years ago. Much cooperation in the GI-field has been done in the Alpine-Adriatic Region since many years but now the EU co-financed ISAMAP project will lay the foundation for a cross border Spatial Data Infrastructure in that region.

The ISAMAP subproject *HarmonISA*, realised at the Department of Geography and Regional Studies at the Alpine-Adriatic-University Klagenfurt, contributes to the thematic aspect of data harmonisation namely *semantic interoperability* for the specific application domain of *land use and land cover* (LUC). In that context semantic heterogeneity exists for instance when one conceptive LUC type (e.g. overbuilt areas) has different names in different application contexts (e.g. urban or built up or man made areas). Another example is when the same LUC type name (e.g. forest) may signify different thematic classes or concepts (e.g. private and public, age, density or kind of trees). Some class names cannot

[1] INTERREG IIIB CADSES framework, http://www.isamap.info

be translated to other languages without changing also its semantics (e.g. forest, Wald, bosco). A final problem arises when geodata from different countries should be unified into one border crossing data set and when one or the other LUC type is missing in one of the data sets (e.g. land use classes like tourist areas in land cover data sets). To overcome these problems HarmonISA provides a *Web Service* to produce harmonised land use and land cover catalogues for the three ISAMAP countries and a *Web Map Service* to produce cross border LUC maps for the entire area.

This paper presents an approach to iteratively creating ontologies of land use and land cover catalogues. Existing methodologies are improved by combining a very strongly iterative development process with a modularised ontology modelling process. This makes it possible to verify at each development step that the formalised knowledge is correct. Additionally the resulting ontologies are highly modular and thus can be integrated into existing systems more easily.

In chapter 2 existing foundational work is given, chapter 3 discusses basic ontology modelling and chapter 4 presents the iterative ontology modelling approach. Finally conclusions are drawn and future work is proposed.

2 Ontologies in GI-Science as the conceptual background of the HarmonISA project

To provide a Web Service for harmonised land use and land cover catalogues and maps for three countries not only data naming and showing the LUC classes and information about the relations between these classes have to be considered but also *knowledge* about higher concepts of the domain of LUC, like hierarchies, heterogeneities, similarities or inconsistencies is necessary. This knowledge has to be described in computer readable form and used during the interactive harmonisation process of the Web Service. Many articles in the field of GI-Science have discussed *formal ontologies* to serve as a knowledge representation model for different purposes (see [13]).

Barry Smith and David Mark [12] discussed the basic concepts necessary in an "Ontology of Geographic Kinds". Andrew Frank presented, in one of his many works about ontologies, several tiers of ontology (human-independent reality, observation of physical world, objects with properties, social reality and subjective knowledge) which should be considered in an overall ontology for GI-Systems [5]. Karen Kemp and Andrej Vckovsky [7] designed the concepts for an ontology of fields, as one of the two basic data model groups in GI-Systems.

Andrea Rodriguez and Max Egenhofer [11] developed a special similarity measure for comparing geospatial entity classes and Isabel Cruz et al. [3] developed a semi-automatic procedure for an ontology alignment to query LUC geospatial data. In other papers ontologies for specific application domains are designed like in the paper of F.T. Fonseca et al. [4] where the knowledge representation for urban information systems is presented.

Last but not least the ideas of semantic reference systems of Werner Kuhn [8] and co-workers in the MUSIL[2] should be mentioned to give the semantic aspects of geospatial data and services a kind of common framework.

Considering this background the key aims of the HarmonISA project are:

1. to create a *common vocabulary* (Protégé: classes) for different land use and land cover type catalogues, made for different purposes (application contexts), using different data material, described in different languages, concerning different countries, based on different scales and resolutions, classification methods, background of the interpreters etc.,
2. to create a "*Skeleton Ontology*" using the common vocabulary and common attributes (Protégé: properties) of all LUC classes,
3. to find *similarity values* between classes from different LUC catalogues,
4. to design a *user interface* for selecting the target catalogue, a selection of wanted classes, display filter criteria and the similarity measure values for each mapping,
5. to *display the harmonised classes in a map* using the colour key of the target catalogue.

The vocabularies of the first aim are given by the used LUC catalogues, which are CORINE for Slovenia[3], MOLAND for Italy[4] and the "Realraumanalyse" für Austria[5]. The methods used for achieving the second aim are described in the next chapters of that paper. A protype of the HarmonISA Web Application can be found under `http://harmonisa.uni-klu.ac.at`.

3 Modelling Ontologies

3.1 Ontology basics

An ontology is a shared specification of a conceptualisation of domain knowledge [6] and is used to define knowledge in such a way that the computer can "understand" the concepts and their relationships. This makes it possible to create more "intelligent" systems that provide a richer experience to the user. There is a wide selection of ontology languages to choose from and in the HarmonISA project we chose to use OWL [9] as ontology language. The reasons were that we were interested in a standards based solution and also that OWL provides a sublanguage that corresponds to the field of description logics [1]. This correspondence makes it possible to use description logics reasoners for automated reasoning on the ontology. Automated reasoning can be used to extract the implicit knowledge from an ontology.

In the project automated reasoning was used to extract the implicit hierarchy among the ontology classes that were defined. To facilitate automated reasoning

[2] Muenster Semantic Interoperability Lab (`http://musil.uni-muenster.de`)
[3] `http://terrestrial.eionet.ei.int/CLC2000`
[4] `http://moland.jrc.it/`
[5] `http://www.uni-klu.ac.at/geo/projekte/realraum/realraum.htm`

it is necessary to have the ontology in a form that maximises the amount of explicit knowledge and does so in a way that leaves the ontology flexible so that the automated reasoner can restructure it. One such approach is ontology modularisation.

3.2 Ontology modularisation

Ontology modularisation is a method of structuring ontologies so as to maximise their expressiveness and integrability. To create modularised ontologies we use a *normalisation approach* as proposed by Alan Rector [10]. In this approach each ontology is decomposed into disjoint skeleton taxonomies, which are restricted to simple trees based on the subsumption principle. These skeleton taxonomies can then be recombined using definitions and restrictions, recreating all knowledge that was represented in the original non-normalised ontology. The advantages of this approach are

- automated reasoning can easily be performed on the ontology,
- the resulting ontology can be evolved and maintained with less effort,
- the amount of knowledge that remains implicit in the class names is reduced to a minimum,
- the resulting ontology can be merged or integrated with other ontologies more easily.

The first two advantages are the main reason that the modular ontology approach was used. Automated reasoning and continuous ontology evolution form the basis for our iterative ontology modelling approach.

Figure 1 shows a non-normalised ontology describing trees and forests. The problem with this structure is that although it is a valid representation of the knowledge, it is not cleanly defined. The ontology structure does not always branch according to the same principle. In some cases the branching condition is an *is-a* relationship (*Tree* - *Broad-leafed*) in others it is *is-in* (*Tree* - *Forest*). These differences are immediately obvious to a human, but since they are implicit within the names of the classes the reasoner cannot understand them.

Figure 2 shows the same knowledge but now in a normalised form. The explicitly defined concepts have been reduced to a simple tree based on subsumption (*is-a*). A new property *vegetation* has been added and the remaining classes have been defined based on this property. Now that most of the knowledge has been made explicit we can use an automated reasoner to infer the hierarchy that exists between the classes. Thus all of the knowledge in the non-normalised ontology is preserved while the amount of implicit knowledge is reduced by making it explicit.

4 Iterative Ontology Modelling

In our project we combined the modular approach described in the previous section with an *iterative approach* to creating the actual ontologies. History has

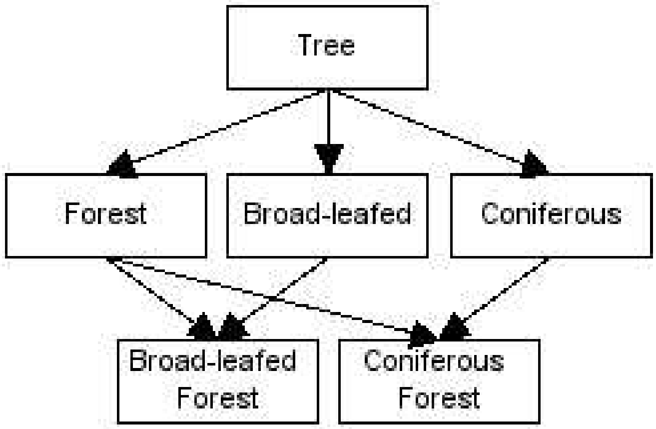

Fig. 1. Non-normalised ontology

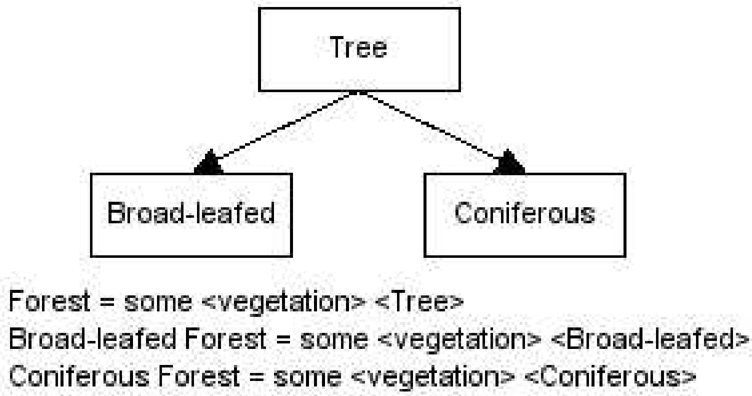

Fig. 2. Normalised ontology

shown that the waterfall approaches to creating large systems tends not to work [2]. This also holds for knowledge engineering and ontology modelling. It is therefore better to use an iterative development methodology. Also iterative ontology development meshes well with modularised ontologies. We used an approach that can be basically split into two parts

1. **Initial development**. In this phase a basic foundation is created from which the complete ontologies then evolved.
2. **Evolution**. Then the ontology is slowly completed by iteratively adding small parts.

Even though this iterative approach discourages creating elaborate solutions to solve future requirements it is nevertheless necessary to decide on a general framework before actually beginning to model the ontologies. For the project this was the *skeleton ontology* which contains the shared vocabulary and the properties. Additionally for each LUC catalogue that to be formalised create an ontology that imports the skeleton ontology and uses the properties to define the individual LUC categories.

4.1 Initial development

In the initial development phase we focused on the forest classes of the Corine LUC catalogue. We analysed the categories and extracted the basic properties that defined these categories. We then created an ontology containing the skeleton tree that contains the subtrees defining the ranges of the basic properties. Importing this skeleton ontology into the CORINE ontology we started creating the necessary classes for the forests categories (part of the result from figure 2).

After we had completed this phase we used the results to test the rest of our application. Since these tests completed without any major problems we moved on to the evolution phase.

4.2 Evolution

The evolution phase forms the backbone of our iterative ontology modelling approach. If taken to the extreme the initial development can be reduced to creating the files and then the complete knowledge engineering process happens in the evolution phase.

In the evolution phase we iteratively added classes and properties to the skeleton ontology and also classes to the CORINE ontology. In some cases only one LUC category was added in one iteration cycle sometimes more. For each iteration cycle the following process was followed

1. **Determine changes**. The first step is to determine which LUC categories to add. How many LUC categories to add in one iteration depends on an estimate of how complicated the change is. This estimate is based on how fast seemingly similar categories were added in the past.

2. **Change analysis**. Analyse which properties define the categories and what values these properties have. This list of changes are the requirements for the next step.
3. **Update skeleton**. Based on the analysis update the skeleton ontology adding classes and properties as necessary and possibly restructure the class hierarchy to better reflect the knowledge. After this step the LUC ontologies might be inconsistent.
4. **Create LUC classes**. Create the new classes in the LUC ontology describing the LUC categories to add.
5. **Propagate changes**. Propagate the changes from the skeleton ontology. This step guarantees that the knowledge represented in the LUC ontologies regains its consistency and correctness.
6. **Verification**. The final step is verification that the applied changes actually implement what was desired.

These six steps are repeated until the complete knowledge of each LUC catalogue has been formalised. This then forms the first version of the LUC ontologies. To retain the level of correctness the ontologies have to be kept up to date with the actual LUC catalogues. This can also be achieved following the above algorithm.

4.3 Possible change propagation scenarios

When changing the skeleton ontology it is very likely that the LUC ontologies become inconsistent, incorrect or imprecise. Therefore it is necessary that the changes in the skeleton ontology are propagated into the LUC ontologies. This guarantees that the LUC ontologies remain consistant, correct and that the defined classes are defined as precisely as possible. There are three major change scenarios

- **New properties**. The addition of new properties requires that all existing classes in the LUC ontologies have to be checked to see if the new properties also apply to any of the existing data. This is necessary because of the minimal approach we use. Although the property might have been found earlier on it was not added because it did not add any differentiating information at that time. Now that it does add differentiating information the property needs to be added to all classes that are at least partially defined by it.
- **New classes**. Newly added classes in the skeleton ontology can also possibly require changes in the LUC ontologies. Similar to the case with new properties existing LUC ontology classes might have previously used classes that are now higher up in the skeleton ontology hierarchy. The new classes might more precisely describe the attributes of the LUC category. In the majority of cases it is sufficient to check those LUC ontology classes that use the parent and sibling classes of the newly added class.
- **Restructured classes**. While adding properties and classes can only lead to imperfectly represented knowledge it cannot lead to inconsistencies or wrong

knowledge. Restructuring the existing classes in the skeleton ontology can create both inconsistencies and wrong knowledge. Inconsistencies arise when classes that were previously unrelated through the restructuration process have become disjoint or ins some other way conflicting. This means that a previously valid definition might now be inconsistent, because it is no longer satisfiable. This kind of error can be caught by the automated reasoning system. Wrong knowledge errors are much harder to find because the ontology remains consistent just the knowledge no longer represents the actual world. In this case all LUC ontology classes must be checked that use the property or classes for which the skeleton was restructured.

Checking these three main change propagation scenarios after each change to the skeleton ontology is a large amount of work and one of the reasons why the number of changes in each iteration should be kept as low as possible. Otherwise changes resulting in incorrect knowledge being represented can be overlooked easily.

In addition to the reviews performed during and after the change propagation it is usefull to perform external reviews of the generated ontologies. Due to the iterative nature of the development these can be performed at any time during the ontology modelling process.

5 Conclusions and future work

The HarmonISA project is in a state where the included ontologies have a satisfying stable structure, which means, that experts are quite content with the mapping of the different LUC catalogues. The border crossing LUC Maps, produced by the HarmonISA Web Application can be used a good base for regional planning studies. Nevertheless the approach for semantic harmonisation of thematic data, information and knowledge will be extended in future projects.

5.1 Spatial properties

One question we have so far not approached is whether there is a qualitative difference between spatial and non-spatial properties. On the slightly more abstract level that we have chosen to work on they are equal because they are both properties that a LUC category can have. For the requirements of our work this is sufficient.

On side effect of this is that certain relationships cannot be modeled. For example assume a spatial property *is-next-to* and what we want to model is a forest next to a river. Our approach forces us to create a class *AdjacentObject* in the skeleton ontology which forms the root of the tree of possible values. So for the forrest alongside the river you would specify that *is-next-to AdjacentRiver*. This in a way cleanly represents the knowledge.

What can not be represented is that the *Forest* is actual next to a *River* where both classes are part of the LUC ontology and not the skeleton ontology.

To model this the range of the property would have to include classes from the LUC ontology. Although this represents the actual real-world semantics more correctly it blurs the distinction between the ontology skeleton and the LUC ontologies. This influences the ability to perform automated reasoning on the ontologies and also makes it harder to evolve the ontologies.

5.2 Property hierarchy and other extensions

Currently all properties are seen as equal and as contributing the same amount of information towards the differentiation of the LUC categories. In future it might be interesting to see if the properties form a hierarchy. This is especially interesting in the context of the similarity measurements.

Other LUC catalogues for other regions and with different thematic and also spatial resolution well be included next. This will increase the skeleton ontology although practical work shows that most LUC catalogues consist of similar LUC categories. Maybe this could be different, when we consider catalogues, which where designed for different application contexts (planning, agriculture, tourism etc.).

The similarity measures which were used for comparing the LUC classes in the HarmonISA Web Application will be extended and empirically evaluated.

The Web Map Output will be extended to including more functionality like numerical statistics, different legend colour scales and the overlay of additional data layers for orientation. In the next future the resulting maps should be accessible in the ISAMAP Spatial Data Infrastructure[6].

6 Acknowledgements

We are grateful to the Lead Partner of the ISAMAP Project, the Office of the Government of Carinthia, Dept. 20, Spatial Planning - KAGIS, represented by Mag. Klaus Gruber (contract nr. 20-Reg-1025/623-04) and the University Klagenfurt that financed the HarmonISA Project. We thank the team of MUSIL, especially Werner Kuhn, Martin Raubal and Michael Lutz and the partners in the ISAMAP project for the very fruitful discussions.

References

1. Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, January 2003.
2. Frederick P. Brooks. *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley, 1975.
3. Isabel F. Cruz, William Sunna, and Anjli Chaudhry. Semi-automatic ontology alignment for geospatial data integration. In M.J. Egenhofer, C. Freksa, and H.J. Miller, editors, *GIScience 2004*, LNCS 3234, pages 51 – 66. Springer-Verlag Berlin Heidelberg, 2004.

[6] `http://www.isamap.info/html/isa-map_viewer.html`

4. F.T. Fonseca, M.J. Egenhofer, C.A. Davis Jr., and K.A.V. Borges. Ontologies and knowledge sharing in urban gis. *Computers, Environment and Urban Systems*, 24:251 – 271, 2000.
5. Andrew U. Frank. Tiers of ontology and consistency constraints in geographical information systems. *International Journal on Geographical Information Science*, 15(7):667 – 678, 2001.
6. T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, Vol. 5(No. 2), 1993.
7. Karen K. Kemp and Andrej Vckovsky. Towards an ontology of fields. In *Proceedings of the 3rd International Conference on GeoComputation*, September 1998.
8. Werner Kuhn. Semantic reference systems. *International Journal of Geographical Information Science*, 17(5):405 – 409, 2003.
9. P. F. Patel-Schneider, P. Hayes, and I. Horrocks. Owl web ontology language semantics and abstract syntax. W3C Working Draft 31 March 2003, March 2003.
10. Alan L. Rector. Modularisation of domain ontologies implemented in description logics and related formalisms including owl. In *Proceedings of Knowledge Capture 2003*, pages 121–128. ACM, 2003.
11. Andrea Rodríguez and Max Egenhofer. Comparing geospatial entity classes: An asymmetric and context-dependent similarity measure. *International Journal of Geographical Information Science*, Vol. 18(3):229 – 256, 2004.
12. Barry Smith and David M. Mark. Ontology and geographic kinds. In *Proceedings, International Symposium on Spatial Data Handling (SDH'98)*, pages 308 – 320, July 1998.
13. Stephan Winter. Ontology: buzzword or paradigm shift in gi science. *International Journal of Geographical Information Science*, 15(7):587 – 590, 2001.