Contributions to the problem of knowledge management in Spatial Data Infrastructures

Javier Lacasta Miguel

PhD DISSERTATION

RESEARCH ADVISORS
Javier Nogueras Iso
Francisco Javier Zarazaga Soria

Computer Science and Systems Engineering Department
University of Zaragoza
María de Luna, 1 E-50018 Zaragoza, Spain

March, 2009

To my family and friends

The important thing is not to stop questioning
Albert Einstein 1879 - 1955

Acknowledgements

There are many people to whom I am grateful for their support during all these years of hard work. So I hope not to forget anybody. First of all, I want to thank to my research advisors Javier Nogueras and Francisco Javier Zarazaga for their guidance and support along the years that have taken to complete this thesis work. Additionally, I also want to thank to Pedro R. Muro and Francisco Javier Lopez by their comments and suggestions that have been so valuable for improving my work. Many thanks have to be given to the members and ex-members of IAAA group of the University of Zaragoza that have provided me with the required technical support and advice along all this years, specially to Juanjo, Jesus, Mariano ,Rodolfo, Covadonga, Christian, Aneta, and Miguel Angel. I also want to thank to Douglas Tudhope by his support along my research stay at the Hypermedia Research Unit of the University of Glamorgan. Additionally, I would like to mention the support given by the research network COST ACTION C21 - Towntology (Urban ontologies for an improved communication in urban civil engineering projects) for the development of experiments in the urban domain. From a personal perspective, I want to thank to my friends in Zaragoza and Wales whom I have shared so many good moments, and that have helped me to keep on forward. Finally, and over all of the previously mentioned ones, I want to thank to my all family that has been supporting me all these years.

Zaragoza, March 2009

Javier Lacasta Miguel

Resumen

Las infraestructuras de información son soluciones integradas basadas en la fusión de información y tecnologías de comunicación. Las infraestructuras de datos espaciales son infraestructuras de información especializadas en la gestión de información geográfica. Se caracterizan por la gran cantidad de datos que tienen que gestionar, por el alto coste de obtención de dichos datos, y por el creciente número de aplicaciones de dichas infraestructuras.

Una infraestructura de información requiere de un sistema de recuperación eficiente y efectivo para proporcionar a los usuarios el acceso a los elementos almacenados en la infraestructura. Sin embargo, los términos de lenguaje natural usados en clasificación, indexado y consulta contienen relaciones semánticas que pueden dificultar la creación de sistemas de descubrimiento de la información efectivos. En este contexto, las ontologías son frecuentemente usadas para mejorar el rendimiento de los sistemas de descubrimiento de información. De entre los tipos de ontologías existentes, las ontologías terminológicas son las más frecuentemente usadas para clasificación y recuperación de información. Sin embargo, la heterogeneidad de estos modelos dificulta su integración.

Esta tesis doctoral analiza los principales problemas que afectan a los sistemas de descubrimiento de las infraestructuras de información en relación con la gestión de ontológicas terminológicas. Primeramente, se centra en los problemas de representación de los modelos terminológicos donde un marco de representación es requerido. Seguidamente, analiza las necesidades de adquisición de ontológicas terminológicas. Especialmente, en la necesidad de crear, relacionar, e integrar los modelos requeridos por una infraestructura. En tercer lugar, profundiza en el problema de acceso a los modelos terminológicos vía servicios y componentes interoperables.

Como respuesta a dichos problemas, esta tesis propone un marco de representación común para ontológicas terminológicas, una serie de métodos y técnicas que facilitan la adquisición e integración de modelos terminológicos, y un conjunto de diseños arquitecturales para facilitar la

gestión y el acceso a dichos modelos. Finalmente, los marcos de representación, metodologías, patrones arquitecturales, procesos, y algoritmos diseñados para solucionar los problemas detectados se han aplicado al contexto de descubrimiento de información de las infraestructuras de datos espaciales.

Abstract

Information infrastructures are integrated solutions based on the fusion of information and communication technologies. Spatial data infrastructures are information infrastructures specialized in managing geographical information. They are characterized by the large amount of data that they have to manage, the high cost of obtaining such data, and the increasing number of applications based on these infrastructures.

An information infrastructure requires an efficient and effective information retrieval system to provide access to the items stored in the infrastructure. However, natural language terms used in classification, indexing and querying contain semantic relations between them may difficult the creation of effective search services. In this context, ontologies are frequently used to improve the performance of discovery systems. From the different types of ontologies, terminological ontologies are the most frequently used for classification and information retrieval. However, the heterogeneity of these models difficult their integration.

This PhD thesis focuses on the main problems that affect the discovery systems of information infrastructures to manage terminological models. Firstly, it analyzes the representation issues of terminological models in a context where a uniform representation is required. Secondly, it focuses on the need of acquisition of terminological ontologies. Specially, it analyzes the need to create, relate, and integrate the models required for an infrastructure. And thirdly, it deepens on the problem of accessing these models in an efficient manner via interoperable services and components.

As solution to such problems, this thesis proposes a common representation framework for terminological ontologies, a set of methods and techniques that facilitates the acquisition e integration of terminological models, and a set of architectural patterns that facilitates the management and access to such models. Finally, the frameworks, methodologies, architectural patterns, processes and algorithms designed to solve the detected issues have been applied to the discovery context of spatial data infrastructures.

Table of Contents

Chapter 1

Introduction

Information infrastructures are integrated solutions based on the fusion of information and communication technologies. An information infrastructure is defined as an advanced, seamless web of public and private communications networks, interactive services, interoperable hardware and software, computers, databases, and consumer electronics to make available vast amounts of information. The term started to be commonly used after the launching of the US plan for National Information Infrastructures [205]. Since then, the term has been widely used to describe national and global communication networks like the Internet and more specialized solutions for communications within specific business sectors. For example, following the US path, the European Union published some years later its own plan for the creation of European information infrastructures [47].

Information retrieval is a basic functionality in any information infrastructure. Information retrieval deals with the representation, storage, organization, and access to information items [11]. It consists in determining which documents of a collection are relevant to the user information request. The primary goal of an information retrieval system is to retrieve all the documents which are relevant to the user information need while retrieving as few non-relevant documents as possible. To do so, it has to be able to extract syntactic and semantic information from the documents, and use this information to rank the documents according to the degree of match with respect to the user information need. However, the interpretation of the user need is not an easy task. It is limited by the expressivity of the used query language and by the inherent ambiguity and terminological dispersion of the written text.

An information infrastructure requires an efficient and effective information retrieval system to provide the users access to the items stored in the infrastructure. It does not really matter how much information about a subject an infrastructure contains; if it is not possible to find it, it is uselessly. Therefore, it is important to distinguish between an information retrieval and a data retrieval process. While data retrieval systems are focused on determining which records stored in a catalog system contain the words specified in the user query, information retrieval ones are more concerned with obtaining information about a subject or topic than retrieving the data which satisfies exactly a given query. Data retrieval techniques are applicable to systems with well structured data where returning a single erroneous item means a total failure. However, in systems working with natural language text that is not always well structured and could be semantically ambiguous, information retrieval systems could be a better option if some inaccuracies and small errors are acceptable.

An information infrastructure is composed of several services and components that have to interact to provide the desired functionality. If each component uses a different set of interfaces and formats, the interoperability between them becomes a difficult task. The use of standards is of great help to solve syntactic interoperability problems establishing a common way to access to information (they provide a common syntax). However, syntactic interoperability is not enough for information retrieval. Natural language terms used in classification, indexing and querying contain semantic relations between them (e.g., synonymy, polysemy, homonymy, meronymy, hyponymy, lexical variants or misspellings) may difficult the creation of effective search services.

In order to increase semantic interoperability in search systems, libraries, museums, and archives have traditionally used controlled vocabularies (list of terms about a certain subject) to describe resources reducing in that way the possible terms used in classification and search to the selected ones. Their use increases the homogeneity in the descriptions, simplifies the query process and improves the results. Controlled vocabularies are used in classification steps to describe (and index) the resources. In the search components, they provide the user with the appropriate terminology for constructing queries. And in information browsing, they are used to provide a browsing structure through the resources based on the selected vocabulary. The selection of an appropriate vocabulary represents nevertheless an important challenge [76], it has to be adapted to the collection requirements avoiding terms irrelevant for the desired context.

Having in mind the increase of terminological precision, the use of simple controlled vocabularies has been progressively displaced by the use of more sophisticated knowledge models. This tendency has been greatly increased in the last years with the impact of internet and the semantic web. The knowledge models stored in paper (taxonomies, thesauri) by libraries and other institutions have been digitalized and transformed into more formal ontology models to provide a higher level of semantic. The term ontology is used in information systems and in knowledge representation systems to denote a knowledge model, which represents a particular domain of interest. A body of formally represented knowledge is based on a conceptualization: the objects, concepts, and other entities that are assumed to exist in some area of interest and the relationships that hold among them. And an ontology provides "an explicit formal specification of a shared conceptualization" [71]. Some ontology types are classification schemes that organize materials at a general level (such as books on a shelf); subject headings that provide more detailed access; authority files that control variant versions of key information (such as geographic names and personal names); or semantic networks and formal ontologies that provide a complete set of formally defined relations.

A special case of information infrastructures that is acquiring a great relevance nowadays is the one of spatial data infrastructures also known as geographical information infrastructures. Furthermore, this type of information infrastructure will be used as the application domain of this thesis. Geographical information represents a vital resource to be managed by institutions, organizations and the general public. Geographical information (also called geo-spatial data) describes phenomena associated directly or indirectly with a location with respect to the earth's surface. It has a complex structure conformed by a mixture of textual, graphical and spatial information that describes natural or artificial features such as rivers, cities or administrative boundaries. Each feature has a graphical facet that is usually represented by a geometry (point, line, polygon) composed of coordinates that allow us to locate it in the real word in a more or less precise way, and a textual component used to describe it. Sometimes, geographical information is also represented using other visualization models such as images, three-dimensional figures and graphs. Geographical information is vital for decision-making and resource management in diverse areas (natural resources, facilities, cadastres, economy . . .), at different levels (local, regional, national or even global). It can be said that around 70-80% of the databases used by the government contain geographical references (geographical coordinates, postal codes, addresses, administrative units . . .) [169].

In this field, the management of geo-spatial data has been traditionally performed by software packages called Geographic Information Systems (GIS) that allow capturing, storing, checking, integrating, manipulating, analyzing and displaying the required geo-spatial data. However, it must be noticed that the creation of geographical information is very expensive and time-consuming. According to Longley et al. [137], when geographical information is involved, the capture costs can be up to 50% of the total project cost. These costs are greatly increased if a high rate of geographical updates exists (e.g., changes in the property or its geometry). Therefore, the high costs of creation have required the development of networked solutions to facilitate the discovery, evaluation and access to geographic data with the objective of avoid, as much as possible, its replication. The potential of geographical information as an instrument to facilitate decision-making and resource management in so diverse areas and the need to provide a broader access to the available geographical resources has led to the evolution of GIS systems into the broader concept of Spatial Data Infrastructure (SDI), as a specific kind of information infrastructure specialized in geographical data.

An SDI is usually defined as a coordinated approach to technology, policies, standards, and institutional arrangements for an effective availability and access to geographic information. The European Committee for Standardization (CEN) defines the SDI concept as a platform-neutral and implementation-neutral technological infrastructure for geospatial data and services, based upon non-proprietary standards and specifications [173]. From the technical perspective, the widespread use of the SDI concept has meant an important revolution in the geographic information community, moving from monolithic and stand-alone applications towards a dynamic and cooperative environment of services and applications.

The SDI software components can be seen as a Digital Library (DL) specialized in geographical information, which, in addition to tools for storing and accessing digital (geographical) information, provides additional services needed in the geographical context (e.g. maps composition or geo-parsing). In the same way as a DL, and opposite to largely unstructured information available on the Web, SDI information is explicitly organized, described, and managed. In order to facilitate discovery and access, the content of their data resources is summarized into small descriptions, usually called metadata (data about the data), which can be either introduced manually or automatically generated (index terms automatically extracted from a collection of documents). Most DLs define their structured metadata in accordance to recognized standards such as MARC21 [155] or the Dublin Core Metadata Element Set [88] (proposed by the Dublin Core Metadata Initiative[1]). Additionally to the used in digital library contexts, SDIs components use other ones specialized in geographical information. For example, the ISO-19115 for Geographic Information Metadata [87] proposed both of them by the ISO/TC 211 Geomatics committee, the ISO-19119 for Geographic Services [94] also from ISO/TC 211, and the CSDGM-FGDC [51] developed by the *"Federal Geographic Data Committee"*[2].

Within the geospatial community the use of ontology models as a knowledge representation mechanism (as well as in the rest of the information infrastructures) is acquiring an increasing relevance for the development of Spatial Data Infrastructures (SDIs) and they are starting to play an essential role in SDI technology for interoperability solutions. One of the main aims in SDIs is to facilitate the so-called geospatial resource access paradigm in a dynamic and cooperative environment where interoperability plays a crucial role. As defined in the Global Spatial Data Infrastructure Cookbook [152], this paradigm represents an end-to-end communication between users and providers/brokers of geographic information where "successive iterations of resource discovery via metadata catalogs, followed by resource evaluation (such as Web Mapping Services), lead to data access either: direct as data sets, or indirect via data access services".

In this paradigm, ontologies play an important role. Depending on the ontology formalism level they are applicable to solve different resource access problems. Sowa [190] distinguishes two main classes of ontologies: terminological (also called lexical) and axiomatized (also called formal). Terminological ontologies are not fully specified by axioms and definitions and the relations are limited to subtype/supertype or part/whole relations. On the other hand, axiomatized ontology concepts and relations have associated axioms and definitions that are stated in logic or in some computer-oriented language that can be automatically translated to logic.

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

[2] http://www.fgdc.gov/

Each family may be applied to different steps in the geospatial resource access paradigm.

As concerns **resource discovery**, some of the most remarkable problems that affect the interoperability and cooperation of discovery systems are metadata schema heterogeneity and content heterogeneity [163].

As regards the problem of metadata schema heterogeneity, given that a metadata schema is a model that contains a set of concepts with properties and relations to other concepts, their structure can be modeled as a formal ontology, where metadata records are instances of this ontology [17]. This kind of formal ontologies may be used to profile the metadata needs of a specific geospatial resource and its relationships with metadata of other related geospatial resources, or to provide interoperability across metadata schemas.

For the problem of metadata heterogeneity, terminological ontologies facilitate classification of resources and information retrieval. Metadata try to exactly describe information resources to enhance information retrieval, but this improvement depends greatly on the quality of metadata content. One way to enforce the quality is the use of selected terminology for some metadata fields in the form of lexical ontologies. These ontologies are used to describe contents but also allow computer systems to reason about them. This role of terminological ontologies is even more significant in the case of developing a multilingual SDI where they can provide the translations of the terms used for classification to all the required languages.

Regarding **resource evaluation**, an SDI must facilitate the task of viewing detailed metadata, and must provide enough means to visualize the data appropriately. In this scenario, one could consider multilinguality and resolution level as main problems for system interoperability.

In the case of viewing metadata in a specific language required by the user, one may face the problem of having to translate it. Once again, formal ontologies and terminological ontologies may facilitate the work in two important aspects. Firstly, a formal ontology may provide the labels, in the appropriate language, for the elements of the metadata schema. Secondly, terminological ontologies may be used in the task of automatic translation of metadata to increase accuracy of translations.

Regarding the case of portrayal services for data visualization, one must face as well the problem of resolution level and "culture and linguistic adaptability". On the one hand, the resolution level affects portrayal of data because not all the features are meaningful at a particular zoom level. For instance, at a city scale level, it is worth visualizing the features of the urban transport network (streets, avenues, squares ...). However, these urban network features are not meaningful for a road network at national level. On the other hand, culture and linguistic adaptability may influence the results offered by portrayal services. Although the visualization of data seems language independent, SDI developers must consider the internationalization of legends and the display of internationalized attribute information if necessary. In this context, one could seriously consider the creation of an ontology of features visualized through portrayal services defining for each feature: the range of scales most appropriate for visualization, its textual label in every language, the most appropriate reference system for a geographic area, or the appropriate symbol (image) for rendering this feature on a map.

Finally, the **resource access** and further processing may benefit as well from the use of ontologies to facilitate data sharing and system development. Once again, formal ontologies help to define the meaning of features contained in geo-spatial data and they can provide a “common basis” for semantic mapping, e.g. to find similarity between two features that represent the same object but that have been defined using different languages. For instance, ISO/TC211 (technical committee for Geographic Information/Geomatics) has proposed several standardization items (ISO-19109 [93], ISO-19110 [91], ISO-19126 [90]) to create data dictionaries defining features and attributes that may be of interest to the wider international community. Furthermore, it is also usual in GI context to hear about extending the metaphor of Spatial Reference Systems (i.e., referencing things to some point on the ground) with the definition of Semantic Reference Systems [117]. The idea is that apart from spatial reference systems commonly used in maps and Geographic Information Systems (GIS), non-spatial components of geographic information should conform to some kind of semantic referencing.

Despite the benefits that ontologies can provide in the context of the geospatial resource access paradigm, the existent works solve specific problems for different SDI components without providing an integrated framework for ontology management. Different works develop their own ad-hoc solutions to solve specific problems but they cannot be easily shared and reused in other contexts. There is no global integration proposal for ontologies. The problem is especially relevant in the context of terminological ontologies, since they are intensively used along all the SDI components. This thesis work focus on providing a framework for the integration of terminological ontologies in an SDI, with the objective of facilitating its creation, management and use for the different components of the infrastructure requiring it. The integration problems that have been faced in this thesis work can be divided into three main general categories: representation, acquisition and access:

- Related to the **representation** problem, it is common to find that each organization has created a new ontology using an ad-hoc representation format, which is only useful in its specific context. This has led to a big heterogeneity of representation models increasing the difficulty and the cost of integrating them into a homogeneous system. In this context, a single and homogeneous representation mechanism for terminological models is vital in an SDI to provide uniformly the ontology models to the components that require them. An additional problem in SDI context is the need to provide a single and homogeneous access to different data collections classified with different terminological ontologies. For example, when integrating data from different countries classified according to different terminological models in different languages. To provide a homogeneous access to the resources, the used ontologies have to be related to be able to identify equivalences and obtain complete results. The process of matching ontologies (called ontology alignment) is difficult and costly but it can be reused for other collections using the same terminological models if they are properly represented. Therefore, in a similar way as it is required a representation format for individual terminological models, it is required another one for storing the mappings between them.

- Regarding the ontology **acquisition** problem, the needed ontologies have to be obtained or created, and adapted to integrate them in the required systems. However, this is not an easy task. On the one hand, the heterogeneity in the creation of terminological models limits their reusability in contexts different from the original ones. Therefore, even if a suitable terminology is found, it has to be transformed to facilitate its integration with the rest of the used in the system. This requirement adds additional integration issues due to the need of a different transformation process for each required ontology. On the other hand, the creation of a new one from scratch is very costly in time and resources. In this context, it is important to be able to reuse sections of other ontology sources that fit with the requirement of the new ontology to construct it, saving in that way time and effort. An additional issue that has been taken into account has been the overlapping of the acquired models. Here, the common elements of different models have to be controlled to diminish the classification problems that their use can cause.

- Finally, with respect to the **access** problem, due to the multidisciplinary character of SDIs and its applicability to a wide range of application domains, there is a great variety of terminological ontologies with very different levels of specificity, language coverage (i.e., from monolingual list of terms to multilingual thesauri covering more than 20 languages), formalization (i.e., from simple glossaries to well-structured thesauri) or size (e.g., AGROVOC thesaurus [130] contains more than 16,000 concepts). Additionally, it is important to note that they are distributed to the public trough ad-hoc services created for each institution providing them. This is not appropriate in an SDI context where it is required to provide them to all SDI components in a simple and common way. In this context, it is needed a coordinated view of the ontologies that can only be obtained through a homogeneous management and access to the terminological ontologies not dependent of the original providers.

In order to solve these specific problems, this thesis describes a homogeneous solution for each of the different discovery scenarios present in SDIs. These solutions are interrelated in such a way that they can be combined to facilitate all the steps required to integrate a new terminological ontology into an SDI.

1. In order to deal with the representation issues, the existent representation formats for terminological models have been analyzed. From them, the most appropriate has been

selected, extending it to cover those information requirements that were not fulfilled in the original format model. A similar work has been done with respect to the representation of mappings between different terminological models. In this case, given that no suitable format exists, a new one based on textual recommendations indicated in the terminological ontology standards has been designed.

2. With respect to the acquisition issue, each problem described has required a different approach, each one relying in the previous one as part of a global transformation process. First, a general transformation process, which harmonize the way a terminological ontology is converted to the selected representation format, is proposed. The format allows defining the structure of the source and destination models and simplifies the definition of relations between them. Additionally, an architectural pattern has been proposed to help to reuse the common elements of the different transformations. Secondly, to facilitate the interrelation of different terminological models, a process based on the use of a lexical database as nexus between the models has been developed. Thirdly, to simplify the construction of a new ontology, a process that uses a set of ontologies as base and combines them into a new model has been proposed. To focus the result into the desired domain, the process limits the content of the new ontology pruning the non relevant concepts. Finally, with the objective to increase the formalization of the models when required, a process that helps in the identification of the existent *is-a* relationships has been developed.

3. In the ontology management context, it has been identified the need for an efficient and common ontology management service to filter and select the most appropriate ontology for each specific context. But, previous to the creation of these services, the design of a common repository is proposed to store all the required terminological ontologies. On top of this repository, the design of an efficient editor and other GUI widgets is proposed to facilitate the annotation and the update of terminological ontologies. Additionally, a centralized ontology service, called Web Ontology Service (WOS), which enables uniform management of terminological ontologies (including discovery services) has been developed to provide access to terminological ontologies via Web services. To provide a full integration with the rest of components of a typical SDI, it follows and extends standard interfaces used in the geospatial community. Specifically, it follows the service definition of the OGC Web Service Architecture (WSA) [134], the standardized architecture for an SDI provided by the Open Geospatial Consortium (OGC).

Apart from this introduction, and the final chapter with conclusions and future research lines, this thesis consists of four chapters describing in detail the integration problems and the proposed solutions. The content of these chapters is are organized as follows:

- Chapter 2 reviews the types of ontologies and the techniques used for relating them as base for the selection of the most suitable ones (with the appropriate extensions) for this thesis. The concepts, ideas and some techniques presented there are used along the entire thesis as part of the integration solutions developed.

- Chapter 3 focuses on the problems of acquisition of terminological ontologies. Firstly, it analyzes the removal of the format heterogeneity through the transformation of ontologies into a single format. Secondly, it reviews the issue of using different overlapping models, and proposes a disambiguation method to relate them. Thirdly, it advances in the creation of new models using subsets of existing ones as base. Finally, it reviews the possibilities of improving the formalization of existing models.

- Chapter 4 analyzes the way to provide access to terminological models. It describes the structure of a terminological ontology repository, a tool for managing and editing them, and a web service for providing access to the systems requiring access to ontologies. Additionally, the query expansion procedures used to increases the quality of results obtained by analyzing the multilingual issues associated with geographical resources are described.

- Chapter 5 describes the integration of the developed systems and tools into SDIs. The systems where the components have been integrated are several: tools for creating geographical metadata that integrates management of terminological models, search clients able to use the stored ontologies to improve the search results, and browsing systems that provide access to the resources on base to the structure of a terminological ontology.

Chapter 2

Terminological ontologies: a framework for their representation

2.1 Introduction

According to Gruber [71], an ontology is an explicit specification of a conceptualization. A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose. A body of formally represented knowledge is based on a conceptualization: the objects, concepts, and other entities that are presumed to exist in some area of interest and the relationships that hold between them. Every knowledge base, knowledge-based system, or knowledge-level agent is committed to some conceptualization, explicitly or implicitly.

Following a more technical perspective, Sowa [190] defines an ontology as a specification of the kinds of entities that exist or may exist in some domain or subject area. Formally, an ontology is specified by a collection of names for concept and relation types organized in a partial ordering by the *type/subtype* relation. In a similar way, Guarino [73] defines an ontology as an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words. This set of assumptions has usually the form of a first-order logical theory where vocabulary words appear as unary or binary predicate names, respectively called concepts and relations. In the simplest case, an ontology describes a hierarchy of concepts related by subsumption relationships; in more sophisticated models, suitable axioms are added in order to express other relationships between concepts and to constrain their intended interpretation.

Ontologies are usually classified according to the amount and type of structure of their conceptualization. Following this criteria, Sowa [190] distinguishes two main families: terminological (also called lexical) and axiomatized (also called formal).

Terminological/Lexical ontology: An ontology whose concepts and relations are not fully specified by axioms and definitions that determine the necessary and sufficient conditions of their use. The concepts may be partially specified by relations such as *subtype/supertype* or *part/whole*, which determine the relative positions of the concepts with respect to one another, but do not completely define them.

Axiomatized/Formal ontology: A terminological ontology whose concepts and relations have associated axioms and definitions that are stated in logic or in some computer-oriented language that can be automatically translated to logic. There is no restriction on the complexity of the logic that may be used to state the axioms and definitions.

Sowa [190] states that a terminological ontology may be expressed in logic, but the logic required is usually simpler, less expressive, and more easily computable than full first-order predicate calculus. The distinction between terminological and axiomatized ontologies is one of degree rather than kind. They are models of different complexity in the same category. Axiomatized ontologies tend to be smaller than terminological ones, but their axioms and definitions can support more complex inferences and computations.

van Heijst et al. [210] propose another classification that divides ontologies into terminological, information and knowledge modeling. According to this classification, terminological ontologies specify the terms that are used to represent knowledge in the domain of discourse; information ontologies do the same with the record structure of databases (e.g., database schemata) and knowledge modeling ontologies are used to specify conceptualizations of the knowledge. Matching these classifications with the ones proposed by Sowa [190], terminological ontology definitions would be equivalent, knowledge modeling ontologies would fit into the axiomatized class and information ontologies would lie somewhat in the middle of the two, having many of the features of the formal ones but lacking some key elements such as clearly defined *is-a* relationships.

The treatment provided to the *is-a* relationship can be considered as the crucial point that separates terminological from axiomatic ontologies. Not everybody considers terminological models as ontologies, due to the lack of a formal explicit *is-a* hierarchy. The name Knowledge Organization Systems (KOS) is then used to refer to all the different models used to organize knowledge, reserving the name of ontology to axiomatized ontologies. As it is described by Hodge [79], the term “knowledge organization systems” intends to encompass all those types of schemas for organizing information and promoting knowledge management. A KOS serves as a bridge between the user information need and the material in the collection. Knowledge organization systems include: classification schemes that organize materials at a general level such as books on a shelf; subject headings that provide more detailed access; and authority files that control variant versions of key information such as geographic names and personal names.

In addition to the classifications based on the structure, ontologies can also be classified according to the subject of the conceptualization, that is to say, their content. Guarino [73] describes the following classes:

Top-level ontologies: They describe very general concepts which are independent of a particular problem or domain (e.g., space, time, matter, object, event and action). Therefore, it seems reasonable, at least in theory, to have unified top-level ontologies for large communities of users.

Domain and Task ontologies: These ontologies describe, respectively, the vocabulary (conceptualizations) related to a generic domain (like medicine, or automobiles) or a generic task or activity (like diagnosing or selling), by specializing the terms introduced in the top-level ontology.

Application ontologies: These ontologies contain all the definitions that are needed to model the knowledge required for a particular application. They describe concepts depending on a particular domain and/or task, and therefore, they are often specializations of ontologies of these classes. These concepts often correspond to roles played by domain entities while performing a certain activity.

Mizoguchi et al. [151] classification is quite similar to the one proposed by Guarino [73] but dividing the ontologies into general/common ontologies (top-level), domain ontologies and task ontologies. Another content-based classification is the one proposed by van Heijst et al. [210] that includes application ontologies, domain ontologies, generic ontologies (top-level) and representation ontologies. Representation ontologies describe the conceptualizations about knowledge representation formalisms [37], which are intended to be neutral with respect to world entities [74]. That is to say, they provide a representational framework without making claims about the world. Representation ontologies provide the primitives to formally represent and share the rest of the ontologies (top, domain, task and application).

Terminological ontologies are domain or application models that contain the terminology required in an area of knowledge for a specific application. They are intensively used by libraries, archives, museums and any other registry of information to facilitate the location of stored resources (classification and information retrieval). Along the years, they have proven to be a useful tool to deal with ambiguity problems, providing inter-relation structure and semantics to the terminology used in these systems.

The use of axiomatic models would be even better, because they provide additional semantics and formal specification of the relationships that could be used to improve the access to information. However, the great size of the required models (thousands of concepts) increases too much the complexity and cost involved in the creation of the model in comparison with the additional benefits obtained. Terminological models provide fewer semantics but they are simpler to create and bigger models are affordable.

Historically, terminological models were printed and used as thematic indexes to locate associated resources. The development of new applications have translated them into the computers and made them to evolve quickly. Nowadays, there is a great deal of terminological models covering every area of interest and they have become a crucial part of the information retrieval systems of digital libraries, catalogues and any other system where information is searched or presented thematically.

The heterogeneity of terminological models (differences in structure, content and representation) has been one of the basic problems with respect to their use in information systems. The problem gets bigger when the systems using different terminological models have to be integrated. Then the used models must be matched to be able to jump from the terminology used in one system to the terminology used in another. The geospatial community is not an exception with respect to the heterogeneity problem. Therefore, the establishment of spatial data infrastructures has raised the need of managing different models together in a simpler and harmonized way.

The first step needed in the process of harmonizing the management of terminological ontologies is the use of a common representation format. A common representation simplifies the construction of software having to manage multiple ontology models (only a single format has to be understood). This is required not only for the terminological ontologies, but also for the relationships defined between them. Relationships between ontology models are difficult to establish and must be properly represented to be able to reuse them when needed.

This chapter analyzes the different ontology types, the available representations and the processes to establish relations between them with the objective of selecting those that are more suitable for the geospatial context. First, a revision of types of ontology models is shown to compare them and detect the common characteristics and the differences. This analysis has the objective of providing the context in which the terminological ontologies are placed and showing how each of model is related to the rest. Additionally, an analysis of the available mapping technology used to relate ontologies is performed, focusing on those most useful for terminological ontologies and in the types of relations that allow establishing between the models. Secondly, having into account the types of models and relations between models described previously, a revision of the most common representation models for ontologies and for relations between ontologies is described. Finally, this chapter proposes a framework for the representation of terminological ontologies and their mappings. It includes a format for stand-alone terminologies and another one for mappings between models.

2.2 An overview of ontology types and the ontology mapping problem

This section reviews different ontology models describing their characteristics and remarking the similarities and differences between them. Additionally, the main mapping applications are analyzed to determine the suitability of application for terminological ontologies and to detect the types of relationships established between them.

2.2.1 Classification of ontologies according to their formalism and semantic degree

The main difference between ontology models is their capacity to express semantics. Terminological models do not have enough expressivity to represent the relationship complexity required by many applications, but they are much easier to create and integrate than formal ones. In this context, the selection of a terminological or a formal model depends exclusively on the required functionality. For example, to provide a list of possible values in a search service, the use of an ontology with hundreds of values and relations is not adequate. Here, a very limited terminological model adjusted to the collection content is much better. However, to model the information structure of a system where reasoning is needed, a formal model is required.

Lassila and MacGuinness [127] propose a classification of ontologies according to the degree of formalism and semantics provided in their specification (figure 2.1). They range from simple lists, passing by subject sets, to complex reasoning models. Following this classification, terminological ontologies correspond with the models on the left part of figure 2.1, and axiomatic ones with the models on the right.

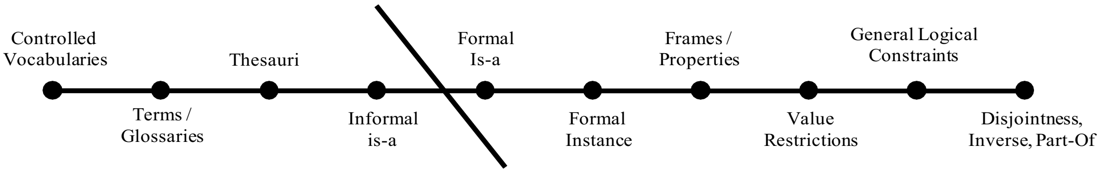


Figure 2.1: Categorization of ontologies extracted from Lassila and MacGuinness [127]

A complementary classification is described by Sigel [186] as shown in figure 2.2. This model focuses on explaining the different types of ontologies from the semantic interoperability point of view and focusing on the ability to express hierarchical relations. The categories range from taxonomies, which are able to express few semantics (subclassification relationship) and only provide syntactic interoperability, to logical theory models, which thanks to their strong semantics provide the most complete form of semantic interoperability. In this categorization, terminological models are those that provide syntactic and structural interoperability and the axiomatic ones those that provide semantic interoperability.

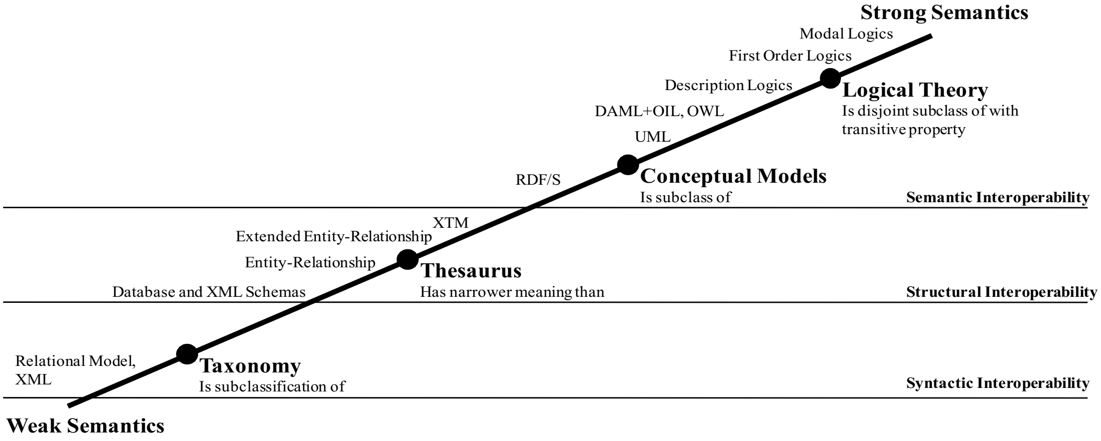


Figure 2.2: Categorization of Ontologies extracted from Sigel [186]

The evolution from one model to another can be done by adding semantics. For example, moving from taxonomies to thesauri involves the addition of the syndetic structure (connecting elements) which comprises the system of *see* and *see also* cross references to other indexing terms. And moving from thesauri to topic maps adds a greater number of typed semantic relations, internal attributes, typed links to external information resources, and sophisticated searching and displaying capabilities.

The most common types of ontologies are described bellow following the categorization by Lassila and MacGuinness [127] in figure 2.1. Additionally, some additional models that do not completely fit in any of its divisions are included within the most similar category or in a new one (if they are too different).

2.2.1.1 Controlled vocabularies

The simplest knowledge structure is a controlled vocabulary. It can be defined as a finite list of terms about a certain subject. Controlled vocabularies can be seen as lists of items published by a certain body that provide them with an unambiguous interpretation in the form of pairs Term-Identifier. A simple example of controlled vocabulary is the list of themes in a library where a unique code associated to each theme is used to classify the books.

More sophisticated controlled vocabularies are authority files (also called authority records). They are lists of terms used to control variant names for an entity or the domain values for a particular field. Authority files contain information to identify common type instances (e.g., geographic entities or administration departments), and provide a normative preferred name for resource classification and indexing. Their main function is to provide enough identifying information so that humans can identify the entity and associate it to a unique identifier and a preferred name. Standards such as ISAAR (CPF) [81] and MARC 21 Format for Authority Data

[154] define their structure and properties. Authority files do not include a deep organization or complex structure, no hierarchical relations are used and the syndetic structure is quite poor (i.e., there are not *see* and *see also* references to other indexing items). Examples of Authority Files are the Library of Congress Name Authority File[1] and the Virtual International Authority File[2]. Figure 2.3 shows an entry of the Library of Congress Name Authority File. In the table, *LC Control Number* field contains the unique identifier, *Heading* has the authorized label of the entry, *Used For* identifies the unauthorized forms of headings and other variants not chosen as an authorized form, and *Found In* has the source of the authorized label.

LC Control Number	n 86807672
Heading	Senate Democratic Caucus (Calif.)
Used For	California. Legislature. Senate. Democratic Caucus
Found In	nuc86-13536: Patino, L. The key that locks the records ... [MI] 1973 (hdg. on PSt rept.: California. Legislature. Senate. Democratic Caucus; usage: Senate Democratic Caucus)

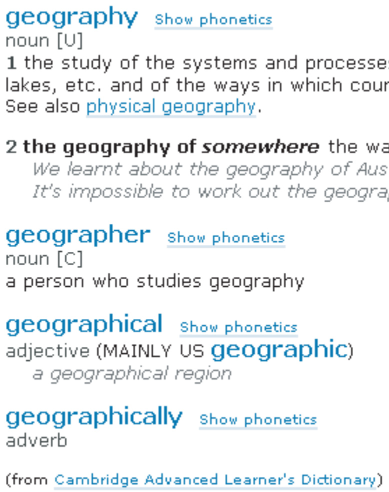
Figure 2.3: Entry of the Library of Congress Name Authority File

2.2.1.2 Glossaries

A glossary is a list of terms that usually contains definitions and cross-reference entries. The terms are defined within a specific environment and rarely include variant meanings. The definitions are directed to humans and therefore specified as natural language statements. Two examples are the terms of environment from the USA Environmental Protection Agency[3] and the UK National Statistics Geography glossary[4].

A dictionary can be considered as an enrichment of a glossary. Similarly to glossaries, dictionaries are alphabetical lists of words in a specific language with their definitions but including synonyms, spelling and morphological variants, multiple meanings across disciplines, etymologies, pronunciations, and other information. A terminology is a kind of domain specific dictionary including phrases instead of single words. Additionally, when the dictionary terms include translations in another language, they are known as lexicons. An example of dictionary is provided by the *Real Academia de la lengua*[5] (Royal Academy of Language) that contains the definitions of the words used in Spanish language. The Cambridge Dictionary[6] does the same for English language but additionally it provides translations to other languages. Therefore, it can also be considered as a lexicon. Figure 2.4 shows an element of this dictionary, showing the properties that it provides.

[1] http://authorities.loc.gov/
[2] http://orlabs.oclc.org/viaf/
[3] http://www.epa.gov/OCEPAterms/
[4] http://www.statistics.gov.uk/geography/glossary/default.asp
[5] http://buscon.rae.es/draeI/
[6] http://dictionary.cambridge.org/



geography Show phonetics
noun [U]
1 the study of the systems and processes involved in the world's weather, mountains, seas, lakes, etc. and of the ways in which countries and people organize life within an area
See also physical geography.

2 **the geography of *somewhere*** the way all the parts of an area are arranged within it:
We learnt about the geography of Australia.
It's impossible to work out the geography of this hospital.

geographer Show phonetics
noun [C]
a person who studies geography

geographical Show phonetics
adjective (MAINLY US geographic)
a geographical region

geographically Show phonetics
adverb

(from Cambridge Advanced Learner's Dictionary)

Figure 2.4: Definition of geography according to Cambridge Dictionary

2.2.1.3 Subject headings and taxonomies

Two types of models providing more semantics than a glossary and less than a thesaurus (subsection 2.2.1.4) are subject headings and taxonomies. These models are more complex than glossaries because of their hierarchical structure. They cannot be considered as thesauri because they lack their explicit definition of relationship types.

A subject heading is a uniform group of words used to describe the subject of library materials. It provides a set of controlled terms to represent the subjects of items in a collection. They have a very limited hierarchical structure and their terms can be coordinated to provide more specific concepts. Examples include the Medical Subject Headings[7] and the Library of Congress Subject Headings[8]. Figure 2.5 shows a subset of the Medical Subject Headings showing the structure and the description of one of its terms. As main elements, it provides *Unique ID* field as identifier, *MeSH Heading* as preferred label and two *Entry Terms* as alternatives. The other elements described provide additional information about the term such as *Annotation*, *Scope Note* and *History Note*, or for management, such as *Date of Entry*.

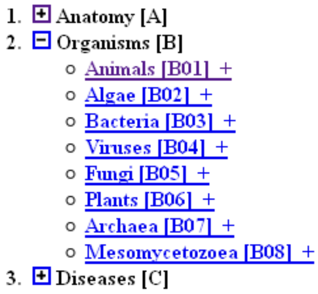
Subject categories are quite similar but their main objective is to group concepts as clusters of preferred and non preferred terms that share a single characteristic. Each subject category may have a number of characteristics, but only one is selected to collect and arrange the terms into a hierarchical order and create sub-categories. Subject categories are often used to group thesaurus terms in broad topic sets that lie outside the hierarchical scheme of the thesaurus. The AGRIS (international bibliographic information system for the agricultural sciences and

[7] http://www.nlm.nih.gov/mesh/

[8] http://www.loc.gov/aba/cataloging/subject/

technology) subject category[9] provided by FAO (Food and Agriculture Organization of the United Nations) is an example of this model.

A taxonomy (also called classification or categorization scheme) is a controlled vocabulary designed for classifying or categorizing resources. It provides aggregation of concepts using a (poly-)hierarchical broader concept based structure. Examples of classification schemes include the Library of Congress Classification[10], the Dewey Decimal Classification[11], or the Universal Decimal Classification[12].



1. ⊞ Anatomy [A]
2. ⊟ Organisms [B]
   - Animals [B01] +
   - Algae [B02] +
   - Bacteria [B03] +
   - Viruses [B04] +
   - Fungi [B05] +
   - Plants [B06] +
   - Archaea [B07] +
   - Mesomycetozoea [B08] +
3. ⊞ Diseases [C]

(a) Subset of the hierarchy

MeSH Heading	Animals
Tree Number	B01
Annotation	NIM as check tag; Manual 18.7+ . . .
Scope Note	Unicellular or multicellular . . .
Entry Term	Animal
Entry Term	Animalia
History Note	2004 (1974); was check tag only . . .
Date of Entry	19750725
Unique ID	D000818

(b) *Animal* term

Figure 2.5: Detail of *Animal* term from Medical Subject Headings

A folksonomy is a social variant of taxonomy. It is a classification scheme created by normal users instead of being created by experts in the area. They have become popular since the massive use of social web applications such as social bookmarking or photograph annotation. A well-developed folksonomy is ideally accessible as a shared vocabulary that is both originated by, and familiar to, its primary users. However, as they are created by many different non technical users, they usually lack consistency and liability in structure and content.

Another taxonomy related model is a gazetteer. A gazetteer can be viewed and modeled as a specialized kind of glossary, but nowadays they are modeled as taxonomies, thesauri or even more formal models. A classical basic gazetteer is a list of place names published as books or as indexes. In addition to a description, each element contains a feature type (e.g. city, road and mountain) and the coordinates for locating the place on the earths surface (if it is geo-spatially referenced). An example is the U.S. Code of Geographic Names[13].

[9] http://www.fao.org/scripts/agris/c-categ.htm

[10] http://www.loc.gov/catdir/cpso/lcco/

[11] http://www.oclc.org/dewey/

[12] http://www.udcc.org/about.htm

[13] http://geonames.usgs.gov/

2.2.1.4 Thesauri

A thesaurus is a set of terms describing the vocabulary of a controlled indexing language, formally organized so that the a priori relationships between concepts (e.g., synonymous terms, broader terms, or narrower terms) are made explicit [9, 83]. The vocabulary is arranged in a known order and structured so that the different relationships among terms are displayed clearly and identified by standardized relationship indicators that should be employed reciprocally.

According to Foskett [60], the main purposes of a thesaurus are to provide a standard vocabulary for indexing, to assist users with locating terms for proper query formulations, and to provide classified hierarchies that allow the broadening and narrowing of the current query request according to the user needs. The thesaurus structure was first defined to be used in the classification of resources (books, documents) to organize them thematically. Nowadays, their digital versions are used in data repositories (e.g., data libraries, management systems, geospatial catalogs) for classification and search purposes.

Thesauri have properties also present in some of the previous models, such as definitions or synonyms, but they add an explicit definition of concept relationships that can be interpreted unambiguously by agents. Different national and international standards have been developed to harmonize their structure. They can be divided into standards for the development of monolingual thesauri such as Z39-19 [9], BS-5723 [26] and ISO-2788 [83], and those having into account multilingual needs such as BS-6723 [25] and ISO-2788[83].

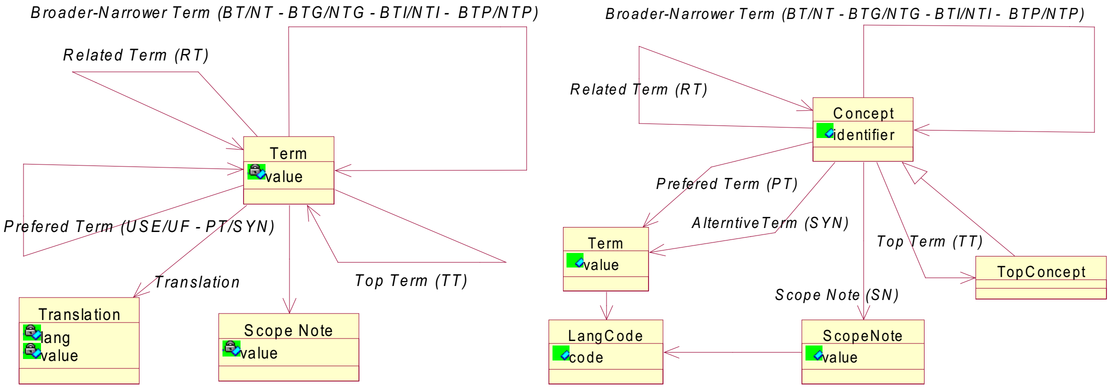
Relationships commonly expressed in a thesaurus include hierarchy represented using the notation BT (broader term) and NT (narrower term); equivalence described as SYN (synonym), and association or relatedness with RT (associative or related term). The root of the broader term hierarchy is described through the TT (top term) relationship. Additional properties such as SN (scope note) are usually included to identify the scope of use of each term. Thesaurus standards define three specializations of the broader-narrower relationship: the generic relationships, the instance relationships and the whole-part relationships.

Generic relationship (NTG/BTG): It identifies the link between a class and its members or species. This type of relationship is often identified with the *is-a* relation (e.g. Fruits NTG Citrus).

Instance relationship (NTI/BTI): It describes the link between a general category of things or events, expressed by a common noun, and an individual instance of that category, often a proper name. In more formal models these relations are managed as class instances and not as concept relationships (e.g. Mountains Regions NTI Alps).

Whole-Part relationship (NTP/BTP): It covers situations in which the meaning of one concept is inherently included in another one, regardless of context, so that the terms can be organized into logical hierarchies, with the whole treated as a broader term (e.g. Canada NTP Ontario).

Traditionally, the model to represent thesauri has been term-based (see figure 2.6a). Following this model, the term (lexical label) is the core of the structure and it is used as identifier. All the relations are between terms, not only including the BT/NT, RT and TT relations but also the synonymy between labels. Synonymy is managed with the USE/UF (used for) relationship that allows identifying which of the synonyms is the preferred one. Multilingualism is managed as translations of the terms and included as additional properties. Therefore there is a main language used as core of the structure and translations of those terms to other languages.



(a) Thesaurus model based on terms

(b) Thesaurus model based on concepts

Figure 2.6: Alternative thesaurus models

Nowadays, in the information science community, the used thesaurus model is concept-based (see figure 2.6b). In this model, the core element is the concept, which has a unique identifier used to distinguish each one from the rest. The BT/NT, RT and TT relations are relations between concepts. USE/UF relationship is replaced for the PT/SYN (preferred/synonym) relationship between concepts and terms containing the labels of the concept. Multilingual features are easily managed since all language dependent labels are managed uniformly. Two terms in different languages are associated to the same concept as preferred/alternative labels but each one containing a language code to indicate the language they are written. The same happens with other optional properties that are language dependent such as the *scope note* indicating the use of the concept and the *definition* of the concepts.

Some examples of thesauri used along this thesis are the following:

GEneral Multilingual Environmental Thesaurus[14]**(GEMET):** It has been created by the European Environment Agency (EEA) and the European Topic Centre on Catalogue

[14] http://www.eionet.europa.eu/gemet

of Data Sources to the European Environment Information and Observation Network for the classification of the developed environmental resources. It is available in 23 different languages (two of them different English dialects) and contains around 6500 concepts and 200000 descriptors.

AGROVOC thesaurus[15]: It is provided by the Food and Agriculture Organization of the United Nations (FAO) for the classification of geographic information resources (with special focus on agriculture resources) [130]. It is available in 19 languages (being 4 more are under construction) and contains around 28000 concepts and 600000 elements between properties and relationships.

European Vocabulary[16](EUROVOC): It is a multilingual thesaurus created by European Communities covering the fields in which the European Communities are active, for example, it provides a means of indexing the documents in the documentation systems of the European institutions and of their users. It is published in 23 languages containing around 6600 different concepts and 100000 descriptors summing all the provided in the different languages.

UNESCO thesaurus[17]: It is a general purpose thesaurus created by the United Nations Educational, Scientific and Cultural Organization (UNESCO) for its use in the indexing and retrieval of information in the UNESCO Integrated Documentation Network [204]. Published in English, Spanish and French contains around 4400 concepts.

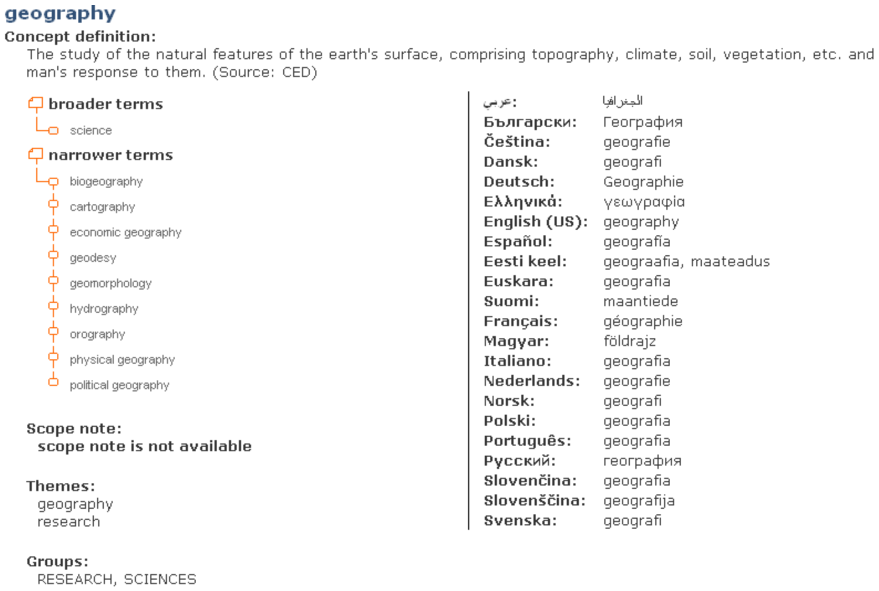
Figure 2.7 shows the content of the *geography* concept in the GEMET thesaurus. The identifier of the concept is not shown in the figure. It is hidden but it is required to identify and locate the concept. Some of the relations described previously are shown: the left part of the figure displays the *broader* and *narrower* relations; the right area contains the *preferred labels* for the available languages; and on the top there is a *definition*. Besides, GEMET includes themes and groups similar to the subject categories described in section 2.2.1.3 to group thesaurus terms in broad topic sets that lie outside the hierarchical scheme of the thesaurus.

In addition to the use of subject categories present thesauri such as GEMET, it is also common to structure thesauri information according to facets integrated in the thesaurus hierarchy. In this context, facet refers to a set of fundamental categories and their combination according to (synthesis) rules. This definition comes from Ranganathan [179], who uses it to denote aspects or viewpoints in library classification systems. Usually, one problem with classification systems is that items can be classified differently based on different purposes. Ranganathans idea was that class hierarchies can be built and combined for different purposes. For example,

[15] http://www.fao.org/aims/ag_intro.htm

[16] http://europa.eu/eurovoc/

[17] http://www.ulcc.ac.uk/unesco/



geography

Concept definition:
The study of the natural features of the earth's surface, comprising topography, climate, soil, vegetation, etc. and man's response to them. (Source: CED)

broader terms
- science

narrower terms
- biogeography
- cartography
- economic geography
- geodesy
- geomorphology
- hydrography
- orography
- physical geography
- political geography

Scope note:
scope note is not available

Themes:
geography
research

Groups:
RESEARCH, SCIENCES

عربي:	الجغرافيا
Български:	География
Čeština:	geografie
Dansk:	geografi
Deutsch:	Geographie
Ελληνικά:	γεωγραφία
English (US):	geography
Español:	geografía
Eesti keel:	geograafia, maateadus
Euskara:	geografia
Suomi:	maantiede
Français:	géographie
Magyar:	földrajz
Italiano:	geografia
Nederlands:	geografie
Norsk:	geografi
Polski:	geografia
Português:	geografia
Русский:	география
Slovenčina:	geografia
Slovenščina:	geografija
Svenska:	geografi

Figure 2.7: Geography concept from GEMET thesaurus

a piano is a musical instrument in an abstract typology of instruments but a piece of furniture for the purpose of interior design.

2.2.1.5 Semantic Networks

A semantic network can be seen as a generalization of a thesaurus where the relationship structure between concepts and terms is not hierarchical but a net of relationships. The relationships generally go beyond the standard BT, NT, RT and usually include specific *whole-part*, *cause-effect*, or *parent-child* relationships. An example is WordNet[18], a lexical database of English created by the Princeton University [52]. WordNet is a large English lexical database that groups nouns, verbs, adjectives and adverbs into sets of cognitive synonyms (synsets), each expressing a distinct concept. Those synsets are interlinked by means of conceptual-semantic and lexical relations.

WordNet is structured in a hierarchy of synsets, defining a synset as a set of strict synonyms representing one underlying lexicalized concept, and providing semantic relations (synonymy, hypernymy, hyponymy, meronymy, holonymy . . .) among these synsets. Evolving from WordNet, EuroWordNet [213] was developed as its multilingual version consisting of a set of cross-related WordNets in several languages (French, German, Spanish, Dutch, Italian, Czech, Estonian and English). It includes the semantic relations between words of each provided European language and the relations among each word and the equivalents in the other languages.

[18] http://wordnet.princeton.edu/

Although WordNet and EuroWordNet hypernymy/hyponymy relationships could be mapped to an *is-a* (in the sense described in section 2.2.1.6 most of the times), they are not used homogeneously, involving a fairly loose semantic association. An example in the medical field is the semantic network of UMLS (Unified Medical Language System) [136].

A specific type of semantic network is a topic map. It is a representation of knowledge, with an emphasis on the find-ability of information. A topic map defines a multidimensional topic space in which the locations are topics, and the kinds of relationships define the path from one topic to another [89]. It can be seen as an aggregated semantic network that group similar items into topics, associations, and scopes. The term "topic" refers to the object or node in the topic map that represents the subject being referred to. In addition, each topic can be assigned to a type of resource to classify it and to an explicit name to refer to it. ISO-13250 standard [89] defines a topic map with the characteristic of being able to assign multiple base names to a single topic, and to provide variants of each base name for use in specific processing contexts. In the standard, variants were limited to *display name* (used for presentation) and *sort name* (using for alphabetic sorting). There is a one-to-one relationship between topics and subjects, with every topic representing a single subject and every subject being represented by just one topic.

A topic may be linked to one or more information resources called occurrences that are relevant to the topic. Such occurrences are generally external to the topic map document and they are referenced using URIs or a similar mechanism. Occurrences may be of different types; such distinctions are supported in the standard by the *occurrence role*, and identified by an *occurrence role type.* To be able to describe relationships between topics, the topic map standard provides a construct called *topic association* where the associations between topics are described and grouped by their type. Each topic that participates in an association plays a role in that association called the *association role.* A topic map example is the UNSPSC Topic Map[19] that documents the entire Universal Standard Products and Services Classification (UNSPSC). The UNSPSC is a schema that classifies and identifies commodities. It is used in sell side and buy side catalogs and as a standardized account code in analyzing expenditure (Spend Analysis). Figure 2.8 shows the content of *Geography charts or posters* node of UNSPSC in XTM format [192]; property *id* in *topic* tag contains the identifier of the topic; *instanceOf* shows the topic type; *baseName* has the topic label; and *subjectIdentity* references to the subject that is reified by the topic. *Association* shows the structure of a relation of a topic with another one, with *instanceOf* containing the type of relation, *member* specifying each topic that is part of the relation and *roleSpec* containing the role that each topic plays in the relation.

In the same way as thesauri provide additional built-in relationships and properties with respect to taxonomies, topic maps extend thesauri adding a more flexible model with an open vocabulary. For topic maps the representation of simple ontology models can be done by

[19] http://www.techquila.com/tmsamples/xtm/unspsc/unspsc_11.zip

```
<topic id="entry.60.10.34.01">
    <instanceOf>
        <topicRef xlink:href="#commodity"/>
    </instanceOf>
    <subjectIdentity>
        <subjectIndicatorRef xlink:href="urn:x-unspsc:60.10.34.01"/>
    </subjectIdentity>
    <baseName>
        <baseNameString>Geography charts or posters</baseNameString>
    </baseName>
</topic>
<association>
    <instanceOf>
        <topicRef xlink:href="#assoc-class-commodity"/>
    </instanceOf>
    <member>
        <roleSpec>
            <topicRef xlink:href="#class"/>
        </roleSpec>
        <topicRef xlink:href="#entry.60.10.34.00"/>
    </member>
    <member>
        <roleSpec>
            <topicRef xlink:href="#commodity"/>
        </roleSpec>
        <topicRef xlink:href="#entry.60.10.34.01"/>
    </member>
</association>
```

Figure 2.8: XTM *Geography charts or posters* node of UNSPSC topic map

establishing the structure of the required classification using topic map roles [62].

2.2.1.6 Is-a Hierarchies and Formal Instances

Formal hierarchies include strict *is-a* subclass relationships. According to Brachman [24], *is-a* hierarchies can be divided into two major subtypes: one relating two generic nodes (classes) where usually the associated node is less general; and the other relating a generic node with an individual being described by the general description (instantiation). From the categories of each subtype described by Brachman [24], *superset/subset* and *generalization/specialization* for relations between generic nodes, and *set membership* between a generic and a individual are the most commonly used. *Instance-of* nomenclature is commonly used for generic to individual relationships leaving *is-a* for generic to generic relationships. From now on, this is the way the nomenclature *is-a* and *instance-of* are going to be used.

An *is-a* relationship is transitive in the way indicated in equation 2.2.1 where *ISA(x,y)* stands for *x "is-a" y*. *Subset* relationship adds the additional requirements shown in equation 2.2.2 where *SUBSET(x,y)* means *x "is-a-subset-of" y* and *MEMBER(z,x)* (defined in equation 2.2.4) means that z is member (instance) of the collection (class) x (e.g., *SUBSET(Citrus, Fruits).*

On the other hand, *specialization* relationship is a relation between predicates that affects two arbitrary predicates (P_1 and P_2 in equation 2.2.3) where *SPECIALIZATION (P_1,P_2)* stands for P_1 *"is-a-specialization-of"* P_2 (e.g. *SPECIALIZATION (creation date, date).* Thesaurus NTG relationship is similar to *is-a* relationship, but mostly defined with the *subset* meaning.

$$\mathtt{ISA}(\mathtt{x},\mathtt{y}) \wedge \mathtt{ISA}(\mathtt{y},\mathtt{z}) \rightarrow \mathtt{ISA}(\mathtt{x},\mathtt{z}) \tag{2.2.1}$$

$$\mathtt{SUBSET}(\mathtt{x},\mathtt{y}) \rightarrow \forall \mathtt{z}(\mathtt{MEMBER}(\mathtt{z},\mathtt{x}) \rightarrow \mathtt{MEMBER}(\mathtt{z},\mathtt{y})) \tag{2.2.2}$$

$$\mathtt{SPECIALIZATION}(\mathtt{P}_1,\mathtt{P}_2) \rightarrow \forall \mathtt{x}(\mathtt{P}_1(\mathtt{x}) \rightarrow \mathtt{P}_2(\mathtt{x})) \tag{2.2.3}$$

$$\mathtt{MEMBER}(\mathtt{x},\mathtt{y}) \rightarrow \mathtt{x} \in \mathtt{y} \tag{2.2.4}$$

Richer models include *instance-of* relationships, i.e., formal membership (instances) associated to the model entities. Equation 2.2.4 shows the membership relationship explained before (e.g., *MEMBER(Alps, Mountains Regions)*). Membership is a transitive relation with respect to the *is-a* relationships: $\mathtt{MEMBER}(\mathtt{x},\mathtt{y}) \wedge \mathtt{ISA}(\mathtt{y},\mathtt{z}) \rightarrow \mathtt{MEMBER}(\mathtt{x},\mathtt{z})$. This property is intensely used in information discovery for expanding query and indexing terms. It is difficult to find an equivalence of *Membership* in simpler ontology models; the most similar is the NTI relationship but with the lack of explicit transitivity. $\mathtt{NTI}(\mathtt{x},\mathtt{y}) \wedge \mathtt{NTG}(\mathtt{y},\mathtt{x}) \rightarrow \mathtt{NTI}(\mathtt{x},\mathtt{z})$ cannot be automatically deduced (it has to be specifically stated if it is required).

There is a great difference between *is-a* used as a formal relationship and previously described similar relationships, such as the NT used by thesauri. Each *is-a* subtype provides a strong semantics and is used coherently and homogeneously in the models (e.g., *subset* is not merged with *instance-of*). However, for thesaurus construction, NT is the relationship used by default, merging the different semantics of NTG, NTI and NTP under the same relation name.

2.2.1.7 Frame based ontologies

A *frame* is a named data structure which is used to represent a concept in a domain. According to Schaerf [182], a *frame* usually represents a concept (or a class) and it is defined by an identifier, and a number of data elements called *slots*, each one corresponding to an attribute that members of the class can have. The values of the attributes are either elements of a concrete domain (e.g. integers, strings) or identifiers of other frames. Frame based ontologies include classes and property information that is specified at a general class level and inherited by subclasses and instances. For example, if a class *travel* has as properties *origin* and *destination*, a specific travel instance may have *origin:Zaragoza* and *destination:London.* Any subclass in the *is-a* hierarchy of travel, such as for example *low cost travel*, has all *travel* properties and relations, and it can add additional specific ones. This structure of organizing knowledge is similar to the classic software object oriented modeling techniques.

Frame models usually include other elements of representation of knowledge such as *facets*. A facet is used here to represent information about a *slot* containing a series of descriptive properties informing about the corresponding attribute and constraining the possible values. They are used to specify default values, value restrictions and attached procedures for computing values when needed or for propagating side effects when the slot is filled. The most commonly used constraints types are: *Cardinality, Minimum-Cardinality, Maximum-Cardinality, and Value-Type.* An example of *Value-Type* facet can be the restriction to a specific range of numbers to the value associated to a relation called *price*. The meaning of facet in this context must not be misunderstood with the sense of facet used in section 2.2.1.4 for thesauri.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:owl="http://www.w3.org/2002/07/owl#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <owl:Class rdf:ID="City"/>
    <owl:Class rdf:ID="Travel">
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty>
                    <owl:ObjectProperty rdf:ID="Origin"/>
                </owl:onProperty>
                <owl:cardinality rdf:datatype=
                    "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
        <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
    </owl:Class>
    <owl:ObjectProperty rdf:about="#Origin">
        <rdfs:range rdf:resource="#City"/>
        <rdfs:domain rdf:resource="#Travel"/>
    </owl:ObjectProperty>
    <owl:DatatypeProperty rdf:ID="Name">
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
        <rdfs:domain rdf:resource="#City"/>
    </owl:DatatypeProperty>
</rdf:RDF>
```

Figure 2.9: Example of OWL-Lite ontology

An example of frame based ontologies is the one proposed by Chaudhri et al. [29], which provides a uniform model based on a common conceptualization of classes, individuals, slots, facets and inheritance to share knowledge. Protégé [165] and Ontolingua Server [50] are examples of two ontology construction tools that support the construction of frame based ontologies (between others models). In this context, but related with software engineering, UML language provides all the needed elements ('is-a" hierarchy, classes, property definitions, cardinality specification and value restrictions) to construct frame based models but with representation capabilities more reduced than the provided with some complex frame based models.

In the semantic web context, the Web Ontology Language (OWL) [15] has become the "de facto" standard to represent formal ontologies. It has a subset called OWL-Lite that supports the set of characteristics required to create frame based ontologies. OWL is constructed on top of RDF-Schema, and therefore it shares many of RDF-Schema properties. It can be said, that RDF-Schema provides the basic structure to construct basic frame based ontologies, and OWL-Lite enriches it by allowing adding *facets* to the model. Figure 2.9 shows a very simple example of a frame based ontology in OWL-Lite format. In the example the classes *city* and *travel* are defined; the property *name* is added to the *city* class but limiting their values to strings. It adds the relation *origin* between a *travel* and a *city*, but adding a constraint to indicate that it must be unique (*owl:cardinality*).

2.2.1.8 General Constraints and Disjointness

Frame based ontologies are limited in their capability to express complex constraints between elements, not being able to support some requirements of the information systems. For example, the value of one property may be based on the value of two other properties; or it may be needed to express disjointness between classes, stating explicitly that an instance of a class A cannot be an instance of a different class B.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:owl="http://www.w3.org/2002/07/owl#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <owl:Class rdf:about="#MusicDrama">
        <owl:equivalentClass>
            <owl:Class>
                <owl:unionOf rdf:parseType="Collection">
                    <owl:Class rdf:about="#Opera"/>
                    <owl:Class rdf:about="#Musical"/>
                </owl:unionOf>
            </owl:Class>
        </owl:equivalentClass>
    </owl:Class>
    <owl:Class rdf:about="#Opera">
        <rdfs:subClassOf rdf:resource="#MusicDrama"/>
    <owl:Class rdf:about="#Musical">
        <rdfs:subClassOf rdf:resource="#MusicDrama"/>
        <owl:disjointWith rdf:resource="#Opera"/>
    </owl:Class>
</rdf:RDF>
```

Figure 2.10: Example of OWL extracted from Bechhofer et al. [15]

The ontologies on top of the formal ontology classification are able to express this additional complexity with arbitrary logical statements using first order logic constraints between terms, but at the cost of an increased complexity that make them to lack of computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). For example, a class can be treated simultaneously as a collection of individuals and as an individual in its own right but no reasoner can automatically make use of it. Figure 2.10 shows an ontology containing the *union of* and *disjointness* properties typical of these ontologies.

These ontologies can be represented using different family of languages focused on expressing first order predicates, each one with their limitations in the statements able to define or use. A first family is the one based on frames such as the described by Minsky [150], Fikes and Kehler [54], that are used in knowledge tools such as Protégé [165] and Ontolingua Server [50], CycLp [132] or the Frame Logic language used by OntoEdit[20] [194]. Other family is based on Description Logics such as Classic [20], KIF [64], its successor CL [96] and Ontolingua language [72] also based on KIF. OWL [15] has also an explicit logical basis for the language based on description logics, but the limitations of the RDF-Schema based representations difficult the expression of some types of predicates (see Horrocks and Patel-Schneider [80]). The last family contains general purpose declarative languages such as CLIPS [65] o JESS [61].

2.2.2 Ontology mapping

Nowadays, there is a great need to provide single homogeneous access to different data collections independently of the type. In this context, the diversity of the used knowledge models hinders in great extent the possibilities of integration. Data collections created by different organizations usually use different ontologies (or different versions of the same ontology) to classify and index the resources. Along the years, each organization with the need to describe a collection has created specific "ad hoc" ontologies for the required purpose. Additionally, the different requirements of data collection, which can range from legacy systems to new electronic data catalogues, have made that the used ontologies vary in the formalism degree.

The definition of equivalence relations between the ontologies used in data collections is a solution usually adopted to provide a unified view. This subsection describes the problem of establishing these relations and the most common types of relations established between concepts of terminological models.

The existent ontology models of any area of knowledge are not completely independent; they usually have fragments also contained in other ones. For example, focusing on the thesauri models, the classification needs in numerous areas of knowledge have promoted the creation and diffusion of well-established electronic thesauri that usually partially overlap in their content with others of the same area. However, the structure provided in each ontology model for the overlapping areas is usually different. For instance, both GEMET and UNESCO thesauri (see section 2.2.1.4) contain equivalent terminology about the natural environment (among other subjects). However, in GEMET this terminology is concentrated in a branch, and in UNESCO it is distributed along the thesaurus.

[20] http://www.ontoknowledge.org/tools/ontoedit.shtml

Some initiatives have tried to harmonize the existent models creating general ontologies that can be used in multiple situations. Some examples are: CYC [131], a universal schema of roughly 10^5 general concepts spanning human reality (it has an open version called OpenCYC[21]); the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [141], that aims at capturing the ontological categories underlying natural language and human commonsense; WordNet [52], a lexical database designed as a semantic network based on psycholinguistic principles; or the Suggested Upper Merged Ontology (SUMO) [156], developed for research and applications in search, linguistics and reasoning. However, these projects are far from creating a single shared ontology. As Lesk [133] states, while a single ontology would be advantageous, it is unlikely that such a system will ever be developed. Culture constrains the knowledge classification scheme, so that, what is meaningful to one culture is not necessarily meaningful to another one. This is not only true with different cultures but also in the same area of knowledge, by the difficulty of obtaining the agreement of different groups about a unified classification scheme.

An additional problem is the semantic heterogeneity of natural languages. In natural languages, many concepts can be characterized in different ways imposing each one a slightly different view of the world. This ambiguity is expressed through semantic relations between the language terms. From the different relations in natural languages, the following ones are the most frequent:

Synonymy: A relationship between two terms that have the same meaning. Two exact synonyms are equivalent and can be used indistinctly; however, it is very common to have partial synonyms that share the meaning only in some contexts.

Polysemy: It is the capacity of a term to have multiple meanings (related in some way).

Homonymy: The relation between two words that are spelled and/or pronounced the same way but differ in meaning. The difference with respect to polysemy is that while homonymy is a relation between two different words, polysemy represents the multiple meanings of a single word.

Meronymy / Holonymy: Meronymy denotes the relation in which a term is a constituent part of, or a member of other one. Its opposite is holonymy.

Hyponymy / Hypernymy: Hyponymy is the semantic relation in which one word meaning is included within other one. Its opposite is hypernymy.

Gender equivalence: Languages such as Spanish and English have terms that change completely depending on the gender (e.g., cow, bull). In some contexts they can be considered equivalent for search purpose, therefore the equivalence between them has to be stated.

[21] http://www.opencyc.org/

Translations: Terms from different languages can also be equivalents in the way described by synonymy (having exactly the same meaning). However, is much more habitual to have terms in different languages that only partially overlap in meaning, making difficult the definition of equivalence relations.

Most of the terminological models are able to represent all these relations. Synonymy and translations are managed through the use of alternative labels (gender equivalence can be represented in a similar way); polysemy and homonymy is reduced by adding definitions, examples and scope notes; and meronymy and hyponymy is represented using subset and specialization relations. However, the models generated are not complete; they only include the information required for the purpose for which they were created. Therefore, queries to systems combining data classified according to different ontologies may perform poorly independently of which ontology has been used in the query construction (many of the records may contain terms different from the ones used in the query).

The identification of these relationships (e.g., equivalence or subsumption) between entities of different ontology models allow reducing the semantic heterogeneity problem and provide a unified view. As remarked by Koch et al. [114], the identification of relations is a complex process that depends on imprecise objectives such as the desired browsing structure, display needs, depth, use of non-topical classes, and the trade-off between consistency, accuracy and usability. The process of finding these relationships is known as ontology alignment or ontology matching, and the representation of these alignments into a format that can be understood for a computer system to perform a specific task is called ontology mapping. Doerr [44] defines a matching as “the process of identifying terms, concepts and hierarchical relationships that are approximately equivalents”.

An alignment between two ontologies means to find for each entity (concept, relation or instance) in the first ontology, a corresponding entity with the same intended meaning in the second one. Alignment is not only restricted to identity functions, other possible alignment relations such as subsumption and instantiation can also be used. Ehrig [46], (chap. 2) defines an ontology alignment as a partial function according to equation 2.2.5, where E is the set of all entities, O the set of possible ontologies, and R the set of possible alignment relations. Given two ontologies $o_1, o_2 \in O$, an entity $e \subset o_1$ is aligned with an entity $f \subset o_2$ according to an alignment relation $r \in R$ if $align_{o_1,o_2}(e, r) = f$.

$$align = E \times O \times O \rightharpoonup E \times R \tag{2.2.5}$$

From the computer science perspective, an alignment is a set of correspondences $\langle e, f, r, l\rangle$ with e and f being the two aligned entities, r representing the relation holding between them, and l expressing the level of confidence if it is needed. The directionality of the alignments depends on the alignment relation. For symmetric relations, the alignment is bidirectional; in any other case, to provide an alignment in the opposite direction an inverse alignment relation have to be identified, or two different unidirectional alignments have to be provided (one for each direction).

Ontology alignment is used in any mediation process where it is needed to obtain interoperation between ontology models for any specific task (e.g., query rewriting or instance transformation). Most of these processes are involved within the following tasks:

Integration: It focuses on reusing one or more ontologies to create a new ontology. The source ontologies are related without losing their independence. It is usually used to increase the domain of the resulting model. Alignment is used to detect overlapping between the integrated ontologies. Fernández-Breis and Martínez-Béjar [53] describe a framework for the integration of ontologies supplied by a predetermined set of expert users. Every user could benefit from what other users had already contributed to create his integration-derived ontology. Calvanese et al. [28] present another framework for ontology integration, where a global ontology is used to provide a unified view for querying local ontologies. It addresses the problem of specifying the mapping between the global and the local ontologies.

Merging: It is the combination of two or more ontologies into a new unified one that replaces the originals. Individual elements of the original ontologies are present in the new one, but they cannot be traced back to their source. Alignment between the source ontologies is used to identify their equivalences and perform the merging. McGuinness et al. [143] show an ontology editing, merging, and diagnostic environment called Chimaera. Kotis and Vouros [115] describe the HCONE approach on ontology merging. It is based on capturing the intended informal interpretations of concepts by mapping them to WordNet senses using lexical semantic indexing, and exploiting the formal semantics of concepts by means of description logics reasoning services.

Transformation: It changes the semantics of an ontology to make it suitable for other purposes. The alignment is needed to model the changes from the original to the transformed ontology. Versioning of ontologies can be viewed as a kind of transformation where the final objective is to obtain an improved model. Klein and Fensel [113] discuss the problem of ontology versioning comparing it with database schema and program library versioning; they propose building blocks for the most important aspects of a versioning mechanism. Tennis [196] focuses on versioning representation and suggests extensions to SKOS Core to make explicit differences between versions.

Translation: It refers to changing the representation format of an ontology. Alignment is needed not for the ontology itself but for the meta-models of the source and destination format, that is, to relate each element of the source and destination format models. van

Assem et al. [209] describe a method for converting existing thesauri and related resources (terminological ontologies) from their native format to RDF(S) and OWL. van Assem et al. [208] advance in the same line, improving the process to provide transformation to SKOS.

2.3 State of the art in the representation of terminological ontologies and ontology mappings

Ontologies have to be properly represented to facilitate their interchange. Not only that, relations between two ontologies need to be represented if they want to be reused in other contexts. As this thesis is focused on the use of terminological ontologies, the analysis of the possible representation models has been centered on these models. Firstly, this section presents a revision of the state of art in the representation of terminological models. Then, it describes the existent alternatives to do the same for the mappings established between two terminological ontologies.

2.3.1 Representation of terminological ontologies

Each different ontology type provides different semantic expressiveness. As mentioned in the introduction, the distinction between the different types of ontologies is one of degree rather than kind, where more complex models are able to represent the features contained in the previous ones, adding new additional characteristics. The representation of these models in a computer system is done through representations formats adapted to each model type. Until recently, the lack of standardized representation formats has produced the creation of a great variety of incompatible ad-hoc formats, created for specific ontologies and only used by the organizations that created them. Nowadays, the information community has reached agreements about the most suitable representation formats for some of the ontology models and it has standardized them. For other ontology models, there is still no complete consensus about their representation.

Along the years, specific adapted representation models have been created for each different ontology type. Sections 2.2.1.7 and 2.2.1.8 introduced some representation formats for the described formal ontology models. These representations formats are very complete and given that they can represent elaborated models, they can also be used for the simpler ones. However, a format that is not perfectly adapted to the model that tries to represent increase its difficulty of use. There are many primitives, properties and attributes that are not required and there are several ways to represent the same.

Focusing on terminological models, even simpler models such as controlled vocabularies or glossaries have some specific representation formats for them. Terminological Markup Framework (TMF) [85] is a meta-model that allows the definition of different Terminological Markup Languages for specific purposes. Two XML based formats created using this framework are the Geneter and the MSC (Machine-Readable Terminology Interchange Format with Specified Constraints) described both of them in the TMF standard. Geneter is a format to describe data categories and their relationships in a terminological data collection while MSC is designed to represent terminological data for the processes of analysis, dissemination, and exchange of information from human-oriented terminological databases (termbases). Another alternative representation framework is Term Base eXchange (TBX) [98], an open XML-based standard for exchanging structured terminological data. In a similar way to TMF, it allows defining a variety of terminological markup languages. For other similar models, also specific representation formats exist. For example, the Lexical Markup Framework (LMF) [97] is an abstract meta-model that provides a common, standardized framework for the construction of computational lexicons that can be mapped to XML based representation. For authority files, the XML representation schema proposed by MARC-21 standard[22] can be considered.

Taxonomies and thesauri have also their own representation formats. Traditionally, each company has created their own ad-hoc formats to represent their taxonomies and thesauri. For example, the most popular thesauri used in geospatial science for classification of resources such as AGROVOC, EUROVOC or GEMET where initially generated in completely different formats.

Nowadays, some initiatives have tried to create homogeneous representation formats for thesauri. For example, the ADL thesaurus Protocol [102] defines an XML and HTTP based protocol for accessing thesauri that returns portions of the thesaurus contained encoded in XML. Another approach is the Thesaurus Interchange Format in RDF proposed by the Language Independent Metadata Browsing of European Resources (LIMBER) project [142]. Additionally, the California Environmental Resources Evaluation System (CERES) and the NBII Biological Resources Division collaborated in a Thesaurus Partnership project[23] for the development of an Integrated Environmental Thesaurus and a Thesaurus Networking ToolSet for Metadata Development and Keyword Searching. One of the deliverables of this project is another RDF format to represent thesauri.

For taxonomies, there are some general representation formats such as the one used in *Dewey Decimal Classification (DDC)*[24] [41] and *Universal Decimal Classification (UDC)*[25] [144, 145]. But they are oriented to human visualization instead of computer processing and interchange. The existent computer oriented interchange formats are specific ad-hoc representations similar to the used for thesauri.

[22] http://www.loc.gov/standards/marcxml/

[23] http://ceres.ca.gov/ thesaurus

[24] http://www.oclc.org/dewey/

[25] http://www.udcc.org/about.htm

Finally, in the topic maps context, XML Topic Maps (XTM) format [192] is the most frequently used, being supported by many tools in quite different contexts.

All these formats have been designed to describe the terminological ontologies of a certain kind, but are not specifically adapted to be able to describe in a coherent way at least a common subset of different types of them. British standards BS-5723 [26], BS-6723 [25] and their international equivalent (ISO-2788 [83] and ISO-5964 [82]) propose models to manage monolingual and multilingual thesauri that can be also applied to simpler models but they lack a suitable representation format. The British Standards Institute IDT/2/2 Working Group has recently finished the 5th part of BS-8723 standard [27] that describes an exchange format and protocols for interoperability for terminological ontologies following the thesaurus model. It is focused on thesauri, but it can be used to represent other terminological models. This format is based on XML, and it is expected that it will be promoted to ISO as part of the revision of the ISO-5964 standard (norm for multilingual thesauri) that is currently undergoing review by ISO-TC46/SC-9.

In the semantic Web area, the Simple Knowledge Organization System (SKOS) project[26] [148, 147] has become the reference to represent a broad set of terminological ontologies used for classification such as subject heading lists, taxonomies, classification schemes, thesaurus, folksonomies, controlled vocabularies, and also concept schemes embedded in glossaries and terminologies. SKOS was initially developed within the scope of the Semantic Web Advanced Development for Europe[27] (SWAD-E). SWAD-E was created to support W3C's Semantic Web initiative in Europe (part of the IST-7 programme). It is based on a generic RDF schema for thesauri that was initially produced by the DESIRE project [34], and further developed in the Limber project [142]. It has been developed as a draft of an RDF/OWL Schema for thesauri compatible with relevant ISO standards, and later adapted to support other types of terminological ontologies. SKOS is still under review but different drafts describing the structure already exists.

2.3.2 Representation of ontology mappings

Creating a good alignment between two terminological ontologies is an expensive task (in time and cost). Even using automated matching process (such as the ones described in section 3.3.1); the results have to be manually revised and updated to remove inconsistencies. Due to the difficulty to reduce these costs, at least the obtained mappings should be represented in a way that facilitates their reuse. However, this aspect has not attracted much attention in the research community. The existent works for alignment focus mainly on improving the quality of the mappings obtained but not in how to represent them in a standardized and reusable way.

[26] http://www.w3.org/2004/02/skos/

[27] http://www.w3.org/2001/sw/Europe/reports/thes/

The representation of these terminological ontologies is being covered with the development of standards such as the ones described in section 2.3.1, but there are no works so advanced for mapping representation. The standards used to describe thesauri and similar models such as BS-5723 [26], BS-6723 [25] and their international equivalent (ISO-2788 [83] and ISO-5964 [82]) describe slightly the mapping needs but they do not provide a suitable representation. In a similar way, the Z39.19-2005 [9] (revision of Z39.19-1995) makes some more specific references to mapping between thesauri but does not provide either a mapping model nor representation format.

The most advanced proposal for mapping representation is the one developed in the context of the SKOS project [146, 148], where a draft version of a mapping model and interchange format (RDF based) called SKOS-Mapping have already been developed (see figure 2.11). However, the proposed representation format is still preliminary and it is under revision due to deficiencies such as the lack of structure in the mapping types and the types of connectors provided. SKOS-Mapping model proposes a set of mapping relations between concepts. Additionally, to provide 1:N relationships the concepts can be aggregated in a *rdf:Bag* structure by different composition functions. The meaning of each mapping relation and each composition function is described next in this section.

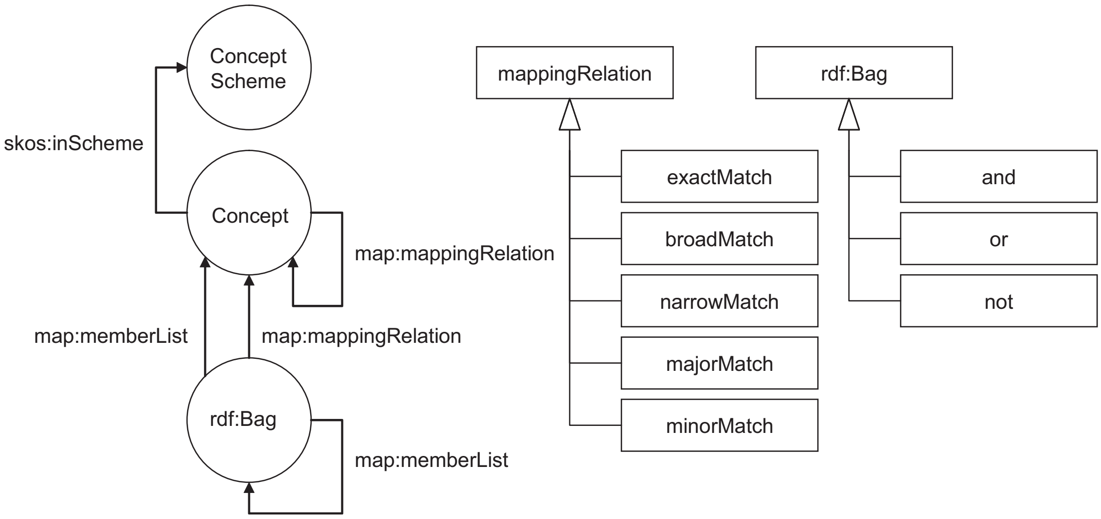


Figure 2.11: SKOS-Mapping Model

Given the lack of an established representation model and interchange format for mappings, it is needed to define one suitable for the context of this thesis (see section 2.4.3). An initial step in this direction has been to analyze the representation requirement, describing the available alternatives in terms of structure, relations, and properties required to represent the mappings.

As previously indicated, a mapping is a representation of an alignment between ontologies.

It represents the axioms that describe how to express concepts, relations or instances in terms of the second ontology [46]. Focusing on the thesaurus model as representative of terminological ontologies, ISO-5964 reduces the required types of inter-thesaurus relations to the following three: exact, inexact and partial equivalence. They correspond to the different types of alignment relations that can be considered in the matching process (see section 2.2.2). The top half of Venn diagrams in figure 2.12 show graphically their semantics. According to ISO-5964 they have the following meaning:

Exact equivalence: An exact equivalence is established between a source language term and a target language term when both of them have identical meanings. It is a bidirectional synonymy relation where the involved concepts can use different identifiers to represent the same concept. This inter-thesaurus relationship is a kind of generalization of the bidirectional intra-thesaurus synonymy relationship between the preferred and the alternative labels of a thesaurus concept or between the different language dependent labels in concept from a multilingual thesaurus.

Partial equivalence: It is the association between a source and a target term when both cannot be matched by an exact equivalence and one has either a broader or a narrower meaning than the other, but not both. That is, the meaning of one of the terms is completely contained within another one. This relation cannot be directly used because there is no way to distinguish which concept is the general one and which is the specific one. However, it can be expressed using two inverse relationships that show the directionality of the relation. It is equivalent to the *hyponymy* and *hypernymy* relationships used to construct the concept hierarchy of a thesaurus.

Inexact equivalence: It is established between a source and a target term when they express the same general concept, but their meanings are not identical and none of them is contained in the other one. They can be considered as partial-synonyms, and in many situations, two concepts holding subtle differences (inexact equivalent) are finally classified as exact equivalent in a given context for practical purposes. This relation provides quite few semantic, it does not give a hit of the degree of similarity between the concepts (they may be almost equivalent or practically different); therefore, different specializations indicating the degree of similarity between the terms are sometimes used. For example, naming the relations as major/minor to indicate more or less similarity between the concepts, or even using a numerical percentage to indicate their degree of equivalence).

In the draft of the SKOS-Mapping (see figure 2.11), mapping relations have been represented by means of the *skos:mappingRelation*, a generic relation to indicate any kind of mapping. It specializes into *skos:exactMatch* for exact equivalences; *skos:broadMatch* and *skos:narrowMatch* for partial equivalence; and *skos:relatedMatch* for inexact mappings.

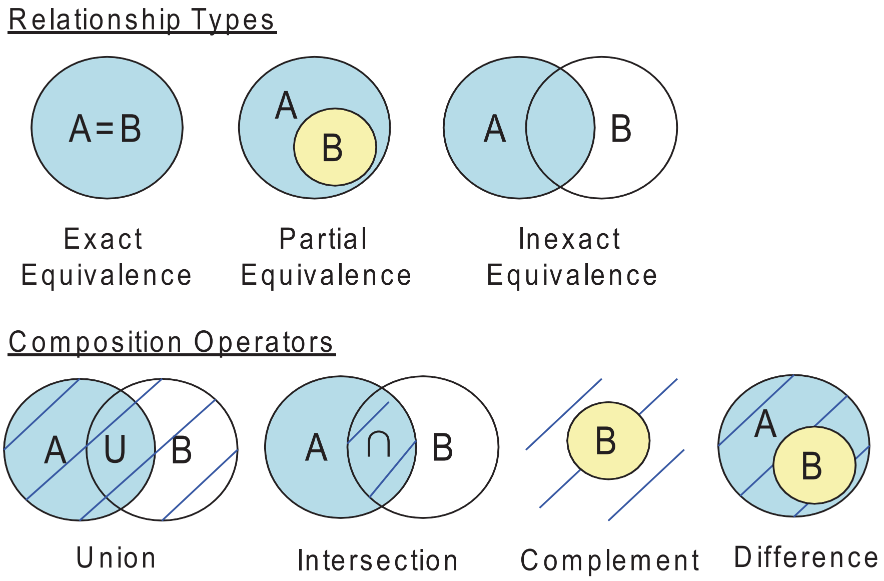


Figure 2.12: Types of Mapping Relationships

Having in mind the application of the defined mappings in the discovery process of an information retrieval system, and especially in query systems, Doerr [44] refines the definition of these relations and shows how they can be used to create a consistent set of mappings between ontologies. He indicates that the creation of an arbitrary set of equivalence expressions for correlation makes the replacement of terms in queries unpredictable. He proposes to provide a broader and a narrower relationship for each concept with the objective of improving automatic translation of queries. The mappings have to be created systematically by assigning to each source concept the nearest broader and narrower in the target model. For those concepts were finding a broader and a narrower cannot be possible, at least one should be provided.

The described mapping relationships allow defining 1:1 relationships between concepts. However, as Doerr [44] states, “the expressive power of the mapping should be at least equivalent to the expressiveness of the search paradigm, otherwise the user could express better queries in each target system than the mapping mechanism could provide”. Greater cardinality such as 1:N (single to multiple equivalence) is then needed to deal with situations were exact mappings cannot be found, but a combination of the meaning of a set of concepts of one ontology is equivalent to a concept in another one.

If multiple equivalence relationships need to be defined, they also have to be properly represented. However, nowadays there is not a real consensus about which composition operators are needed. ISO-5964 does not define precisely the nature and types of composition. The technical specifications provided by the Z39.50 protocol use any combination of mathematical (logical) operators such as *intersection*, *union*, and *complement* to create combined concepts and map them [8]. Boolean algebra operators (AND, OR, NOT) are used indistinctly to *union*, *intersection* and *complement*, for example in SKOS-Mapping first draft [146] (see figure 2.11). BS-8723 remarks that only the *intersection* operator should be used because, in a practical context, the only composition operator really used to perform equivalences with other concepts is the intersection between a few concepts. BS-8723 draft goes further completely rejecting the *complement* operator as a viable option for composition of concepts.

The *intersection* composition operator is accepted due to it covers practical mapping requirements. It is used to create concepts whose meaning is restricted to the common elements of two (or more than two) other concepts. For example, the concepts *animal* and *biology* can be combined to create the *animal biology* concept of GEMET; then this concept can be used to classify the records that are about both of the original subjects. The set of records classified according this new concept would be the intersection of those classified with *animal* and those with *biology*.

With respect to the *union* operator, it is easy to imagine situations where it can be required. However, they are usually hypothetical applications with a low interest for the construction of real systems. For example, a composed concept equivalent to *tree* would be a set of concepts containing all the different tree spices. However, it is not reasonable to think that the ontologies to mach, if are not specifically focused on that matter, would contain all those elements. A subset of them could be composed with *union*, but the associated mapping could not be considered as exact. Additionally, the semantic meaning of this possible equivalence would be the same as providing different partial equivalences for each concept (which is simpler). For mappings in contexts with very specific terminology it can be applicable, but is not a typical situation.

The use of the *complement* operator is even more limited. It has to be used in combination with other ones due to the extent of the result obtained (everything except the indicated concept). A suitable alternative is the *difference* operator (*A and not B*), which is commonly employed in information retrieval systems to reduce the possible senses of a concept used in a query. It is applicable for multilingual mappings where two terms can be considered as exactly equivalents if a part of the meaning of one of them is removed. For example, the Spanish term *pierna* is equivalent to English *leg* but it is only used for humans. Therefore, *pierna* can be seen as *exact equivalent* to the *difference* between *leg* and *animal leg*. However, it can be replaced many times by the intersection operator (e.g., *pierna* is also the intersection of *leg* and *human*).

The bottom half of the figure 2.12 describes the semantics of these operators by means of Venn diagrams.

2.4 A framework for the representation of terminological ontologies

This section presents a framework for that has as objective to facilitate the management of terminological ontologies. Having into account all representation formats described previously and the representation needs in discovery components of SDIs, some suitable representation formats for terminological ontologies and mappings between ontologies have been selected.

The selected format for terminological ontologies is SKOS based. It has been extended to fulfill some specific requirements, such as the need to represent additional concept properties and the selection of a suitable description model to identify and classify the used ontologies. With respect to the representation of mappings, given the lack of a suitable representation format, a new one based on BS-8723 terminology has been developed.

2.4.1 Representation of ontology concepts

From the different available representation alternatives described in section 2.3.1, SKOS can be used to describe many different terminological models. This format is the most suitable for the desired classification and retrieval context where several ontology models are required. However, it is still under development and not all the needed characteristics are covered. Given this situation, it has been needed to extend it to deal with the situations not covered in the original SKOS format.

An advantage of using SKOS is that is it is becoming a de-facto standard for represent some types of terminological models. SKOS has been already used to represent some thesauri such as GEMET, AGROVOC, ADL Feature Types or some parts of WordNet lexical database (see SKOS project web page[28]).

As it is described in the SKOS reference document [147], the SKOS data model is formally defined as an OWL Full ontology. The "elements" of the SKOS data model are classes and properties, and the structure and integrity of the data model is defined by the logical characteristics of and interdependencies between those classes and properties. However, SKOS is not a formal knowledge representation language because terminological ontologies do not assert any axioms or facts, their structures do not have any formal semantics, and they cannot be reliably interpreted as either formal axioms or facts about the world. As mentioned by Miles and Brickley [147], SKOS is needed because OWL structure is not the most adequate for expressing terminological models, "It is not appropriate to express the concepts directly as classes of an ontology, or to express an informal (broader/narrower) hierarchy directly as a set of class subsumption axioms". Using SKOS data model, the "concepts" are modeled as individuals, and the informal descriptions and the links between those "concepts" are modeled

[28] http://esw.w3.org/topic/SkosDev/DataZone

as facts about those individuals.

SKOS is still under review, but the core structure of the format described in the drafts is expected to be quite stable. At the moment, SKOS is a collection of three different RDF-Schema application profiles:

SKOS-Core: It provides a model for expressing the basic structure and content of concept schemes, understanding them as a set of concepts, optionally including statements about semantic relationships between them. It is the basic profile used to define terminological ontologies and provides a model to represent the common properties and relations shared by most of the terminological models.

SKOS-Extensions: They are a set of terms extending the SKOS Core vocabulary to support some features of specific knowledge organization systems, especially thesauri.

SKOS-Mapping: Its purpose is to describe relations between different ontologies. It is done providing mappings between concepts of different concept schemes. It is reviewed in section 2.3.2

It is expected that, when finished, SKOS documentation will provide some guidelines to facilitate its extension with specific properties and relations existent in some terminological models. In this context, SKOS-Core would contain the set common to all terminological models, creating specific extensions to provide the required additional elements.

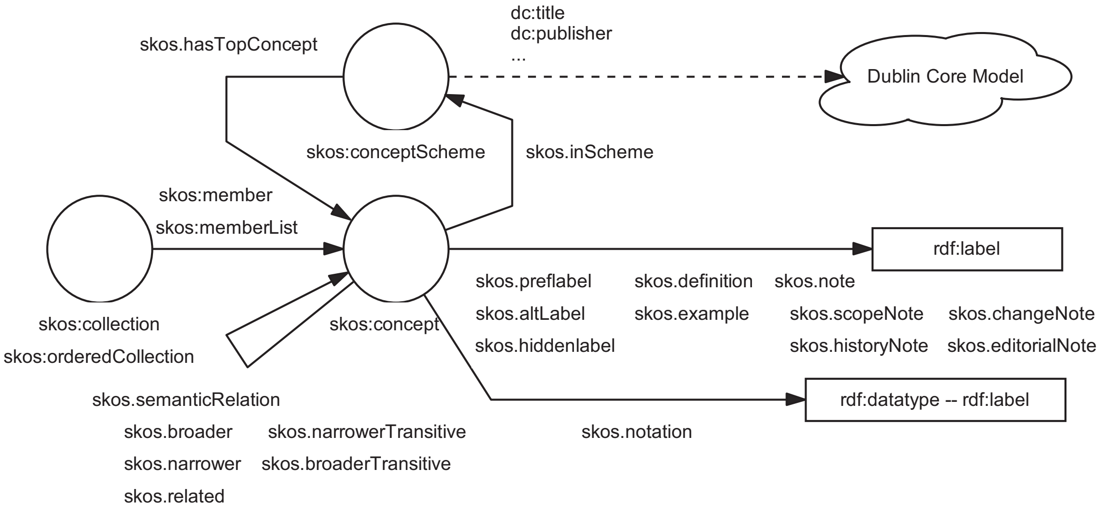


Figure 2.13: SKOS-Core Model

The structure of elements and relations of the SKOS-Core application profile is described in Figure 2.13. The model can be divided in two kinds of elements: firstly, those used to define the ontology structure; and secondly, those describing the lexical properties of each represented term.

The structure of the model is described by a small set of elements. The basic one is the *skos:concept*. It is used to represent an abstract or symbolic tag that attempts to capture the essence of the reality (it is identified by an URI). A SKOS-Core file consists of a set of concepts grouped in a *skos:conceptScheme*. The *skos:conceptScheme* structure is the entry point to the ontology. It identifies the whole ontology with an URI and refers to the upper concepts contained inside. Additionally, the *skos:conceptScheme* can contain metadata describing its content to facilitate its use to the persons requiring it.

To indicate that a *skos:concept* is part of a *skos:conceptScheme* (belongs to it), the *skos: inScheme* relation is used. This relation allows a concept to be part of more than a schema, making possible to create views of a model containing only certain subsets of it by defining different concept schemes on the same set of concepts. The relation of the *skos:conceptScheme* with the concepts of the ontology is defined by the *skos:hasTopConcept* relation. This relation points to the *skos:concept*(s) which are topmost (top concepts) in the hierarchical structure of concepts for that scheme. If the represented model is flat (no hierarchy), there will be a *skos:hasTopConcept* relation for each concept in the model.

To provide relations between concepts, SKOS defines a general relationship called *skos: semanticRelation* that indicates that exist a link between two *skos:concept* (the type is not indicated). All the different hierarchical and associative relationship types defined by SKOS are specializations of it. *skos.broader* and *skos.narrower* relations are inverse relations used to model the hierarchical characteristics of many terminological ontologies. They indicate that one concept is, in some way, more general than the other. *skos.broader* is used to describe the relation from the specific concept towards the general one and *skos.narrower* for the opposite. These two relations are not transitive, and therefore can only be used to assert an immediate hierarchical link between two *skos:concept*. Transitive equivalents for these relations are *skos:broaderTransitive* and *skos:narrowerTransitive*. Associative relations between concepts are represented using *skos.related*. It indicates that two concepts are related in some way maintaining a symmetric relation between them. Figure 2.14 contains a subset of EUROVOC thesaurus that shows how some of these elements and relationships are represented in SKOS.

An additional element included in the last version of SKOS model has been the capacity to define faceted based structures (in the sense described in section 2.2.1.4). To do this, the *skos:collection* and *skos:orderedCollection* are used. They allow defining labeled and/or ordered groups of SKOS concepts that share a property, when the value of this property can be used to group the concepts under different categories. *skos:collection* is used for general collections and *skos:orderedCollection* for collections where the order of the elements is relevant (e.g. for visualization). The relation between the collections and the *skos:concept*(s) contained inside is done using the relationship *skos:member* for *skos:collection* and *skos:memberList* for

```
<rdf:Description rdf:about="http://europa.eu/eurovoc">
    <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#ConceptScheme"/>
    <dc:title>EUROVOC 4.1</dc:title>
    <skos:hasTopConcept rdf:resource="http://europa.eu/eurovoc/Domain04"/>
    <skos:hasTopConcept rdf:resource="http://europa.eu/eurovoc/Domain68"/>
    ...
</rdf:Description>
<rdf:Description rdf:about="http://europa.eu/eurovoc/Domain68">
    <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
    <skos:inScheme rdf:resource="http://europa.eu/eurovoc/eurovoc"/>
    <skos:prefLabel xml:lang="en">INDUSTRY</skos:prefLabel>
    ...
    <skos:narrower rdf:resource="http://europa.eu/eurovoc/MicroThes6811"/>
    <skos:narrower rdf:resource="http://europa.eu/eurovoc/MicroThes6821"/>
    ...
</rdf:Description>
<rdf:Description rdf:about="http://europa.eu/eurovoc/MicroThes6811">
    <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
    <skos:inScheme rdf:resource="http://europa.eu/eurovoc/eurovoc"/>
    <skos:prefLabel xml:lang="en">chemistry</skos:prefLabel>
    ...
    <skos:broader rdf:resource="http://europa.eu/eurovoc/Domain68"/>
    <skos:narrower rdf:resource="http://europa.eu/eurovoc/Concept3810"/>
    ...
</rdf:Description>
```

Figure 2.14: Fragment of SKOS file from EUROVOC Thesaurus

skos:orderedCollection.

The lexical properties of the terminological ontologies are directly included into the *skos: concept*(s) structure. Since these properties are language dependent (they contain terms that are part of a specific natural language), an attribute is used to specify the language used in their content. The most relevant properties are *skos.preflabel* and *skos.altLabel*, which provide the labels used for classification and visualization. *skos.preflabel* contain the label that better identifies a concept (for thesauri it must be unique). On the other hand, *skos.altLabel* contains synonyms or spelling variations of the preferred label, and it is used to redirect to the preferred label when required. *skos.hiddenlabel* is a kind of alternative label but containing common misspellings of the preferred term. It can be used for comparison in search systems, but not to be visualized by the user. The SKOS example shown in 2.15 shows the preferred and alternative labels of some concepts according to the SKOS format.

In addition to these properties, *skos:notation* has been defined to represent alternative identifiers, not recognizable as a word or sequence of words in any natural language, that identify uniquely a concept within the scope of a given concept scheme. Since they are not described in a natural language, they cannot be represented using the label properties. *skos:notation* is especially useful for classification schemes that provide multiple codes of terms. An example of this category is the ISO-639 [84] (ISO standard for coding of languages), which proposes

```
<rdf:Description rdf:about="http://europa.eu/eurovoc/Concept3810">
    <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
    <skos:inScheme rdf:resource="http://europa.eu/eurovoc/eurovoc"/>
    <skos:prefLabel xml:lang="en">chemical compound</skos:prefLabel>
    <skos:prefLabel xml:lang="es">compuesto químico</skos:prefLabel>
    ...
    <skos:altLabel xml:lang="en">compound, chemical</skos:altLabel>
    <skos:altLabel xml:lang="es">químico, compuesto</skos:altLabel>
    ...
    <skos:broader rdf:resource="http://europa.eu/eurovoc/MicroThes6811"/>
    <skos:narrower rdf:resource="http://europa.eu/eurovoc/Concept3817"/>
    ...
    <skos:related rdf:resource="http://europa.eu/eurovoc/Concept2739"/>
</rdf:Description>
```

Figure 2.15: Fragment of SKOS concept from EUROVOC Thesaurus

different types of alphanumeric codes (e.g., 2 letter and 3 letter codes) to represent the existent languages. At the moment, it has not been established a representation for this property.

The need to represent models with this characteristic has required the definition of a representation able to manage notations of different types. The solution used has been to add inside the *skos:notation* an *rdf:datatype* containing the type of notation defined, with the objective of being able to distinguish between different identifiers created with different purposes. Figure 2.16 shows a fragment of the ISO-639 in SKOS using the *skos:notation* property, which distinguishes between the three code-sets for languages using notations with different *rdf:datatype*.

```
<rdf:Description rdf:about="http://www.iso.org/ISO639-3/eng">
    <skos:inScheme rdf:resource="http://www.iso.org/ISO639/ISO639"/>
    <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
    <skos:prefLabel xml:lang="en">English</skos:prefLabel>
    <skos:prefLabel xml:lang="es">inglés</skos:prefLabel>
    <skos:prefLabel xml:lang="fr">anglais</skos:prefLabel>
    <skos:prefLabel xml:lang="de">englisch</skos:prefLabel>
    <skos:scopeNote xml:lang="en">Living language</skos:scopeNote>
    <skos:notation rdf:datatype="http://www.iso.org/ISO639-3/">eng</skos:notation>
    <skos:notation rdf:datatype="http://www.iso.org/ISO639-2/">eng</skos:notation>
    <skos:notation rdf:datatype="http://www.iso.org/ISO639-1/">en</skos:notation>
</rdf:Description>
```

Figure 2.16: Fragment of SKOS file of ISO-639 classification scheme

The last set of properties in the *skos:concept* model are those focused on documentation, providing human-readable, informal documentation to the user. SKOS provides a *skos:note* property for general documentation purposes and it is used to indicate additional information associated to the concept. To provide documentation elements with more specific semantic, some specializations of *skos:note* are defined. The main documentation properties are *skos:definition*, and *skos:example*. As it can be deduced by their name, *skos:definition* supplies a complete explanation of the intended meaning of a concept, and *skos:example* an example of the use of the concept. *skos:scopeNote* is also quite relevant; it provides information about the intended meaning of a concept in the specific context of the ontology. It is especially used as an indication of how the use of a concept is limited for indexing. Finally, *skos:historyNote* describes significant changes to the meaning or the form of a concept.

Other *skos:note* specializations also exist, but they are oriented to be used as part of the management process of the terminological ontology, and not to be provided to the final user. *skos:editorialNote* supplies management information (e.g., reminders of editorial work still to be done). *skos:changeNote* documents fine-grained changes to a concept for the purposes of administration and maintenance.

2.4.2 Metadata to describe terminological ontologies

A terminological ontology, independently of the representation, has to be properly described to be able to identify its content. A user has to know what each terminological model is about to be able to decide which one suits better to his requirements.

In order to describe general ontologies the Ontology Metadata Vocabulary[29] (OMV) developed in OWL can be used. This is a metadata vocabulary to describe any type of ontology, and it is quite complete. However, to describe the content of a SKOS terminological ontology, a simpler metadata model adapted to the description requirements of terminological models is preferred. A suitable alternative is Dublin Core [88] because it is a standard for representing metadata. Additionally, it is extensively used in the digital library area to classify resources and there is a lot of experience in its use in different contexts. It provides a simple way to describe a resource using general metadata terms, which can be easily matched with complex domain-specific metadata standards.

Although Dublin Core metadata vocabulary is general, this is not a problem since it can be extended to define application profiles for specific types of resources such as terminological ontologies. Following the metadata profile hierarchy described in Tolosana-Calasanz et al. [198], an application profile for the description of ontologies that refines the definition and domains of Dublin Core elements has been created (see table 2.1) using the IEMSR format[30] [77].

The metadata profile includes a subset of the basic Dublin Core elements adding the *applied in* field for describing the thematic context in which the ontology can be used, and the following metadata management fields extracted from ISO-19115 standard [87]: *metadata language* to indicate the language of the metadata, *metadata identifier* to identify the metadata record, *metadata creation date* to store the date when the metadata was created and *metadata point*

[29] http://ontoware.org/projects/omv

[30] IEMSR is an RDF based format created by the JISC IE Metadata Schema Registry project to define metadata application profiles

of contact to indicate who created the metadata. Table 2.1 contains all the metadata elements included in the metadata profile, together with their identifiers, the label used to describe them, their obligation, their cardinality, and a description of the element.

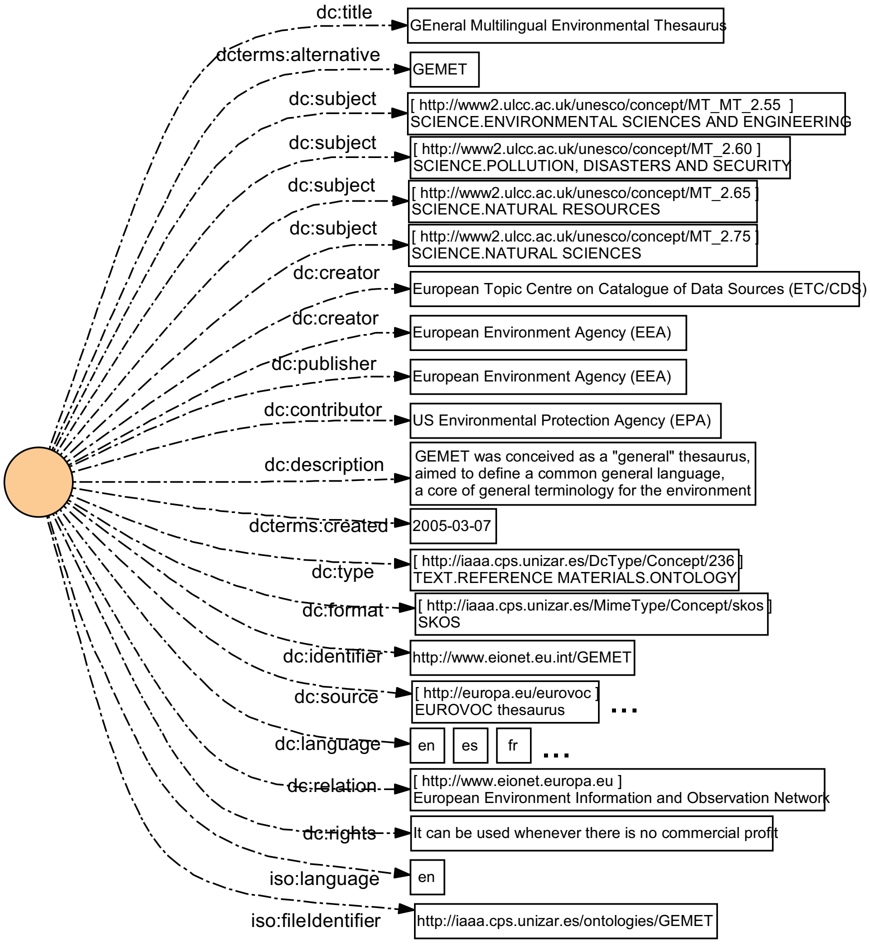

dc:title
GEneral Multilingual Environmental Thesaurus
dcterms:alternative
GEMET
dc:subject
[http://www2.ulcc.ac.uk/unesco/concept/MT_MT_2.55]
SCIENCE.ENVIRONMENTAL SCIENCES AND ENGINEERING
dc:subject
[http://www2.ulcc.ac.uk/unesco/concept/MT_2.60]
SCIENCE.POLLUTION, DISASTERS AND SECURITY
dc:subject
[http://www2.ulcc.ac.uk/unesco/concept/MT_2.65]
SCIENCE.NATURAL RESOURCES
dc:subject
[http://www2.ulcc.ac.uk/unesco/concept/MT_2.75]
SCIENCE.NATURAL SCIENCES
dc:creator
European Topic Centre on Catalogue of Data Sources (ETC/CDS)
dc:creator
European Environment Agency (EEA)
dc:publisher
European Environment Agency (EEA)
dc:contributor
US Environmental Protection Agency (EPA)
dc:description
GEMET was conceived as a "general" thesaurus, aimed to define a common general language, a core of general terminology for the environment
dcterms:created
2005-03-07
dc:type
[http://iaaa.cps.unizar.es/DcType/Concept/236]
TEXT.REFERENCE MATERIALS.ONTOLOGY
dc:format
[http://iaaa.cps.unizar.es/MimeType/Concept/skos]
SKOS
dc:identifier
http://www.eionet.eu.int/GEMET
dc:source
[http://europa.eu/eurovoc]
EUROVOC thesaurus
...
dc:language
en
es
fr
...
dc:relation
[http://www.eionet.europa.eu]
European Environment Information and Observation Network
dc:rights
It can be used whenever there is no commercial profit
iso:language
en
iso:fileIdentifier
http://iaaa.cps.unizar.es/ontologies/GEMET

Figure 2.17: Metadata describing the GEMET thesaurus

Figure 2.17 shows an example of ontology metadata describing the GEMET thesaurus. The RDF metadata is displayed as a hedgehog graph (reinterpretation of RDF triplets: resources, named properties and values). The purpose of these metadata is not only to simplify discovery, but also to identify which ontologies are useful for a specific task in a peer to peer communication (e.g., ontologies that cover a restricted geographical area or about a specific theme).

Resource	Label	Obligation	Cardinality	Description
http://purl.org/dc/elements/1.1/title	Name	Mandatory	Unbounded	A name given to the ontology
http://purl.org/dc/terms/alternative	Short name	Mandatory	Unbounded	Any form of the title used as a substitute or alternative to the formal title of the ontology
http://purl.org/dc/elements/1.1/creator	Creator	Mandatory	Unbounded	An entity primarily responsible for making the content of the ontology
http://purl.org/dc/elements/1.1/subject	Subject	Mandatory	Unbounded	The topic of the content of the ontology
http://iaaa.cps.unizar.es/iaaaterms/AppliedIn	Applied in	Mandatory	Unbounded	Field in which the ontology can be used
http://purl.org/dc/elements/1.1/description	Description	Optional	Unbounded	An account of the content of the ontology
http://purl.org/dc/elements/1.1/publisher	Publisher	Optional	Unbounded	An entity responsible for making the ontology available
http://purl.org/dc/elements/1.1/contributor	Contributor	Optional	Unbounded	An entity responsible for making contributions to the content of the ontology
http://purl.org/dc/terms/created	Date of creation	Mandatory	Unbounded	Date of creation of the ontology
http://purl.org/dc/terms/issued	Date of publication	Optional	Unbounded	Date of formal issuance (e.g., publication) of the ontology
http://purl.org/dc/terms/modified	Date of modification	Optional	Unbounded	Date on which the ontology was changed
http://purl.org/dc/elements/1.1/type	Type	Mandatory	Unbounded	The nature or genre of the content of the ontology
http://purl.org/dc/elements/1.1/format	Format	Optional	Unbounded	The physical or digital manifestation of the resource
http://purl.org/dc/elements/1.1/identifier	Ontology identifier	Mandatory	Unbounded	An unambiguous reference to the ontology within a given context
http://purl.org/dc/elements/1.1/source	Source	Optional	Unbounded	A reference to a resource from which the present ontology is derived
http://purl.org/dc/elements/1.1/language	Ontology language	Mandatory	Unbounded	A language of the intellectual content of the ontology
http://purl.org/dc/elements/1.1/relation	Relation	Optional	Unbounded	A reference to a related ontology or resource
http://purl.org/dc/terms/conformsTo	Conforms to	Optional	Unbounded	A reference to an established standard to which the ontology conforms
http://purl.org/dc/terms/isVersionOf	Is version of	Optional	Unbounded	The described resource is a version, edition, or adaptation of the referenced ontology. Changes in version imply substantive changes in content rather than differences in format
http://purl.org/dc/terms/isReplacedBy	Is replaced by	Optional	Unbounded	The described ontology is supplanted, displaced, or superseded by the referenced resource
http://purl.org/dc/terms/replaces	Replaces	Optional	Unbounded	The described ontology supplants, displaces, or supersedes the referenced resource

Table 2.1 – continue on next page

Resource	Label	Obligation	Cardinality	Description
http://purl.org/dc/terms/hasVersion	Has version	Optional	Unbounded	The described ontology has a version, edition, or adaptation, namely, the referenced resource
http://purl.org/dc/elements/1.1/coverage	Coverage	Optional	Unbounded	Place covered by the ontology (if it is the case)
http://purl.org/dc/terms/spatial	Spatial characteristics	Optional	Unbounded	Spatial characteristics of the intellectual content of the ontology
http://purl.org/dc/terms/temporal	Temporal	Optional	Unbounded	Temporal characteristics of the intellectual content of the ontology
http://purl.org/dc/elements/1.1/rights	Rights	Mandatory	Unbounded	Information about rights held in and over the ontology
http://purl.org/dc/terms/accessRights	Access rights	Optional	Unbounded	Information about who can access the ontology or an indication of its security status
http://purl.org/dc/terms/license	License	Optional	Unbounded	A legal document giving official permission to do something with the ontology
http://purl.org/dc/terms/audience	Audience	Optional	Unbounded	A class of entity for whom the ontology is intended or useful
http://purl.org/dc/terms/mediator	Mediator	Optional	Unbounded	A class of entity that mediates access to the ontology and for whom the ontology is intended or useful
http://www.isotc211.org/19115/MD_Metadata.dateStamp	Metadata creation date	Mandatory	Unbounded	Date in which the metadata has been created
http://www.isotc211.org/19115/MD_Metadata.contact	Metadata point of contact	Mandatory	Unbounded	Person who has created the metadata
http://www.isotc211.org/19115/MD_Metadata/language	Metadata language	Mandatory	Unbounded	Language used for documenting the metadata record
http://www.isotc211.org/19115/MD_Metadata/fileIdentifier	Metadata identifier	Mandatory	Unbounded	Unique identifier for the metadata record

Table 2.1: Terminological ontology metadata application profile

2.4.3 Representation of mappings

As described in section 2.3.2, the most developed representation model for terminological ontologies is SKOS-Mapping. However, it lacks some necessary characteristics such as the possibility to store the liability of the mappings when they are generated by an automatic system, and the representation of the inverse relationships of the mappings (given a concept, to know which other concepts consider it as equivalent according to certain type of mapping function).

The first step to define the mapping representation format was to select an appropriate terminology for the elements that are needed to be represented. In this area, the nomenclature used is quite heterogeneous. Each standard and model that takes into account mappings needs use its own. For example, an exact mapping is described in ISO and BS standards as *exact equivalence*; but in SKOS Mapping is called *exact match* or *equivalent concept* depending on the version; and in the Getty Art & Architecture thesaurus [99] it is represented using mathematical notation ("=" symbol).

The selected notation has been the one used in the BS-8723 standard to describe the mapping needs. BS-8723 standard describes the structure of a multilingual thesaurus that allows representing the structure of concepts, the possible types of relations between them, the term structure of the concept labels and the types of relations between the elements. Although it does not propose a representation model for relations between thesauri, it reviews the mapping requirements.

Figure 2.18 proposes a representation for a mapping that is based on the notation proposed in BS-8723, but it has been adapted to be used in mappings between terminological ontologies different from thesauri.

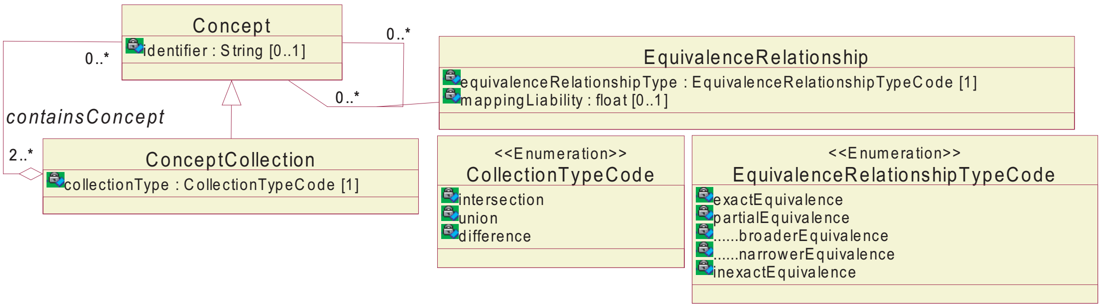


Figure 2.18: Proposed Mapping Model

The *Concept* class shown in the model is equivalent to the *ThesaurusConcept* used in the BS standard to represent each concept in a thesaurus. Its name has been generalized to use it in terminological models different from thesauri, and it can be identified with the *skos:Concept* defined in SKOS where the identifier field is the URI of the *skos:Concept*. The mapping model adds to the *Concept* class the *EquivalenceRelationship* to indicate the equivalence between two concepts of different terminological ontologies. The type of equivalence (exact, inexact or partial) is described by the *EquivalenceRelationshipType*. The way in which this property is defined is based on the one used in the BS-8723 model to represent relations between concepts. It facilitates the creation of a parallel hierarchical vocabulary of types of relationships for *EquivalenceRelationshipType* with specializations of the basic mapping relationships. The hierarchy of mapping relations could have been included in the model as different classes inheriting from *EquivalenceRelationshipType*. However, it would create the need of defining extensions of the model each time a new relationship is used. The other property of an *EquivalenceRelationship* is the *mappingLiability*, which contains, if it is required, the quality of the defined mapping (value between 1 and 100).

As commented previously, the representation of composed concepts is needed to be able to provide the same expressivity as the one in the search paradigm. In the proposed model, the *ConceptCollection* class is used to define composed concepts. It is a specialization of the *Concept* class that aggregates several concepts through a collection type such as *intersection*, *union* or *difference*. The possible composition values are indicated as a controlled list in an equivalent way to the types of equivalence relationships. This facilitates the addition or elimination of different composition types, allowing the customization to the needs of each system. Since *ConceptCollection* extends *Concept*, a *ConceptCollection* can be part of another one providing a constructor for the nesting of several levels of composed concepts, e.g. *(A intersection B intersection (C union (D intersection F))) exact equivalent to G.*

The representation of direct mappings (a single concept related to another one) is quite simple: an *EquivalenceRelationship* must be defined with a specific type between the two desired *Concepts*. The representation of composed concepts increases the complexity of the mapping representation, but this increase is proportional to the composition complexity. For instance, one level of composition requires: the definition of a *ConceptCollection* with a set of *Concept* associated to it; and the relation of the collection with the equivalent *Concept* in the same way as it is done for direct mappings.

The defined mapping model has to be represented in a format suitable for interchange, having into account that the mapping representation has to be independent of the terminological models that it relates, not modifying them in any way. This is required to allow the use of the mapped ontologies independently of the mapping developed between them. As the developed mapping model is inspired on BS-8723 terminology, the first approach for the mapping representation format was to base it on the XML based format of BS-8723. However, basic XML representation is not appropriate for the mapping structure where each mapping relation is independent from the rest and there is not a deep hierarchical structure of properties.

More suitable XML based alternatives are RDF and OWL. They are languages that have been designed in the semantic web context to define relations between any two concepts. The use of any of them has the additional advantage of facilitating their integration with other

RDF/OWL representations for terminological models such as SKOS. The solution adopted has been to define an RDF-Schema with the structure defined in the model using OWL to express the characteristics that cannot be expressed using RDF (e.g., cardinality).

Figure 2.19 presents an example of representation of a direct mapping between two concepts from different thesauri. The *Concept* class is defined as an RDF resource with the identifier field transformed into a URI resource. The use of URIs makes the defined mapping very easy to relate with the original source and destination concepts because modern terminological ontology representation formats use URIs, instead of labels, to identify univocally the defined concepts. That is to say, independently of the format used by the terminological ontologies involved in the mapping (e.g., BS-8723 draft format or SKOS Core), the mapped concepts can be located in the original structures since they share the URIs with the *Concept* classes used in the mapping (they refer to the same entity).

The equivalence relationship shown in figure 2.18 cannot be directly represented using RDF-Schema because it contains attributes. The solution adopted has been to model it with an additional *Equivalence* class that contains the attributes and relates the source and the destination concept of the mapping. Each *Concept* conforming the mapping is related to an *Equivalence* instance through an *equivalenceRelationship* relation. Additionally, each *Equivalence* contains a *mappingOrigin* relation and a *mappingDestination* relation to the source and destination concepts involved in the mapping. Finally, the *Equivalence* class contains as attributes the *equivalenceRelationshipType* with the type of relation between the concepts, and the optional *mappingLiability* to describe the mapping quality. If in a specific application context it is not required for each concept to know which other concepts describe it as an equivalent, the *equivalenceRelationship* of the destination concept and the *mappingOrigin* relation in its associated *Equivalence* instance can be omitted.

The representation of composed concepts and their mapping is described in figure 2.20. The equivalence relationship, instead of relating two concepts, relates a *Concept* with a *ConceptCollection* containing a set of *Concepts* grouped by a composition type (e.g., union, intersection or difference). Thanks to the fact that a *ConceptCollection* is a *Concept*, an *EquivalenceRelationship* can be directly defined between them. This approach is flexible in the sense that allows the definition of more general mappings than the required ones, such as the aggregation of concepts from different terminological ontologies (described by their URIS) in a *ConceptCollections* or the mapping between two *ConceptCollections*.

The set of mappings between two terminological ontologies have to be managed as a whole to be able to integrate them in systems where they are required (e.g., a query expansion system). Each mapping is generated following specific matching criteria being only consistent with respect to the others in the same set. The combination of mappings from different sources without knowing if they are compatibles can lead to misinterpretations in the meaning of the associated concepts.

```
<rdfs:Class rdf:ID="Concept"/>
<rdfs:Class rdf:ID="Equivalence"/>
<rdf:Property rdf:ID="equivalenceRelationship">
    <rdfs:domain rdf:resource="#Concept"/>
    <rdfs:range rdf:resource="#Equivalence"/>
</rdf:Property>
<rdf:Property rdf:ID="equivalenceRelationshipType">
    <rdfs:domain rdf:resource="#Equivalence"/>
    <rdfs:range rdf:resource=
        "http://iaaa.cps.unizar.es/schemas/mapping#equivalenceRelationshipTypeCode"/>
    <owl:cardinality rdf:datatype=
        "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
</rdf:Property>
<rdf:Property rdf:ID="mappingLiability">
    <rdfs:domain rdf:resource="#Equivalence"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <owl:maxCardinality rdf:datatype=
        "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
</rdf:Property>
<rdf:Property rdf:ID="mapping">
    <rdfs:domain rdf:resource="#Equivalence"/>
    <rdfs:range rdf:resource="#Concept"/>
    <owl:cardinality rdf:datatype=
        "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
</rdf:Property>
<rdf:Property rdf:ID="mappingOrigin">
    <rdfs:subPropertyOf rdf:resource="#conceptInvolved"/>
</rdf:Property>
<rdf:Property rdf:ID="mappingDestination">
    <rdfs:subPropertyOf rdf:resource="#conceptInvolved"/>
</rdf:Property>
```

(a) RDF-Schema elements needed for direct mappings

```
<map:Concept rdf:about="http:/T1/HealthCare">
    <map:equivalenceRelationship rdf:nodeID="A28660"/>
</map:Concept>
<map:Equivalence rdf:nodeID="A28660">
    <map:mappingOrigin rdf:resource="http:/T1/HealthCare"/>
    <map:mappingDestination rdf:resource="http:/T2/HealthCare"/>
    <map:equivalenceRelationshipType
        rdf:dataType="map:equivalenceRelationshipTypeCode">
        exactEquivalence<map:equivalenceRelationshipType/>
    <map:mappingLiability rdf:dataType="xsd:float">90<map:mappingLiability/>
</map:Equivalence>
<map:Concept rdf:about="http:/T2/HealthCare">
    <map:equivalenceRelationship rdf:nodeID="A28660"/>
</map:Concept>
```

(b) RDF example of a direct mapping

Figure 2.19: RDF-Schema section required for a direct mapping and example of use

```
<rdfs:Class rdf:ID="ConceptCollection">
    <rdfs:subClassOf rdf:resource="#Concept"/>
</rdfs:Class>
<rdf:Property rdf:ID="containsConcept">
    <rdfs:domain rdf:resource="#ConceptCollection"/>
    <rdfs:range rdf:resource="#Concept"/>
    <owl:minCardinality rdf:datatype=
        "http://www.w3.org/2001/XMLSchema#int">2</owl:minCardinality>
</rdf:Property>
<rdf:Property rdf:ID="collectionType">
    <rdfs:domain rdf:resource="#ConceptCollection"/>
    <rdfs:range rdf:resource=
        "http://iaaa.cps.unizar.es/schemas/mapping#CollectionTypeCode"/>
    <owl:cardinality rdf:datatype=
        "http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
</rdf:Property>
<rdf:Property rdf:ID="containedInCollection">
    <rdfs:domain rdf:resource="#Concept"/>
    <rdfs:range rdf:resource="#ConceptCollection"/>
</rdf:Property>
```

(a) Additional RDF-Schema elements for composed mapping

```
<map:Concept rdf:about="http:/T1/DentalHealthCare">
    <map:equivalenceRelationship rdf:nodeID="A28661"/>
</map:Concept>
<map:Equivalence rdf:nodeID="A28661">
    <map:mappingOrigin rdf:resource="http:/T1/DentalHealthCare"/>
    <map:mappingDestination rdf:nodeID="C3456"/>
    <map:equivalenceRelationshipType
        rdf:dataType="map:equivalenceRelationshipTypeCode">
        exactEquivalence<map:equivalenceRelationshipType/>
    <map:mappingLiability rdf:dataType="xsd:float">90<map:mappingLiability/>
</map:Equivalence>
<map:ConceptCollection rdf:nodeID="C3456">
    <map:collectionType rdf:datatype=
        "map:collectionTypeCode">intersection</map:CollectionType>
    <map:containsConcept rdf:about="http:/T2/HealthCare">
    <map:containsConcept rdf:about="http:/T2/DentalHealth">
    <map:equivalenceRelationship rdf:nodeID="A28661"/>
</map:ConceptCollection>
<map:Concept rdf:about="http:/T2/HealthCare">
    <map:containedInCollection rdf:nodeID="C3456"/>
</map:Concept>
<map:Concept rdf:about="http:/T2/DentalHealth">
    <map:containedInCollection rdf:nodeID="C3456"/>
</map:Concept>
```

(b) RDF example of a composed mapping

Figure 2.20: RDF-Schema section required for a composed mapping and example of use

```
<rdfs:Class rdf:ID="MappingScheme"/> <rdf:Property
rdf:ID="inMappingScheme">
    <rdfs:domain rdf:resource="#Concept"/>
    <rdfs:range rdf:resource="#MappingScheme"/>
</rdf:Property>
```

(a) RDF-Schema elements needed for mapping schemes

```
<map:MappingScheme rdf:about="http:/Mapping1">
    <dc:title> Mapping between AGROVOC and EUROVOC </DC:Title>
    <dc:creator>Javier Lacasta<DC:Creator>
    ...
</map:MappingScheme> <map:Equivalence rdf:nodeID="A28661">
    <map:inMappingScheme rdf:resource=http:/Mapping1>
    ...
</map:Equivalence> <map:ConceptCollection rdf:nodeID="C3456">
    <map:inMappingScheme rdf:resource=http:/Mapping1>
    ...
</map:ConceptCollection>
```

(b) RDF example of a mapping scheme

Figure 2.21: RDF-Schema section required for a mapping scheme and example of use

To be able to identify properly the origin of each mapping, a mapping scheme similar to the one used in SKOS-Core for concepts has been defined (see figure 2.21). Each mapping contains a reference to its associate mapping scheme to facilitate its identification (*inMappingScheme* relation). Given the large amount of mappings that can be defined between two terminological models, a relation between the scheme and all the mappings contained in the scheme (inverse of *inMappingScheme*) would increase greatly the size of the scheme. In addition, since this relation can be deduced from the existent *inMappingScheme* relations if it is needed, it has not been defined.

Terminological ontologies are designed as discrete entities intended to be domain consistent. Mappings between them do not have to affect the integrity of their concepts/relations. Integrating a set of mappings into the files of the original ontologies is discouraged because it would add many relations non relevant in most of the contexts and reduce the generality of the ontology model. On the other hand, the independent storage of mappings allows using the ontologies when no mapping is needed and this facilitates the changes. If a new alignment between the ontologies is provided, it only has to replace the older version of the mapping, without any change in the involved ontologies. Additionally, if one of the ontologies changes, only the mapping has to be updated (it does not affect the other ontology).

2.4.4 Metadata to describe ontology mappings

In the same way that it is required to describe the content of each terminological ontology to identify its purpose, function and origin, each set of mappings between two terminological models must be also properly described to facilitate its identification and simplify its reuse.

The use of metadata to describe mappings enables a user to locate all the mappings generated between two terminological ontologies for a specific use, and it makes possible to compare different approaches defined in different contexts.

In parallel to the work shown in section 2.4.1 for the description of terminological ontologies, a metadata application for ontology mappings based on the Dublin Core Metadata Element Set [88] has been defined (see table 2.2). The metadata profile is similar to the one described in section 2.4.1 for terminological ontologies, but changing some metadata elements and redefining the use of the common ones. The specific metadata fields related to ontology mapping features are: *source of mapping* and *destination of mapping* that are used to identify the ontologies that the mapping relates; *generation process* that is used to indicate the alignment techniques and processes used in the generation of the mapping; and *quality* that it is thought to contain the measure of the average mapping quality obtained in the alignment process.

Resource	Label	Obligation	Cardinality	Description
http://purl.org/dc/elements/1.1/title	Name	Mandatory	Unbounded	A name given to the mapping
http://purl.org/dc/terms/alternative	Short name	Mandatory	Unbounded	Any form of the title used as a substitute or alternative to the formal title of the mapping
http://purl.org/dc/elements/1.1/creator	Creator	Mandatory	Unbounded	An entity primarily responsible for making the content of the mapping
http://purl.org/dc/elements/1.1/description	Description	Optional	Unbounded	An account of the content of the mapping
http://purl.org/dc/elements/1.1/publisher	Publisher	Optional	Unbounded	An entity responsible for making the mapping available
http://purl.org/dc/elements/1.1/contributor	Contributor	Optional	Unbounded	An entity responsible for making contributions to the content of the mapping
http://purl.org/dc/terms/created	Date of creation	Mandatory	Unbounded	Date of creation of the mapping
http://purl.org/dc/terms/issued	Date of publication	Optional	Unbounded	Date of formal issuance (e.g., publication) of the mapping
http://purl.org/dc/terms/modified	Date of modification	Optional	Unbounded	Date on which the mapping was changed
http://purl.org/dc/elements/1.1/type	Type	Mandatory	Unbounded	The nature or genre of the content of the mapping
http://purl.org/dc/elements/1.1/format	Format	Optional	Unbounded	The physical or digital manifestation of the mapping
http://purl.org/dc/elements/1.1/identifier	Mapping identifier	Mandatory	Unbounded	An unambiguous reference to the mapping within a given context
http://purl.org/dc/elements/1.1/source	Source	Optional	Unbounded	A reference to a resource from which the present mapping is derived
http://purl.org/dc/terms/conformsTo	Conforms to	Optional	Unbounded	A reference to an established standard to which the mapping conforms
http://purl.org/dc/terms/isVersionOf	Is version of	Optional	Unbounded	The described mapping is a version, edition, or adaptation of the referenced resource. Changes in version imply substantive changes in content rather than differences in format
http://purl.org/dc/terms/isReplacedBy	Is replaced by	Optional	Unbounded	The described mapping is supplanted, displaced, or superseded by the referenced resource
http://purl.org/dc/terms/replaces	Replaces	Optional	Unbounded	The described mapping supplants, displaces, or supersedes the referenced resource
http://purl.org/dc/terms/hasVersion	Has version	Optional	Unbounded	The described mapping has a version, edition, or adaptation, namely, the referenced resource
http://purl.org/dc/elements/1.1/rights	Rights	Mandatory	Unbounded	Information about rights held in and over the mapping
http://purl.org/dc/terms/accessRights	Access rights	Optional	Unbounded	Information about who can access the mapping or an indication of its security status

Table 2.2 – continue on next page

Resource	**Label**	**Obligation**	**Cardinality**	**Description**
http://purl.org/dc/terms/license	License	Optional	Unbounded	A legal document giving official permission to do something with the mapping
http://purl.org/dc/terms/audience	Audience	Optional	Unbounded	A class of entity for whom the mapping is intended or useful
http://purl.org/dc/terms/mediator	Mediator	Optional	Unbounded	A class of entity that mediates access to the mapping and for whom the mapping is intended or useful
http://iaaa.cps.unizar.es/mapping/source	Source of mapping	Mandatory	Unbounded	An unambiguous reference to the ontology source of the mapping
http://iaaa.cps.unizar.es/mapping/destination	Destination of mapping	Mandatory	Unbounded	An unambiguous reference to the ontology destination of the mapping
http://iaaa.cps.unizar.es/mapping/process	Generation process	Mandatory	Unbounded	Description of the process used to generate the mapping
http://iaaa.cps.unizar.es/mapping/quality	Quality	Mandatory	Unbounded	Measure of the quality of the mapping
http://www.isotc211.org/19115/MD_Metadata.dateStamp	Metadata creation date	Mandatory	Unbounded	Date in which the metadata has been created
http://www.isotc211.org/19115/MD_Metadata.contact	Metadata point of contact	Mandatory	Unbounded	Person who has created the metadata
http://www.isotc211.org/19115/MD_Metadata/language	Metadata language	Mandatory	Unbounded	Language used for documenting the metadata record
http://www.isotc211.org/19115/MD_Metadata/fileIdentifier	Metadata identifier	Mandatory	Unbounded	Unique identifier for the metadata record

Table 2.2: Ontology mapping metadata application profile

2.5 Conclusions

This chapter has reviewed how the concept of terminological ontology is defined in the literature. The representation formats and the possibilities of mapping between related terminological models have been analyzed. As part of the revision, this chapter has introduced some terminology, elements, information structures, methodologies and processes that are used along the rest of the thesis, contextualizing them in the field of ontologies.

It has been shown that the term ontology covers a wide area of knowledge organization models, and there is still no whole consensus about an exact definition of what an ontology is and the models that can be considered as an ontology. This is to the fact that different areas define the ontology concept in different ways, with definitions that range from the philosophical, logical, engineering and information retrieval perspectives among others. The classification of ontologies proposed by Lassila and MacGuinness [127] has been used as a basis to describe the main types of existent ontology models. It has been noted that the main difference between the different ontology models is their capacity to express semantics, being the main difference between them in the degree of formalism and semantics provided rather than in the kind of model.

Two big groups of ontology models has been studied: on the one hand, terminological ontologies, which are simpler and less expressive than full first-order predicate calculus; on the other hand, axiomatized or formal ontologies, which are much more complex to construct but they provide more semantics and their axioms and definitions can support more complex inferences and computations.

Given the size of the needed vocabularies and the cost of constructing formal models, terminological ontologies have been presented as the most suitable alternative for classification and information retrieval. However, it has been remarked that there is a heterogeneity problem in the representation of terminological models that difficult their use by different communities (different groups and organizations have created their own ad-hoc representation models).

With the objective of reducing these heterogeneity problems this chapter has presented a framework for the representation of terminological ontologies that focus on the harmonization of the representation formats of terminological models and the relations between them. As a first step in this harmonization process, this chapter has reviewed the different existent representation approaches and how suitable they are to represent the different used terminological models. From the analyzed models, SKOS has been selected as the most suitable one, but it has adapted to include some additional elements that were required and a suitable metadata application profile to describe the terminological models in such a way that facilitate to the user the identification of the content of each used terminological ontology.

Because information infrastructures usually require single access to heterogeneous data collections classified according to different ontologies, this thesis remarks the need to relate the used ontologies to be able to jump from the terminology used in one system to the terminology used in the other one.

The high cost of developing good quality mappings has shown the need to define a representation model for mapping between terminological models to reuse them when possible. However, given the lack of a suitable one, the required characteristics have been analyzed and a new one has been proposed. The developed format has been based on the thesaurus model and in the mapping nomenclature and definitions described in the BS-8723 standard, but adapted to be applicable to different terminological models.

Chapter 3

Reengineering of terminological ontologies

3.1 Introduction

The use of suitable terminological ontologies is vital in classification and information retrieval to obtain high quality results. For example, in a discovery system, if the resources are annotated using a vocabulary that does not contain all the required terms to describe the collection, the construction of an effective search system becomes much more difficult.

As it has been shown in chapter 2, terminological ontologies such as taxonomies, glossaries or thesauri are widely used in the context of resource classification and information retrieval to provide a uniform way to describe the resource contents and improve discovery systems. Nowadays, there is a great deal of terminological models covering each area of interest. Therefore, when constructing a new system, it is usually possible to find a model (or a small set of models) containing all the required terminology.

However, using existent vocabularies from external sources is not an easy task because there is a great heterogeneity of representation models. Ad-hoc formats are very common, and extensions to the general models are frequently created. Integrating such a variety of models and formats in a common system is not viable. Quite the opposite, they have to be managed homogeneously using a common interchange format. As it has been described in the representation framework proposed in section 2.4, the interchange format selected has been SKOS. Therefore, once the required terminological models have been transformed to it, they can be provided to the different components requiring them without additional work.

The heterogeneity of the used ontology formats increases the cost of translation. Each translation process requires a deep knowledge of the models involved in the transformation and the development of specific translation software. Therefore, if the number of terminological models to translate to SKOS format is high (as happens in the context of this thesis), it is important to define the translation processes as similar as possible, and reuse as much created software as possible to save development efforts. With this cost reduction objective in mind, this chapter proposes in section 3.2 a methodology for the creation of translation systems. It defines the steps that must be followed to create a new translation process, and a software library containing the common functionality required for all the translators (to reduce the creation effort).

In an information retrieval system it is quite common to have to integrate resources classified according to different vocabularies. However, the use of overlapping terminological models increases the difficulty of providing a good quality retrieval system. The solutions adopted range from reclassification of the resources to a single homogeneous terminological vocabulary to the establishment of relationships (mappings) between the different vocabularies.

Section 3.3 advances on this last line of work improving a preexistent alignment mechanism to relate different terminological models through WordNet. The system is based on the disambiguation of the different concepts in each analyzed terminological model by analyzing the structure of the surrounding concepts and WordNet. The system is expanded by allowing the use of upper level ontologies different to WordNet, and by transforming the output of the alignment process into the SKOS-Mapping based format proposed in section 2.4.3.

Reusing existent terminological ontologies allows saving a lot of effort given that the translation cost is much less than the cost of creating a new model from scratch. However, sometimes an ontology with the required vocabulary and semantics does not exist and must be created. In this context, it has to be taken into account that the construction of a new terminological ontology is a long and difficult task that requires a deep knowledge of the domain vocabulary and the nature of the relations holding between them. Therefore, if it has to be done frequently, it is important to reduce this cost as much as possible. One of the approaches commonly used to reduce the creation cost is to reuse existent terminological ontologies and modify them until the desired model is obtained. This approach avoids the picking of part of the required vocabulary and provides a core structure of relations to work with. Section 3.4 describes a method for the construction of ontologies that goes in this direction. It shows a method that extracts from several terminological ontologies (following a multilingual thesaurus structure) those common elements and relations about the desired theme and combines them to generate a new single model. The different views of the same knowledge provided by each ontology are merged to obtain a better definition of the vocabulary and the relation structure.

Finally, it may happen that terminological models are not enough for the desired objectives of a system and a formal model have to be used instead. However, the creation of formal ontologies is much more complex than the creation of terminological ontologies because they include many more defined properties and relations. In this case, a terminological ontology can be used as a base for the construction of a formal model. Section 3.5 shows the analysis of the suitability of a process of this kind by analyzing the structure of a set of terminological models that includes the thesauri generated using the process described in section 3.4.

3.2 Conversion of terminological ontologies to a common representation model

For decades, the evolution of digital libraries has encouraged the use of terminological ontologies describing the vocabulary of an area of knowledge (in the form of taxonomies, classification schemes or thesauri), promoting in that way the creation and diffusion of well-established collections in different domains. However, the lack of standardization has produced a huge variety of incompatible formats that difficult their manage and use. To be able to manage the different ontologies we need to harmonize their representation, transforming all the used terminological ontologies to the same interchange format.

As proposed in section 2.4, the most adequate alternative to represent terminological ontologies is the Simple Knowledge Organization System (SKOS) format [148]. The creation of terminological models following this format must be based on the SKOS model documents[1], but other complementary documents such as the guide created by the Porting Thesauri Task Force PORT[2] (subgroup of the W3Cs Semantic Web Best Practices and Deployment Working Group[3]) to facilitate the creation of terminological models such as thesauri in SKOS can be used.

Once the terminological ontologies have been translated to SKOS, they can be directly provided to the software components described in this thesis, simplifying in this way their management. Additionally, since the selected interchange format is becoming very used for different organizations and research groups, the translated ontologies can be provided to the general public, avoiding other users to perform the transformation.

The translation of a terminological ontology from one format to another one requires the construction of some kind of translation software to perform this task. Information translation processes are usually ad-hoc systems specifically created for a required transformation and discarded after that (two translations processes with different source and target models have very few in common). However, when the number of terminological models to translate is high, creating completely different translation processes for each model is too costly.

This problem has been tackled by developing a general process to help to identify equivalences between the source and target models. It has as objective to harmonize the structure and content of the different translation processes, to reduce the cost of creation a new one, and to make them easier to understand. The process defines the steps to follow, techniques to apply and documentation to define. Additionally, to simplify the construction of the translation

[1] http://www.w3.org/2004/02/skos/

[2] http://www.w3.org/2004/03/thes-tf/mission

[3] http://www.w3.org/2001/sw/BestPractices/

software, a library providing the common functionality required for the different translation processes has been created.

3.2.1 State of the art in translation of terminological ontologies

Translation of models (from one format to another one) is required in many different contexts that need to integrate external information or update existent information. In the context of terminological ontologies there are many different ad-hoc translation approaches to change the format of selected models. Some ad-hoc translations examples are Golbeck et al. [69] that describe the transformation to OWL of the National Cancer Institute Thesaurus[4], and Soualmia et al. [189] that do the same with the Medical Subject Headings[5].

An ad-hoc solution solves the problem of how to transform a specific model into another one, but it is not applicable to other translation processes. If a new transformation is needed, it has to be constructed from scratch without the possibility of using previous works as help or guideline. Therefore, to simplify and reduce the creation cost of new translation processes, some approaches focus on establishing a methodology that can be applied to different situations. Following the methodology, all the translation processes are constructed in a harmonized way, using the same structure, creating the same documentation and following the same steps. They focus on performing in the same way those steps that are common in the different translation processes such as the need to define the source and target model, and the establishing of mappings between them. Internally they are different but follow the same construction patterns. Therefore they are easier to create, understand, and update if it is needed.

An example of methodology to create translation processes is described by van Assem et al. [209]. It provides a guideline and some recommendations about how to perform the transformation of thesauri into RDF/OWL. The first step is an analysis of the thesaurus model and related documentation. Then, it is converted to a basic RDF version and enriched with properties and attributes. Finally, the RDF/OWL meta-model developed is mapped to SKOS. The methodology is applied to create the software needed to translate to RDF/OWL the Medical Subject Headings and WordNet [52]. van Assem et al. [208] describes an upgraded version of this methodology that simplifies the steps to perform to only two: the analysis of the thesaurus model and the mapping to SKOS model. The upgraded methodology is more detailed, and the matching between model properties and relations is represented more formally. Additionally, while in the first process the construction of the software is almost secondary, this latter one includes a specific stage for its construction. This improved methodology is applied to translate to SKOS the previously commented Medical Subject Headings, the Integrated Public

[4] http://www.cancer.gov/cancertopics/terminologyresources

[5] http://www.nlm.nih.gov/mesh/

Sector Vocabulary[6] and the Common Thesaurus for Audiovisual Archives[7].

Another translation methodology is the one proposed by Miles et al. [149]. The proposed translation process consists of three steps: generate the RDF encoding, error checking, and validation and publishing of the encoding on the web. It provides a specific guideline to transform thesauri following standard structure; additionally, it comments how some non standard structure such as groups, themes and supergroups present in GEMET thesaurus could be mapped. The methodology is applied to: the Australian Public Affairs Information Service Thesaurus[8], represented in XML following a model based on the Z39.50 profile for thesaurus; the English Heritage Aircraft Type Thesaurus[9], defined by comma delimited files; and GEMET, described in XML using a non standard model.

3.2.2 Proposed translation process

3.2.2.1 General description of the method

One of the main drawbacks of existent approaches is the lack of harmonized description of the source (and also the target) models. They use the available documentation that can be described in any way, ideally using graphical models but mostly by a list of entities, properties and relations described in natural language. A formal representation of the models simplifies the identification of matchings because it gives a detailed semantic to each element of the model and additionally it facilitates the use of automatic matching techniques (such as the ones described in section 3.3). However, even when a formal model is available, if each time it follows a different notation and/or representation, the way to define the mappings must be redesigned for each particular case. This is a problem in the sense that it implies an additional learning effort each time the alignment is performed or reviewed.

Another relevant problem is concerned with the conversion of data types. The reviewed techniques seem to consider that the data-type of the properties in the source format is the same as in the target one. However, this is not always true. For instance, the identifier of the concepts can be a number in the source format and an URI in the target one. When a data-type changes, a transformation pattern has to be provided.

The last identified problem is the lack of a homogeneous policy for the construction of translation tools. In the analyzed proposals, each translation tool has been created independently from the rest, using different languages and tools. This is a waste of effort. The cost of new developments should be minimized by reusing existent software of previous transformations.

This section proposes a new methodology to define translation processes between terminological ontology models that focus on the problems previously described. The proposed

[6] http://www.esd.org.uk/standards/ipsv/index.html

[7] http://www.beeldengeluid.nl/index.jsp

[8] http://www.nla.gov.au/apais/thesaurus/

[9] http://www.english-heritage.org.uk/thesaurus/aircraft/

methodology consists of the following four steps (see figure 3.1):

- First, the analysis of the source and target format models. The two models have to be properly reviewed and described to simplify the later translation process.
- Second, the matching of each entity, property and relation of the source model with the equivalent one (if it exists) in the target model. The mappings obtained between the elements of the two models have to be described in a suitable way that facilitates the automation of the translation process.
- Third, the development of the translation tool. In this step the objective is to reuse previous translation works and implement as few new software as possible.
- Fourth, the validation of the generated result. Once the translation software has been created, it is needed to check whether the transformation performed is valid.

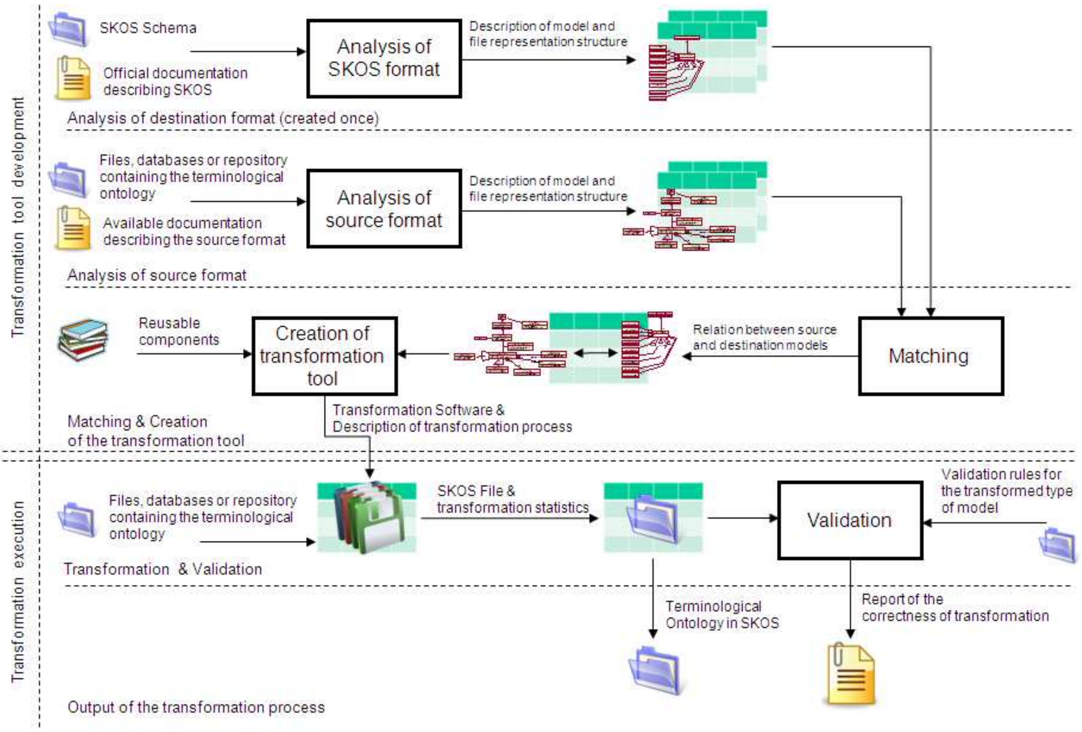

Transformation tool development
SKOS Schema
Official documentation describing SKOS
Analysis of SKOS format
Description of model and file representation structure
Analysis of destination format (created once)
Files, databases or repository containing the terminological ontology
Available documentation describing the source format
Analysis of source format
Description of model and file representation structure
Analysis of source format
Reusable components
Creation of transformation tool
Relation between source and destination models
Matching
Matching & Creation of the transformation tool
Transformation Software & Description of transformation process
Transformation execution
Files, databases or repository containing the terminological ontology
SKOS File & transformation statistics
Validation
Validation rules for the transformed type of model
Transformation & Validation
Terminological Ontology in SKOS
Report of the correctness of transformation
Output of the transformation process

Figure 3.1: Process to develop SKOS translation tools

3.2.2.2 Analysis of the source and target formats

The first step to translate a terminological ontology into SKOS format is to understand deeply the structure of the source and target models to be able to relate them. In the developed work, the different representation formats for the source models that have been found can be grouped in the following categories:

- Databases: The terminological ontology is stored in a database where the different entity types are represented as tables, the concepts as elements stored in a table and the relationships are managed as foreign keys between elements of the table model.
- XML files: The content ontology is described in XML following a specific XML schema. Other approaches use RDF instead of plain XML, and use an RDF-Schema to describe the model (e.g., SKOS).
- HTML files: Some terminological ontologies are only accessible on the web, therefore must be downloaded as HTML files which have to be processed to extract the elements and structure of the terminological ontology.
- Plain text: The most basic representation is directly using text, where the structure has to be deduced by the documentation. For example, the NT relationship can be indicated by the use of tabulation between two different lines of the text.

The main problem found to process the source formats has been the heterogeneity in their structure and the low quality of the documentation. In many cases, there was no documentation and the structure of the model had to be deduced from the data. In other cases, textual descriptions of the model existed, but they were full of ambiguities and inconsistencies. Only one of all the translated models provided a graphical representation of its model.

Therefore, in order to facilitate the automation of the translation process, each element, property and relation has to be identified and described using a common conceptual schema language (or common description language).

On the one hand, to perform the translation it is required to understand the structure and purpose of each element contained in analyzed model (source and target). This can only be done if it is properly described. On the other hand, to be able to access to the content of the analyzed model (source and target), the storage organization structure (e.g., files or databases) of has also to be perfectly known (e.g., name and format of the files, part of the model they contain . . .). As in the previous case, this is only possible if it has been properly described.

To facilitate the use and management of these model descriptions (content and structure), they have been harmonized by using specific models created for this purpose.

The model created for the description of the content structure is shown in figure 3.2. In this model, the main element to take into account is the *Entity* class. Each element of the analyzed model is registered as a new *Entity* with an identifier, a description, and its cardinality. The different *Entities* of the same model are related (through the *relation* relationship) to indicate which structural elements are contained inside the others. Additionally, the entities of the same model are grouped as a terminological *Model* through the *composedOf* relation. The *Model* class contains the name and a description of the format model of the ontology to translate to facilitate its identification.

As it can be seen in figure 3.2, two different subtypes of entities have been identified. On the one hand, the *Registry* class is used to represent those entities that are information containers of collections of other entities; that is, they provide the organization of the ontology content. On the other hand, the *Field* class is used to represent those entities in the model that really contain the data of the terminological ontology. The type of the contained value is represented through the *ValueType* class that contains the type name and the range of values. Each type used in the analyzed terminological ontology is created once and related to the elements in the model that use them through the *contentType* relationship. Additionally, the *Field* class contains a *type* element that indicates the structural purpose of the *Field*. In this context, two purposes have been considered: as *properties* that contain the value associated with a relation; and as *attributes* whose value qualifies the relation itself.

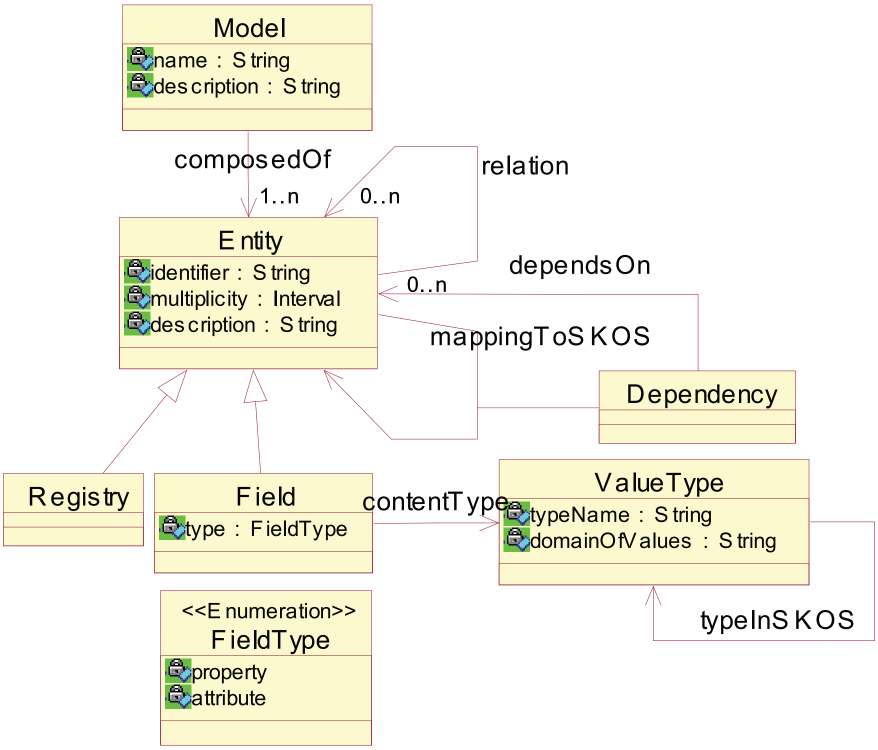


Figure 3.2: Meta model for describing the source, target model and translation rules

The model created for the description of the storage organization structure is shown in figure 3.3. This schema relates a terminological model (*Model* class of figure 3.2) with the storage items used to store it. These storage items are described through the *Resource* class. Its objective is to provide enough information to identify each storage item used for a model. The defined fields of the *resource* class are detailed in table 3.1. The fields included in this class are based on a subset of the core elements of the Dublin Core metadata standard [88].

The storage resources can be divided into two groups: those containing a section or the whole analyzed model, and those describing the structure of the elements in the previous category (e.g., an XML schema of the XML files containing the model). With respect to the first group, to indicate which model elements are stored in each resource, the *source* relation between the *entity* class of figure 3.2 and the *resource* class has been defined. With respect to the second group, to indicate in which *resource* the structure of each *entity* is described, the *descriptiveModel* relation is provided. Additionally, it has been needed to indicate the relation between the *resources* containing the terminological model and the *resources* describing their structure. This has been done with the *relation* relationship shown in figure 3.3. To indicate the type of relation between these resources the *relation* relationship is qualified with a description through the *relationDescription* class.

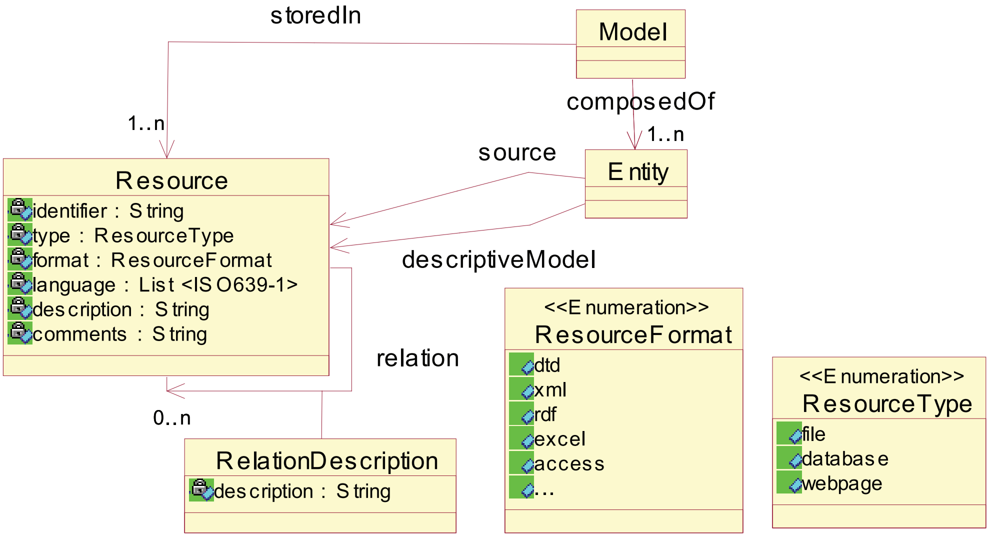


Figure 3.3: Meta model for describing the storage characteristics

3.2.2.3 Mapping between source and target model

Having completely described the source and target models, the next step in the translation process consists in matching them (classes, properties and relations).

In a first step, the models of source and target formats are manually matched. The matching is performed by analyzing both models and the descriptions of the format structure created

ID	DESCRIPTION
Identifer	Name/URL of a resource containing part of the terminological ontology
Type	Type of resource (e.g., file, database, web page)
Format	Specific structure of the resource, (e.g., SKOS, OWL, HTML, Access 2000
Language	Languages in which the content of the resource has been created. For example, it indicates that the tags used in XML labels are French based or that the comments included in the file are in English.
Description	Small description of the subsection of the model contained in the resource
Comments	Additional relevant information about the content

Table 3.1: Fields in the *Resource* class

following the model described in figure 3.2 to find equivalences.

Figure 3.4 shows as example of matching the established between a traditional thesaurus model and the SKOS model (represented both of them using UML notation). As it can be observed in the figure, the following transformations are applied:

- Each *term* in the source model that is selected to be used for classification (it has not a USE relationship with another one) is translated into a *skos:concept*.
- The URI required for each *skos:concept* is generated by converting the *term* value into a URI through the addition of an *http://* prefix.
- The *translation* instances related to the previous *term* are transformed into *skos:prefLabel* of the newly generated *skos:concept*.
- The *terms* related to the converted one through an USE relationship are converted into *skos:altLabel* of the generated *skos:concept*.
- The *description* of a term (related through the *SN* relationship) is converted into the *skos:definition* of the generated *skos:concept*.
- The *RT* relationships between *terms* are converted into *skos:related* relations between the corresponding *skos:concept*(s). With respect to the *BT* and *NT* relationships they are converted into *skos:broader* and *skos:narrower* relationships respectively.
- A model in SKOS format requires of a *skos:conceptScheme*. Because nothing equivalent exists in the source model a *skos:conceptScheme* is created using an URI based on the available information of the source model.
- The *skos:concepts* whose source *term* is marked as *TT* are included in the *skos: conceptScheme* concept scheme as top terms (*hasTopConcept* relationship).

The obtained matchings are stored in the model described in figure 3.2 to facilitate its latter processing. They are represented through the *mappingToSKOS* and the *typeInSKOS* relationships.

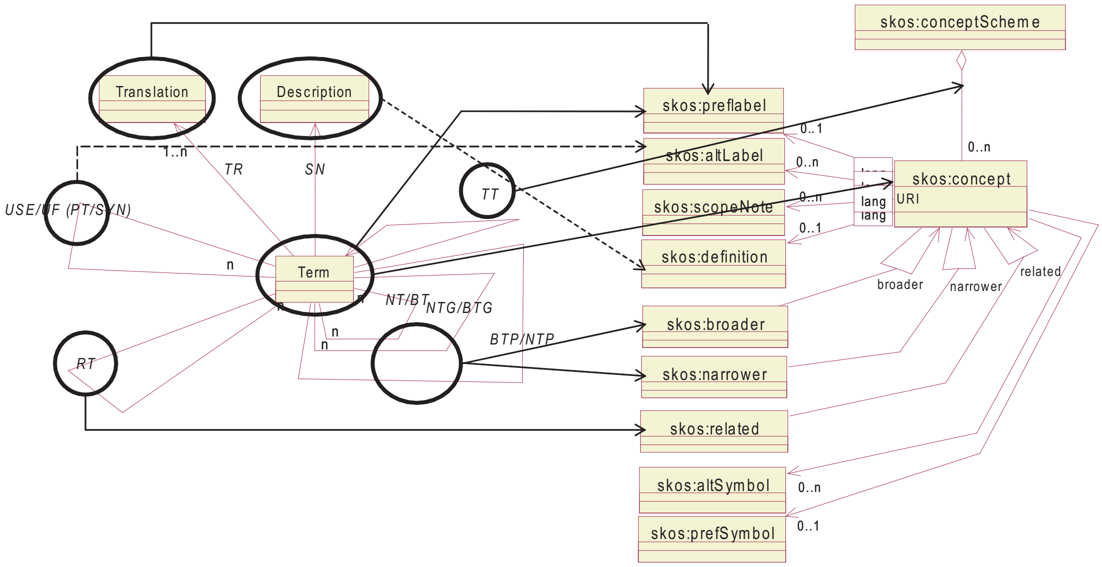


Figure 3.4: Mappings between a traditional thesaurus model and the SKOS-Core model

The *mappingToSKOS* relationship is used to establish an equivalence relationship between an *entity* of the source model and another one of the target model (SKOS model). In general the conversion is direct (an *entity* is converted in another one). However, in some situations the conversion is more complex and depends on other elements in the model. For those cases, the *mappingToSKOS* relationship can be qualified with other *entities* through the *dependsOn* relationship.

In addition to the transformations in the structure (conversion of *entities*), the types of the content must be also converted if they are different. This task is performed by the *typeInSKOS* relationship, which describes the equivalences in the types of the source and target model. In the translation process described in the following steps, for each *typeInSKOS* relationship, a procedure to transform the data from the source type to the destination type has to be provided. For example, in the matching example described in figure 3.4. the label of a *term* has to be transformed into an URI, i.e., a transformation method from a type string into a type URI is required (it is done by adding an *http://* prefix to the string).

3.2.2.4 Creation of the translation tool

The uniform representation of the source format, the target format and the transformation rules simplifies the management of different translation requirements, and it facilitates the creation and automation of a new translation process.

The creation of a new translation tool may not seem a very costly process because the structure of source and target models is usually small. However, if lots of translations are required, the cost of creating a new translator becomes a factor to take into account. The solution adopted to reduce the needs of new software has been to define an architectural pattern based on harmonized description of the models that allow reusing a set of common elements in the different translation processes. Figures 3.5 and 3.6 show this architectural pattern with different levels of detail.

Figure 3.5 shows that each translator tool consists of three different components: a reader of the source format, a matcher of the source model to the SKOS model, and a writer of the SKOS model to RDF. As the target format is always SKOS, the writer component is directly reusable in all the different translation processes. On the other hand, the reader and the matcher have to be adapted to each specific translation, but they share a common part that can be reused.

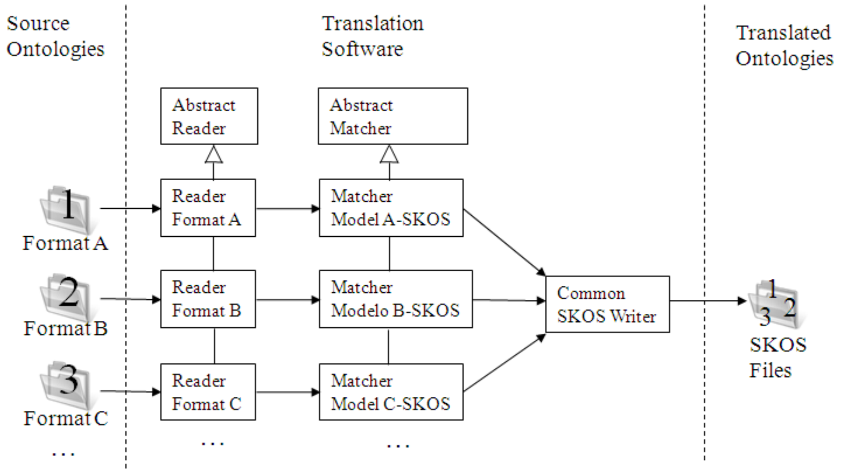


Figure 3.5: Architectural pattern for the format translator

Additionally, figure 3.6 shows a detailed view of this architectural pattern. On the one hand, it provides a set of common elements that can be reused in different translation processes. On the other hand, it defines a set of abstract components (and the required methods) that each new translator tool has to implement with the non-reusable functionality required for the translation. Using the components provided by the library the construction of a new translation tool is reduced to the following two steps: definition of a new reader for the source format that implements the *AbstractReader* class, and creation of a matcher that implements the abstract functions provided by the *AbstractMatcher*.

The translation components described in figure 3.6 use a triples based model <element, property, value> as a common model for processing. In this context, the reader task is to retrieve the entire source model content, construct the triples model required for the rest of the translator components (specially the matcher) and return it through the *getSourceTriples*

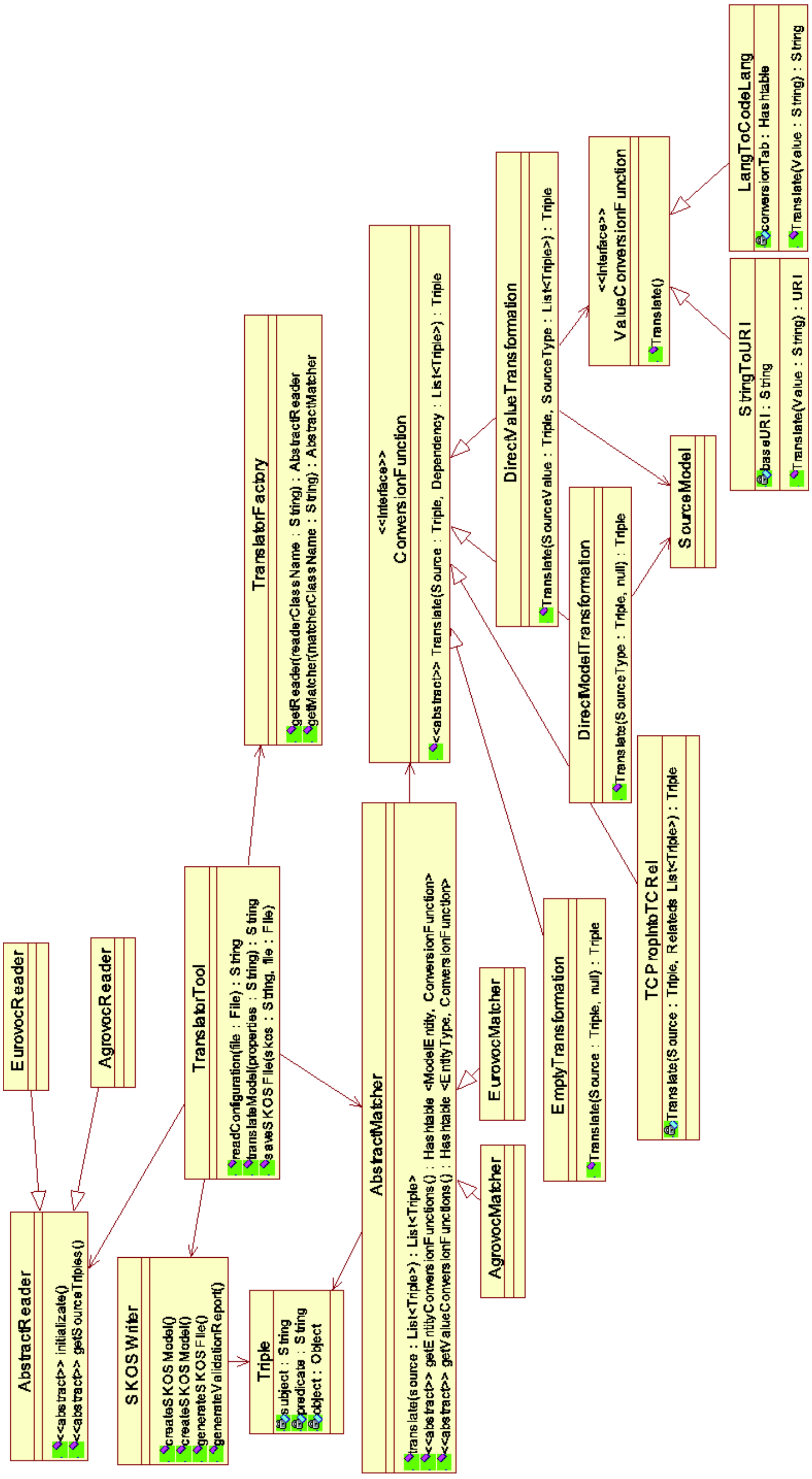

LangToCodeLang
conversionTab : Hashtable
Translate(value : String) : String
StringToURI
baseURI : String
Translate(value : String) : URI
<<Interface>>
ValueConversionFunction
Translate()
DirectValueTransformation
Translate(SourceValue : Triple, SourceType : List<Triple>) : Triple
SourceModel
<<Interface>>
ConversionFunction
<<abstract>> Translate(Source : Triple, Dependency : List<Triple>) : Triple
DirectModelTransformation
Translate(SourceType : Triple, null) : Triple
TranslatorFactory
getReader(readerClassName : String) : AbstractReader
getMatcher(matcherClassName : String) : AbstractMatcher
TCPropIntoTCRel
Translate(Source : Triple, Relateds List<Triple>) : Triple
EmptyTransformation
Translate(Source : Triple, null) : Triple
EurovocMatcher
AgrovocMatcher
AbstractMatcher
translate(source : List<Triple>) : List<Triple>
<<abstract>> getEntityConversionFunctions() : Hashtable <ModelEntity, ConversionFunction>
<<abstract>> getValueConversionFunctions() : Hashtable <EntityType, ConversionFunction>
TranslatorTool
readConfiguration(file : File) : String
translateModel(properties : String) : String
saveSKOSFile(skos : String, file : File)
EurovocReader
AgrovocReader
AbstractReader
<<abstract>> initializate()
<<abstract>> getSourceTriples()
SKOSWriter
createSKOSModel()
createSKOSModel()
generateSKOSFile()
generateValidationReport()
Triple
subject : String
predicate : String
object : Object

Figure 3.6: An architectural pattern for the design of translation tools

method. This component is different in each translation process because it has to access to different source format in each translation. The definition of the source format created previously facilitates the work of the programmer to implement the reader.

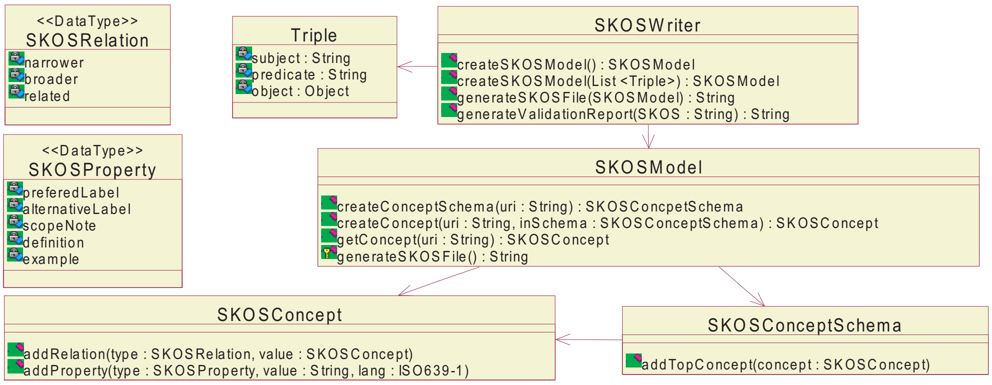
The generated triple model contains the information about the structure of the source model and about their content. Each entity in the source model (registry, field, or attribute) is marked with a unique identifier, and its type. For example, a narrower relationship that in the description model (figure 3.2) is labeled as type "*CONCEPT.RECORD.NT*" would be identified with the tuple <ID@1546, *type*, CONCEPT.RECORD.NT>. In this context, if two concepts are identified as ID@196 and ID@15, the triple <ID@196, ID@1546, ID@15> indicates a narrower relation between them.

The purpose of the matcher is to take the set of triples provided by the reader according to the source model and transform it into another set of tipples according to the target model (SKOS model). This matcher can be implemented manually, analyzing the mappings between the source and target model, and creating the required translation code. However, the harmonization of the source model, the target model, and the matchings between them (described through the model shown in figure 3.2) facilitates the automation of the matcher.

The most usual matchings between the models are 1:1 equivalences; that is to say, an entity in the source model is converted into one entity in the SKOS model. The transformation required in these cases is to change the type of the entity and to perform a transformation in the value; e.g., a string may have to be converted into a URI. In addition to these 1:1 equivalences, some models have very specific situations where the translation of an entity depends on the value of another one (N:1 equivalences). Fortunately, the kind of dependencies that have been found have not been very complex (the most complex one has been to translate a property value into a relation of the concept associated to the property) and it has been possible to accommodate them to the general system.

The matcher performs the transformation of the triples by applying a conversion function for each type of element and type of value. For general transformations, the conversion functions *DirectModelTransformation* and *DirectValueTransformation* shown in figure 3.6 are used. These translators use the descriptions of the mappings between the source and target models created previously (figure 3.2) to perform the translation. For more specific translations, adhoc translators must be developed. The outcome of these translations depends on additional entities of the source model that have to be provided to the translator to perform its task. In addition to the changes in the structure, the vales of the model have to be sometimes converted (changes in data types). These conversions are managed through the creation of *ValueConversionFunctions* that indicate how to translate the value associated to an element into the type required in the target model. For instance, a string may have to be converted into a URI by adding to it an http prefix.

The output result of the matcher is a set of triples following the SKOS model. The task of the writer is to take these triples and generate from them the target SKOS file. Thanks to the fact that the output of the matcher is always a set of triples following the SKOS model, the writer can be reused in all the constructed translation tools. Figure 3.7 describes the writer structure. The set of triples provided as input are used to fill the SKOS-RDF model constructed by the *SKOSModel* class through the use of a library for RDF management such as Jena.



<<DataType>>
SKOSRelation
narrower
broader
related
Triple
subject : String
predicate : String
object : Object
SKOSWriter
createSKOSModel() : SKOSModel
createSKOSModel(List <Triple>) : SKOSModel
generateSKOSFile(SKOSModel) : String
generateValidationReport(SKOS : String) : String
<<DataType>>
SKOSProperty
preferedLabel
alternativeLabel
scopeNote
definition
example
SKOSModel
createConceptSchema(uri : String) : SKOSConcpetSchema
createConcept(uri : String, inSchema : SKOSConceptSchema) : SKOSConcept
getConcept(uri : String) : SKOSConcept
generateSKOSFile() : String
SKOSConcept
addRelation(type : SKOSRelation, value : SKOSConcept)
addProperty(type : SKOSProperty, value : String, lang : ISO639-1)
SKOSConceptSchema
addTopConcept(concept : SKOSConcept)

Figure 3.7: Structure of the writer component

3.2.2.5 Validation of the translated terminological ontology

Once the target SKOS file has been generated, its structure must be checked to verify whether it is correct (no errors have been introduced by the translator tool) and consistent (it follows the terminological ontology model that it is suppose to follow). This verification can be manually done by reviewing the obtained model. However, it may require quite a lot of time and effort.

To facilitate the validation of the correctness, an implementation of the components proposed in the previous step may also provide different statistics. Specifically the following data would be necessary: the number of concepts, number of relations of each type, and number and type of the properties. In this context, given the heterogeneity of the source models, some of the source element may be lost in the translation (for these, no conversion function is defined) due to they do not have correspondence in the target SKOS model. In general, this is not a critical issue, because it only affects extensions that are not vital in the model. However, it is important to take into account the elements that are lost in the translation to be able to validate that their lost is not a mistake. To facilitate this task, a report with the elements from the source model that have not equivalence in the target one is generated.

With respect to the validation of the consistency of the generated model, for each different type of terminological model generated a set of rules indicating the structural conditions to validate must be defined. For example, to validate that a thesaurus represented in SKOS format is valid the following restrictions have to be checked:

- A single *ConceptScheme* must exist. All the concepts contained in the SKOS file must refer to this *ConceptScheme*.
- Every concept must have a single *broader* concept except when it is a top concept. Top concepts do not have any *broader* concept and must be referenced in the *ConceptScheme* structure.
- Each concept must have one and only one preferred label for each available language. Additionally, that label must be unique along the thesaurus.
- All the relations between concepts must reference to existent concepts (orphan relations are not allowed).
- The structure of *broader/narrower* relationships must not contain cycles.
- The *related* relation is symmetric; therefore, if "A" is *related* with "B", then "B" must be *related* with "A".
- The *broader* relation is the inverse of the *narrower* one; therefore, if "A" is the *broader* of "B", then "B" must be *narrower* of "A".

Each different type of terminological ontology requires their specific set of validation rules. For example, dictionaries do not have broader/narrower relationships but it is mandatory that they have a definition associated to each concept. The application of the validation rules corresponding to the obtained models has been done by ad-hoc software. A specific program has been created for each type of terminological ontology translated to SKOS using the translation process described in this section (glossaries, taxonomies, and thesauri).

Many errors detected in different translation processes are usually caused by mistakes in the translation tool. Finding the problem and fixing the tool corrects all those errors. However, it has been found that it is also quite common to have errors in the terminological ontology source formats. Three types of errors have been detected: syntactic errors in the representation format (e.g., a file with a wrong name), semantic errors in the model (e.g., the format does not fulfill some part of the defined model), and structural problems related to the consistency (e.g., a model is supposed to be a thesaurus but some of the concepts does have several preferred labels). Each of these types of errors is detected at a different stage of the translation process. Syntactic errors are directly found when the translation tool reads the sources. Semantic problems are detected by the matcher. And errors in the general structure of the terminological ontology are detected by the validation process.

3.2.2.6 Management of the translation tool

Usually, the designed process for the creation of the translation tools is expected to produce simple tools that receive as input the source model and provide as output the translated SKOS file. However, this is not possible in all situations. Sometimes, the existence of errors in source models requires an additional preprocessing step to correct them. Other times, a simple manual processing of the source data reduces the complexity of the reader component. For instance, a transformation of the source files from ASCII to UTF-8 simplifies the work of managing the non Latin-1 characters. Another example is to save an Excel file as a text based tabular format to simplify the reader component (reading an excel file requires additional programming that is not needed to access a text file).

These additional steps are vital for a translation tool to perform its work properly. But since they must be done manually, it is quite common to forget them. Therefore, the proposed methodology documents them in such a way that the next time a translation tool has to be used they can be easily identified and applied. The solution proposed is to define for each translation process a flow diagram that shows how the different transformation steps are concatenated. Figure 3.8 shows an example of a flow diagram for an XML based source where tree different steps are performed. First, a transformation of the source files to UTF-8. The second step is the correction of all the errors found in the source (the identification of the errors in the source files is described later in this point). The last step is the execution of the transformation tool that generates the SKOS file.

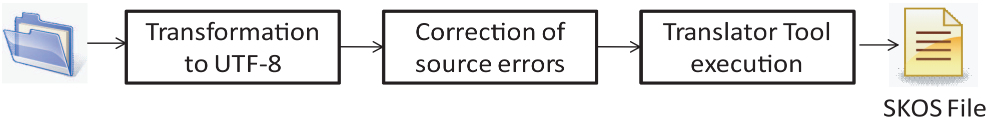


Figure 3.8: Example of transformation flow diagram

In addition to the flow diagram, the set of processes implied in a translation are described according to the model depicted in figure 3.9. In this model, a translation process is represented with the *Translator* class and identified with a name and a description of the source format. A translation consists of a set of translation tasks (*Task* class) that describe each process to perform. The *Task* class contains the following properties to describe a process: a name that identifies the task; the order of execution of the process (e.g., it has to be done in third place); the activity to perform (e.g., execution of the *EurovocTranslator* tool); a description of the task that describes how it has to be done; a description of the input and output of each task (e.g., a review of the files used as input, and the specific directory where they have to be located); and additional information that can be relevant for the process execution (e.g., the description of the different configuration files required to run the translation software).

The errors identified in the source model and the way used to correct them are documented with the *Problem* class described in figure 3.9. Table 3.2 describes the attributes in the *Problem* class: type, description, correction process, and comments.

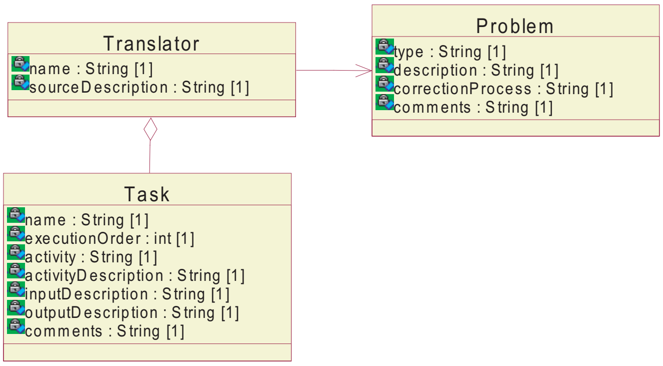


Figure 3.9: Model used to describe the tasks involved in the translation step

ID	Description
Type	It indicates the type of error or issue detected. They can be syntactic, semantic or structural
Description	It describes the cause of the error. For example, there is an XML tag not closed (detailing the name and line where the tag is placed)
Correction process	It indicates what has been done to correct the error. In the previous case, it would be to add a close tag in the proper place
Comments	Additional information about the element that causes the error and how to correct it. For example, external sources used to determine how to correct an error

Table 3.2: Attributes in the *Problem* class used to describe the errors found in the source files

3.2.3 Testing the method

In order to test the feasibility of the methodology proposed, we have created a default implementation for the translation of different terminological ontologies. Table 3.3 shows a review of the translated ontologies indicating their name, the type of model they follow, the format in which they have been originally found, and an approximate number of concepts.

Name	Type of model	Original format	Concepts
GEMET	Thesaurus	Expanded SKOS	6500
AGROVOC	Thesaurus	Access Database	28000
EUROVOC	Thesaurus	XML based	6600
UNESCO	Thesaurus	Access Database	4400
URBISOC	Thesaurus	HTML based	3600
ISOC-GEOGRAFIA[10]	Thesaurus	HTML based	5100
Spanish and French Administrative Units	Taxonomy-Authority File	Excel	45000
ISO-639 Language Code List	Authority File	Formatted text	7600
EPSG Reference systems[11]	Authority File	Access Database	5200
40 ISO-19915 & ISO-19139 lists	Controlled vocabularies	Tables in a PDF file	300
12 CSDGM-FGDC lists	Controlled vocabularies	Tables in a PDF file	100
Inspire Spatial Themes[12]	Controlled Vocabulary	Text document	35

Table 3.3: Review of translated models

Most of the translated terminological ontologies are simple controlled vocabularies used for classification purposes, such as the controlled lists contained in ISO-19115, ISO-19119, and CSDGM-FGDC standards. The rest of them are mainly taxonomies and thesauri such as the Spanish and French administrative units models, AGROVOC, EUROVOC, GEMET, URBISOC[13] or UNESCO thesauri. There are also some authority files such as the ISO-639 language code list, and the set of EPSG codes for coordinate reference systems (including datums, ellipsoids and projections).

As it can be seen in table 3.3, the source formats are quite heterogeneous. There are databases using different table models (UNESCO and AGROVOC thesauri), XML files following different DTD or Schemes (EUROVOC and GEMET thesauri), HTML representations (URBISOC and ISOC-GEOGRAFIA thesauri), Excel files with different degrees of organization (Spanish and French Administrative Units), and text directly extracted from text or even pdf files (ISO-19115 and ISO-639 code lists). This heterogeneity has increased the number of required translation tools because the created ones were not reusable in other translation processes. The only one that it has been possible to reuse has been the created for translating controlled vocabularies. Altogether, nine different translation tools have been required.

An additional factor that has complicated the creation of the translation tools has been the irregularities found in some of the terminological models. The existence of extensions to the basic model or the lack of some mandatory elements have hindered their processing. For example, GEMET and AGROVOC are thesauri but they do not completely follow the structure dictated by the thesaurus standards and include some ad-hoc elements and relationships.

Finally, the last problem tackled has been the existence of errors in the different translated source models. Many of them were not very important, but others were critic because it was not possible to perform a complete transformation if they were not corrected (e.g., syntactic errors). In the transformed models, all the critic errors have been manually corrected. However, it has not been possible to correct some of the non vital consistency problems (e.g., the lack of a preferred label for a concept) due to the lack of information to fix them.

3.3 Matching of terminological ontologies

Terminological ontologies have been traditionally used in information systems as independent entities without relation between them. The proliferation of these models dealing with overlapping areas in different collections has been traditionally an issue that has hindered the retrieval performance of system integrating data from several sources, given that different terminological models use different organization schemas and different terminology to refer to the same concepts.

The solutions usually adopted to tackle this information integration problem are usually the following: the use of a single terminological model in all metadata records of the collections to integrate, the creation of mappings between the models, and the merging of the existent models into a new one.

[13] http://thes.cindoc.csic.es/index_URBA_esp.html

Each solution has its advantages and drawbacks. For example, the use of a single model for all the collections in an information system is simpler, but it requires the modification of all the collection records (to remove the old terminology and include the new ones). On the other hand, the definition of mappings between terminologies does not require the modification of the collection records but it complicates the access to the terminology. With respect to the merging of models, it can be seen as a combination of the other approaches. A single model is used, and there is no need to modify the records (all the source terminology is integrated in the new model). However, it is much more complex to create since it is needed to combine the source models in a coherent way.

This section analyzes the state of the art in matching techniques (section 3.3.1) and studies with detail a matching algorithm and how the obtained matchings can be represented through the representation framework proposed in section 2.4.3. This matching algorithm (section 3.3.2) is based on disambiguation techniques and it will be applied later as part of a query expansion component to improve the search results obtained from heterogeneous collections(see section 5.4). Additionally, section 3.4 proposes a method for the merging of terminological ontologies where a matching technique based on lexical similarity measures will be used to detect the equivalences between different concepts. This matching algorithm is presented together with the different tasks of the merging method instead of showing it in this section to maintain the coherence in the description of the merging process.

3.3.1 State of the art in matching techniques

Along the years, different approaches to solve the problem of identifying the similarities between concepts of different vocabularies and the establishing of equivalence relations between them have been developed. Euzenat and Shvaiko [49], Kalfoglou and Schorlemmer [107] and Rahm and Bernstein [177] remark the following matching techniques as the most relevant:

Techniques based on the analysis of entity names: They match names (classes, attributes . . .) and name descriptions of ontology entities. These *terminological* techniques work directly with strings and some matching algorithms use them to analyze textual element descriptions in natural language to find equivalences. They analyze entities (or instances) in isolation, ignoring their relations with other entities (*element level* techniques) and use only the existent data without using additional elements (*syntactic* techniques). They can be divided into *string-based* and *linguistic* or *language-based* techniques. *String-based* techniques consider terms as sequences of characters and suppose that the more similar the strings, the more likely they denote the same concepts. Cohen et al. [31] and Noy and Musen [166] describe techniques such as string normalization (case, accent, separator removal), equality, edit distance, token and path comparison. *Linguistic* or *language-based* techniques interpret terms as linguistic objects and exploit morphological properties of the entity names to perform linguistic normalization and compare them. Some examples are lemmatization, term extraction and stop-words removal [48, 53].

Techniques based on the analysis of the entities structure: They analyze the structure between the entities (*structure level* techniques). Additionally, since these techniques do not use external elements to work, they are also *syntactic* techniques. They can be subdivided into *internal* and *relational* techniques. *Internal* techniques are based on the use of constraints in the structure of the entities (e.g., types, names or multiplicity of attributes) to find equivalences [177, 33]. *Relational* techniques use the relations between the entities of each ontology (they consider ontologies as a graph) to find the commonality in the structure of relations [140, 105]. They use the similarity of names, types and structure of the relations between the entities. In this context, taxonomic (*is-a*) and mereologic (*part-of*) relations are the most usually used.

Techniques based on the use of external resources: These techniques use external resources to find or improve the matchings (*external* techniques). Usually, the external resources have a linguistic nature (e.g., lexicons, domain specific thesauri or terminologies) and the linguistic relations of the external resources (e.g., synonymy, hyponymy, hypernymy . . .) are used to map the terms [75]. An example of this kind of technique is the work of Aleksovski et al. [5] that match poorly structured resources using pre-existent upper formal ontologies (or formal domain ontologies if the area of interest is narrow enough).

Techniques based on previous alignments: These are also *external* techniques that, instead of using other ontology models to detect matchings, use previous available alignments between ontologies in the same area of knowledge [177]. For example, the work of Rahm et al. [178] proposes to reuse fragments of ontology alignments to detect more easily structural equivalences between the pre-existent matchings and the desired one. These ontology alignment fragments allow detecting subsets in the ontologies to match that are similar to others already matched in previous alignment processes.

Techniques based on semantic interpretation: These are techniques that require some semantic interpretation of the ontology. They usually apply a semantically compliant reasoner based on some formal semantics (model-theoretic semantics) to deduce the correspondences and perform the matching (*semantic* techniques). In this field, two entities are the same if they have the same semantic interpretation. Some examples are the techniques based on propositional satisfiability or description logics reasoning. Propositional satisfiability is based on deciding whether a formula of propositional logic is satisfiable or not. This is used to check exhaustively all possible matchings [67]. Description logics reasoning techniques are based on the use of a description logics language for representing structural meaning and any additional constraints (axioms) from the domain knowledge. The reasoning techniques use the language formalism to deduce the mappings [22].

Techniques based on data analysis: They are based on the use of data instances (i.e., a populated ontology) as input (*extensional* techniques). They use a representative sample of a population and analyze the similarity of the property values of the entity instances to find equivalences and discrepancies [193, 43]. For instance, FCA-Merge [193] uses a Formal Concept Analysis (FCA) based process to analyze the instances and perform the matching. An application of this merging schema is described by Nogueras-Iso et al. [160]. Another example is IF-Map [106] that shows a theory and method for automated ontology mapping based on channel theory, a mathematical theory of semantic information. It formalizes the notion of ontology, ontology morphism and ontology mapping linking them to the formal notions of local logic and logic info-morphism stemming from information-flow theory.

These matching approaches have been broadly used in alignment systems from areas such as schema translation and integration, knowledge representation, machine learning and information retrieval [177]. In these fields, there are many relevant works that use one or several of these matching techniques to generate an alignment between ontologies. For example, Lim et al. [135] apply linguistic and structure-based matchers as part of a data analysis process that deduce concept equivalences. In this same line of work it is SMART [166], a Protégé-2000 [165] plugin that looks for linguistically similar class names, and the structure of relations to establish the mappings. Similarly, PROMPT [167] and Chimaera [143] search for linguistic similarities for classes and attributes, but focusing on human contribution as a vital element in the definition of the mappings. Working with semantic matchers there is the work of Compatangelo and Meisel [33] that use a description logic reasoner to find class equivalences, and linguistic and heuristic inferences to compare attributes. Somewhat related is the CROSI Mapping System [104] which integrates linguistic and semantic matchers to perform the alignment process. A completely different approach is the work of Prasad et al. [175] that analyzes the relations between classes using a Bayesian matcher. To finish, two works based on instances analysis can be highlighted. On the one hand, CAIMAN system [126] considers the concepts in an ontology implicitly represented by the documents assigned to each concept and provides these relations to a machine-learning process to generate the matchings. On the other hand, Doan et al. [43] present a system that uses probabilistic distribution based similarity measures on the ontology instances to find matchings.

3.3.2 Mapping of terminological ontologies to an upper level ontology by means of disambiguation

In the context of this thesis, to be able to relate different terminological ontologies it was decided to use the alignment method described by Nogueras-Iso et al. [163]. This process is focused on relating a terminological ontology with respect to WordNet lexical database and it is similar to the methods described in Sussna [195], Agirre and Rigau [1], and Resnik [180]. However, this process does not require a training corpus to estimate probabilities for calculating the semantic similarity. It identifies the similarity using the thesaurus hierarchical structure as the context to evaluate each particular term.

Following the classification of matching algorithms described in section 3.3.1, this matching algorithm can be considered as a *Relational* technique, because it is based on the analysis of the entities structure using the relations between the concepts in the source ontology and the lexical database. In addition, to be able to match the labels from the ontologies they are also processed using *Linguistic* techniques, such as lemmatization to reduce the terms to their original forms and term extraction to obtain the different words contained in each term. Additionally, this technique can be viewed as an *external* matching process because it has as final objective to use the lexical database as a pivot to relate a set of terminological ontologies between them.

As commented previously, the problem of ontology alignment consists in finding equivalences between concepts from different models. To do so, it is needed to determine for each term which of its possible senses is the used for the analyzed terminological model. In this case, in the same way that the sense of a word in a natural language text can be determined by the context of the word (the other words in the same phrase or paragraph) the sense of a concept in a terminological ontology such a thesaurus can be determined by analyzing the concepts that are related to it (broader and narrowers).

The work proposed in [163] uses this context information to determine which of the senses of WordNet lexical database concepts fits better with the intended meaning of each concept in the source thesaurus. The objective of establishing the mapping between different thesauri and WordNet is to use it as a kernel to unify, at least, the broader concepts included in distinct thesauri. The proposed alignment method can be classified as an unsupervised disambiguation method and applies a heuristic voting algorithm that makes profit of the hierarchical structure of both WordNet and the thesauri. Whereas thesaurus hierarchical structure provides the disambiguation context for terms, the hierarchical structure of WordNet enables the comparison of senses from two related thesaurus terms.

The initial step of the disambiguation process divide the thesaurus into branches (a branch corresponds to a tree composed by a top term and all the descendants in the "broader/narrower" hierarchy). The branch provides the disambiguation context for each term in the branch. Secondly, the disambiguation method finds all the possible WordNet synsets (WordNet is structured in a hierarchy of synsets which represent a set of synonyms or equivalent terms) that may be associated with the terms in a thesaurus branch. If a term is compound (more than one word) and it is not included in WordNet, the senses for each word are extracted. Finally, a voting algorithm where each synset related to a thesaurus term votes for the synsets related to the rest of terms in the branch is applied. This method uses the hierarchical structure of WordNet on the assumption that: "the more similar two senses are, the more hypernyms they share". Given a synset path (i.e., a possible sense) of a term, the voting system compares it with the rest of synset of the other terms in the same branch (i.e., the context). Additionally, in the case of having a compound term, a synset path of a subterm would also vote for the synset paths associated with the rest of subterms of this compound term. For each pair of synset paths, the system counts the number of hypernyms (WordNet synsets) that subsume both of them, giving an accumulated result for the initial synset path. The main factor of this score is the number of subsumers in synset paths (the synset and its ancestors in WordNet). The synset with the highest score for each term is elected as the disambiguated synset.

Table 3.4 shows as disambiguation example the final score of synsets for the branch accident of GEMET thesaurus. For the sake of clarity, some terms and their corresponding synsets have not been shown.

Regarding the score given by one synset path to another, the initial idea was to assign each other the total number of shared hypernyms. For instance, the two synset paths for the term *accident* would assign each other two votes because they share the synsets *event* and *happening.* Let us observe that they would not receive the third vote by the synset *accident* because the depth is different:

- synset path 1: *event*→*happening*→*trouble*→*misfortune*→*mishap*→*accident*
- synset path 2: *event*→*happening*→*accident*

In this algorithm, three criteria have been applied to correct this score. These criteria are slightly related to the aspects that Agirre and Rigau [1] define the conceptual distance (the length of a path of concepts in WordNet, the hierarchy and the density depth). In order to facilitate the understanding of these criteria, they will be explained in parallel with the example in table 3.4 that shows the scores given by synset paths in the branch *accident* to the synset path *event*→*happening*→*trouble*→*misfortune*→*mishap*→*accident* of the term *accident.* The column *sco* shows the final score given by each synset path after applying the three criteria. and the total score for the voted synset is marked on the right of this synset path.

1. Firstly, lower level WordNet concepts (synsets) have longer paths and then, share more sub-hierarchies. Therefore, the number of shared hypernyms (*sub* column in table 3.4) is

Term	Subterm	Synset path	sub	dep	dis	pol	sco
accident							
		event→happening→trouble→misfortune→mishap →accident	**total score = 3.143**				
		event→happening→accident	*it doesn't vote*				
accident→accident source							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	1	4	0.250
		event→happening→accident	2	3	1	4	0.167
	source	*7 synsets without subsumers*					
accident→accident source→oil slick							
		entity→object→film→oil_slick	0	4	2	1	0.000
accident→environmental accident							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	1	4	0.250
		event→happening→accident	2	3	1	4	0.167
	environmental	*2 synsets without subsumers*					
accident→environmental accident→explosion							
		event→happening→discharge→explosion	2	4	2	3	0.083
		act→action→change→change_of_integrity→explosion	0	5	2	3	0.000
		act→action→change→change_of_state→termination →release→plosion	0	7	2	3	0.000
accident→environmental accident→leakage							
		event→happening→movement→change_of_location →flow→discharge→escape	2	7	2	1	0.143
accident→major accident							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	1	4	0.250
		event→happening→accident	2	3	1	4	0.167
	major	*1 synset without subsumers*					
accident→major accident→nuclear accident							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	2	4	0.125
		event→happening→accident	2	3	2	4	0.083
	nuclear	*2 synsets without subsumers*					
accident→major accident→nuclear accident→core meltdown							
	core	*8 synsets without subsumers*					
	meltdown	*no synsets in WordNet*					
accident→traffic accident							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	1	4	0.250
		event→happening→accident	2	3	1	4	0.167
	traffic	*3 synsets without subsumers*					
accident→traffic accident→shipping accident							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	2	4	0.125
		event→happening→accident	2	3	2	4	0.083
	shipping	*2 synsets without subsumers*					
accident→work accident							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	1	4	0.250
		event→happening→accident	2	3	1	4	0.167
	work	*7 synsets without subsumers*					
accident→technological accident							
	accident	event→happening→trouble→misfortune→mishap →accident	6	6	1	4	0.250
		event→happening→accident	2	3	1	4	0.167
	technological	*2 synsets without subsumers*					

Table 3.4: Voting for synset path *event* → *happening* → *trouble* → *misfortune* → *mishap* → *accident* of term accident

divided by the length of the path, i.e. the depth of the WordNet concept. For instance, synset path *event* → *happening* → *trouble* → *misfortune* → *mishap* → *accident* (depth=6) is likely to receive more votes than synset path *event*→*happening*→*accident* (depth=3) if this restriction is not applied. In table 3.4, the depth of every synset path is shown in column *dep*.

2. Secondly, not all the terms in the context should be valued in the same way. The number of votes provided by the synset paths of a term A to a synset path of a term B are divided by the distance between the two terms (A and B) in the thesaurus. For instance, obtaining the scores for the synsets of the term *accident*, the term *environmental accident* is more important than the term *explosion* because it is closer in the hierarchy. In table 3.4, the distance of every synset path is shown in *dis* column.

3. And thirdly, the most polysemic terms in the context vote more times since each one of their senses has the opportunity to vote. The number of votes provided by a synset path is divided by the number of senses of the term to which it belongs. For instance, term *accident source* votes with its nine synset paths, meanwhile term *leakage* only votes with one synset path. In table 3.4, the polysemic value of every synset path is shown in *pol* column.

This method provides an automatic way to match different terminological models with WordNet. The definition of these relations makes possible to use WordNet as a union kernel between the different mapped terminological models that allows jumping between equivalent concepts in different models. However, WordNet cannot be used to relate all kinds of vocabularies because it is quite general and does not contain many specialized terminology. To solve this problem it has been needed to generalize the described process to be able to relate terminological ontologies with lexical models different to WordNet focused on the subject of the models to relate. Any ontology with a suitable structure of concepts could be used as kernel. For example, Dolce [141] has as aim capturing the ontological categories underlying natural language and human commonsense. Another example is EuroWordNet [213], which improves the structure of WordNet with terms and senses from languages different from English.

The other issue of the original disambiguation process is the one related to the input and the output of the disambiguation system. On the one hand, the thesaurus to match has to be provided in an ad-hoc format. This increases the complexity of the system in the sense that the thesauri have to be translated into the required input format with the only purpose of performing the expansion. On the other hand, there is a lack of representation of the defined matchings. The disambiguation system is used as a library that directly provides the obtained matching to the system that integrates it. This makes the approach not reusable because the alignments have to be created in each system independently of the rest even if they are performed over the same terminological models.

The input problem has been resolved by modifying the disambiguation system to accept SKOS files as input. With respect to the output problem, there is the need of generating the obtained matchings in a suitable interchange format that facilitates their generation in a single system and their distribution between the different tools and components requiring them. This issue has been solved by modifying the disambiguation system to establish as output the updated version of SKOS-Mapping described in section 2.3.2. On the one hand, this interchange format facilitates the distribution of the generated mappings along an information infrastructure. On the other hand, as it is based on SKOS (to describe the concepts involved in the relation), the mappings represented with this format can be easily integrated with the terminological ontologies used along this thesis (also represented with SKOS).

Figure 3.10 shows the components architecture of the improved matching systems. The *DisambiguationSystem* has been modified by adding to it the *IF_LexicalDatabaseManager* and the *InputOutputManager*.

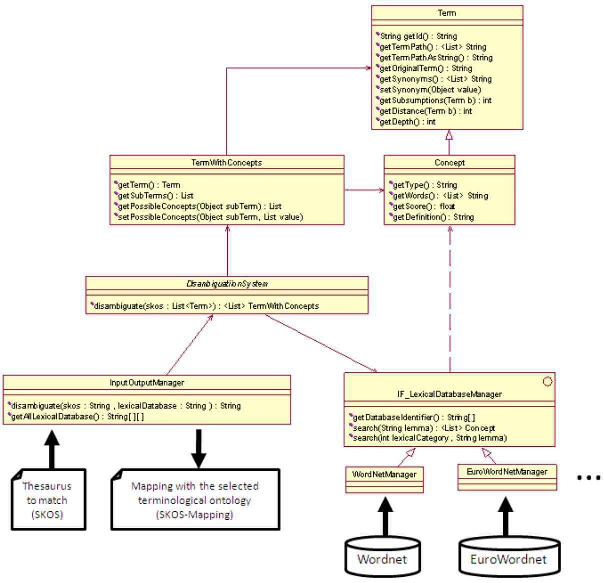

Term
String getId() : String
getTermPath() : <List> String
getTermPathAsString() : String
getOriginalTerm() : String
getSynonyms() : <List> String
setSynonym(Object value)
getSubsumptions(Term b) : int
getDistance(Term b) : int
getDepth() : int
TermWithConcepts
getTerm() : Term
getSubTerms() : List
getPossibleConcepts(Object subTerm) : List
setPossibleConcepts(Object subTerm, List value)
Concept
getType() : String
getWords() : <List> String
getScore() : float
getDefinition() : String
DisambiguationSystem
disambiguate(skos : List<Term>) : <List> TermWithConcepts
InputOutputManager
disambiguate(skos : String , lexicalDatabase : String) : String
getAllLexicalDatabase() : String[ ][ ]
IF_LexicalDatabaseManager
getDatabaseIdentifier() : String[ ]
search(String lemma) : <List> Concept
search(int lexicalCategory , String lemma)
Thesaurus to match (SKOS)
Mapping with the selected terminological ontology (SKOS-Mapping)
WordNetManager
EuroWordNetManager
...
Wordnet
EuroWordnet

Figure 3.10: Architecture of the disambiguation component

The main problem found to facilitate the access to different lexical databases is that each one has a different structure and method of access. This has been solved by creating for each required lexical database a manager that implements the general *IF_LexicalDatabaseManager* interface whose mission is to provide access to the database. In the initialization of the system, one of lexical database is selected (by configuration); then, the corresponding manager is loaded and used to retrieve its content. The interface provide two general search functions to provide access to the different lemmas contained in the selected lexical database.

With respect to the *InputOutputManager*, it performs a dual transformation. On the one hand, it reads the input thesaurus represented with SKOS format and fills the inner structures used to perform the matchings with the selected lexical database. On the other hand, it processes the generated matchings and generates an SKOS-Mapping file.

Conceptually, the changes performed to the disambiguation process, can be seen as the creation of two layers, one under the original disambiguation system and another on top of it. The first one mask the complexity of the lexical databases, and the other one establishes SKOS and SKOS-Mapping as the input and output of the system. However, to be able to isolate the lexical database used trough a common interface and to facilitate the reading of SKOS and the creation of the SKOS-Mapping, it has been needed to modify all the inner structures used for the original disambiguation process described by Nogueras-Iso et al. [163]. The inner processing structures have been generalized to be able to manage data from different lexical databases. The resulting inner data structure consist of the *Term* class, the *Concept* class and the *TermWithConcepts* class. The *Term* class is used to store the information of the terms of each concept in the terminological source model, and it includes methods required for the disambiguation process to obtain information such as the identifier of the term (*getId*), the position in the hierarchy of the source model (*getTermPath*, *getDepth*), the distance with other term (*getDistance*), or the alternative labels of the selected term (*getSynonyms*). The *Concept* class extends the previous one to add additional information required for the concepts of the lexical database such as the type of lexical category of the selected concept (*getType*), its definition *getDefinition* and the score (*getScore*) associated with the term that is being processed (see table 3.4). Finally, the *TermWithConcepts* class stores the matches identified between the terminological terms and the lexical database concepts through the *getPossibleConcepts* method. Additionally, if the processed term is composed it stores the results obtained for each one of the separated sub terms (*getSubTerms*).

The disambiguation component, designed as an independent module, receives an input in SKOS format and returns the disambiguation with respect to the upper-level ontology using the mapping model described in section 2.4.3. Since the disambiguation algorithm cannot assure a 100% exact mapping, they have been marked as inexact with the liability factor showing the probability of equivalence. The mapping with the highest liability may have been marked as exact equivalence. However, since an exact equivalence cannot be assured without a manual revision the mappings are left as inexact. An example of a mapping found with the algorithm used is shown in figure 3.11. There, the concept *3154 (fen)* of GEMET is correctly mapped to the WordNet concept *8763104 (marsh, marshland, fen, fenland)* with a probability of 91.08755%. Also another unrelated mapping is found, but it is given a low probability (8.912453%).

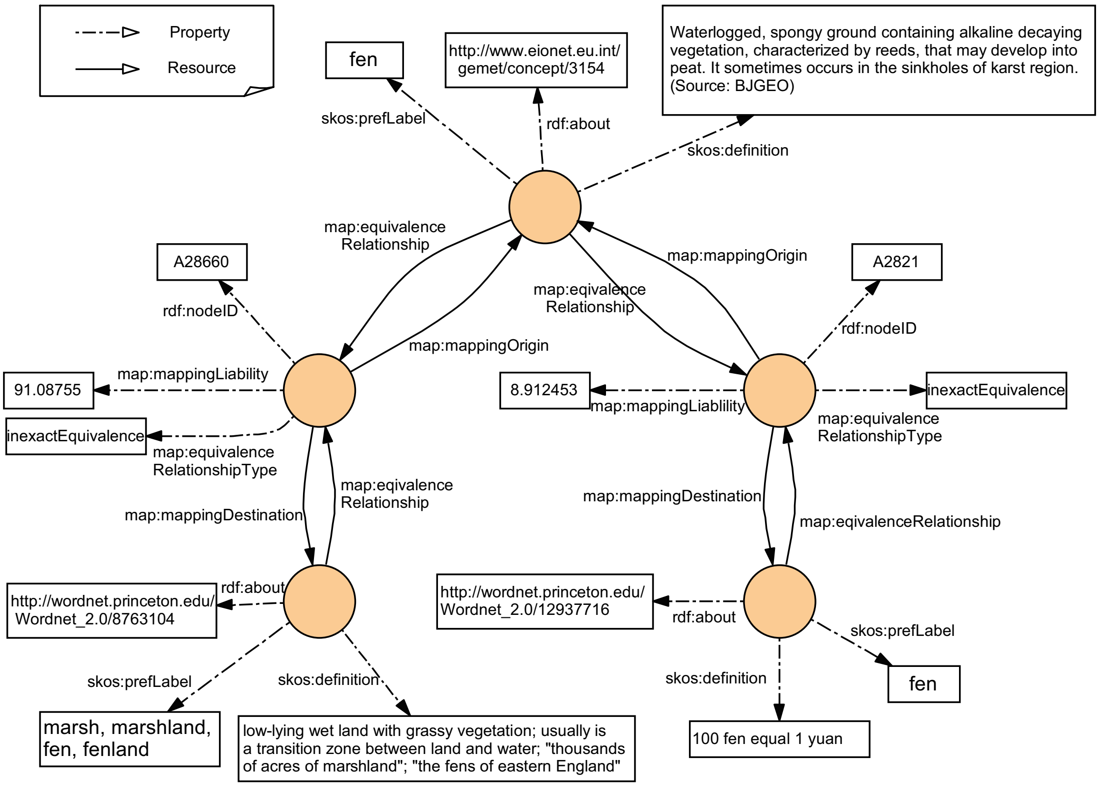

Property
Resource
fen
http://www.eionet.eu.int/gemet/concept/3154
Waterlogged, spongy ground containing alkaline decaying vegetation, characterized by reeds, that may develop into peat. It sometimes occurs in the sinkholes of karst region. (Source: BJGEO)
skos:prefLabel
rdf:about
skos:definition
map:equivalence Relationship
map:mappingOrigin
A28660
A2821
rdf:nodeID
map:eqivalence Relationship
rdf:nodeID
map:mappingOrigin
map:mappingLiability
91.08755
8.912453
map:mappingLiablility
inexactEquivalence
inexactEquivalence
map:equivalence RelationshipType
map:equivalence RelationshipType
map:eqivalence Relationship
map:mappingDestination
map:mappingDestination
map:eqivalenceRelationship
rdf:about
http://wordnet.princeton.edu/Wordnet_2.0/8763104
http://wordnet.princeton.edu/Wordnet_2.0/12937716
rdf:about
skos:prefLabel
skos:prefLabel
skos:definition
skos:definition
fen
marsh, marshland, fen, fenland
low-lying wet land with grassy vegetation; usually is a transition zone between land and water; "thousands of acres of marshland"; "the fens of eastern England"
100 fen equal 1 yuan

Figure 3.11: Mapping example

3.4 Merging of terminological ontologies to create a new model

Reusing existent terminological ontologies is usually a good choice to reduce development efforts. However, it is not always possible to find a terminological model suitable for the required purpose. Existent terminological ontologies may not contain the required vocabulary, or may not provide enough semantics for the required purpose.

If no suitable model exists and a new ontology has to be created, instead of creating it from scratch (it takes a lot of effort), it is a good policy to take as base other existent ontologies and reuse as much as possible from them. Although existent models do not completely fit the needed requirements, part of their vocabulary and relations may be reused in the new model. These elements can provide the core of the new model saving a lot of time and effort in its construction.

To do this, it is important to take into account that each terminological ontology is a specific view of certain knowledge area. Each one provides a partial view of a knowledge area because its content is biased by the application context and purpose for which the ontology has been created. The meaning of a term in certain context can be completely different from the meaning in other contexts. It is said that one of the biggest challenges in information retrieval is the identification of concept meaning in a specific domain of interest. In terminological models this meaning is indicated by a natural language definition and by the context provided by the rest of the model structure (e.g., in a thesaurus the broader/narrower relations).

The use of several terminological ontologies (in this case thesauri) covering the same area of knowledge helps to obtain a more general interpretation of the domain. The different views provided by each terminological model complement each other. If a set of concepts and relations between them can be found in most of the source models, it indicates that the organization of these subsets of knowledge is generally recognized and it is quite stable. Therefore, due to its relevance in different contexts, it should be included in the model to generate.

The main issue to take into account when selecting these relevant knowledge elements is the identification of the set of them that is part of the desired knowledge area. The terminological models selected as source for the merging process contain terminology required by the model to construct. However, they may also contain others terms and relations relevant for other contexts, but not for the required one.

This section describes a process to construct a terminological ontology using other terminological models as base. The objective of the process is to generate a terminological ontology following the thesaurus structure that is focused on a selected theme (that is, a domain model).

3.4.1 Proposed merging process

The objective of the developed process is to generate a thesaurus which contains the main concepts of the required area of knowledge and the relations between them. The process is based on two elements: a glossary of terms about the desired subject (set of required concepts) and a set of cross-domain terminological ontologies with a rich inter-concept structure such as thesauri or taxonomies (simpler models such as glossaries cannot be used due to the lack of term relationships) that contain subsets of terms about the desired theme. The glossary is used as the core of the new model (to focus the result in the desired theme) and it is enriched with the structure of concepts and relations between concepts in the desired area of knowledge through the analysis and comparison of cross-domain models. The structure resulting of such combination is a network of concepts that can be considered as a domain terminological ontology of the area of interest. To obtain the desired thesaurus, this network is processed adjust its structure to the specified by the thesaurus standards.

The obtained thesaurus provides a general view of the knowledge of the desired area. It provides an initial structure of concepts and relations to work with that can be updated and modified according to the desired requirements. Although the generated model is not perfect and it has to be manually refined to adjust it to the application requirements, the cost of doing it is much less than the cost of having to construct it from scratch.

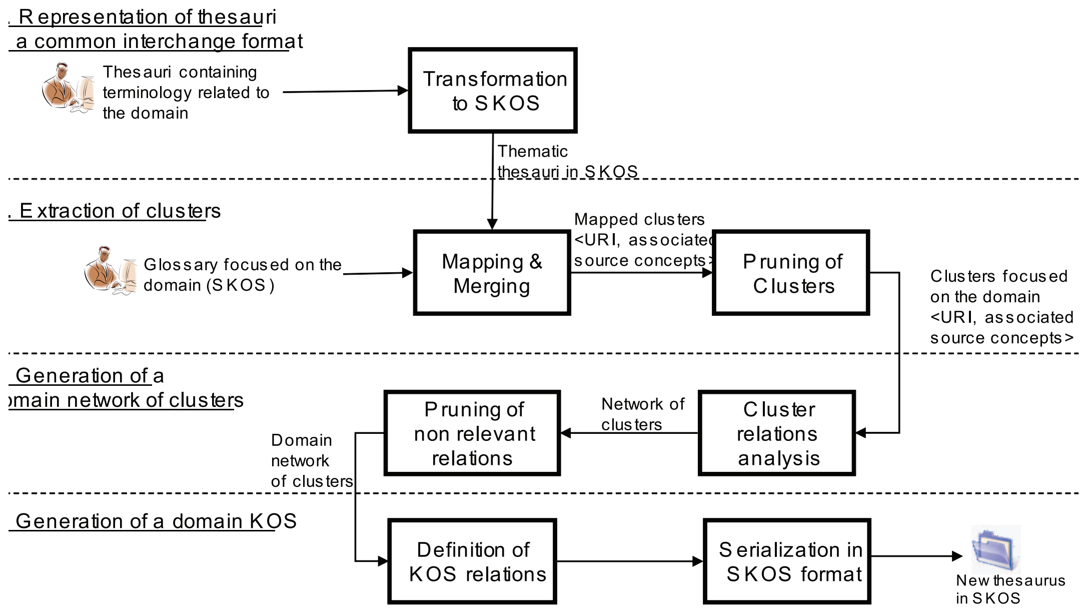

. Representation of thesauri
a common interchange format
Thesauri containing terminology related to the domain
Transformation to SKOS
Thematic thesauri in SKOS
. Extraction of clusters
Glossary focused on the domain (SKOS)
Mapping & Merging
Mapped clusters <URI, associated source concepts>
Pruning of Clusters
Clusters focused on the domain <URI, associated source concepts>
Generation of a
ımain network of clusters
Pruning of non relevant relations
Network of clusters
Cluster relations analysis
Domain network of clusters
Generation of a domain KOS
Definition of KOS relations
Serialization in SKOS format
New thesaurus in SKOS

Figure 3.12: Work-flow for the generation of a new terminological ontology

Figure 3.12 remarks the different steps of the developed process showing the inputs and the produced results. The creation process is divided into four different tasks: the harmonization of the interchange format used for the thematic thesauri, the mapping of concepts from the thematic thesauri, the generation of a network of thematic concepts about the selected area of knowledge, and the transformation of this network into a new thesaurus.

The harmonization of the interchange format is an external step required to provide a homogeneous input to the generation system. The other steps take this input and transform it into a new thesaurus focused on the desired theme. Following subsections describe in detail each one of these processes steps and the main components of software created to perform them (see figure 3.13).

3.4.1.1 Harmonization of the interchange format

As it has been indicated in section 3.2, along the years, a lot of well-established terminological models have been created in different domains. However, the lack of standardization has produced a huge variety of incompatible formats.

Managing many different formats as input for the merging process is not viable because the process should be modified each time the set of used ontologies is changed. Therefore, to make the process reusable, the first step in the generation process is to homogenize all the source

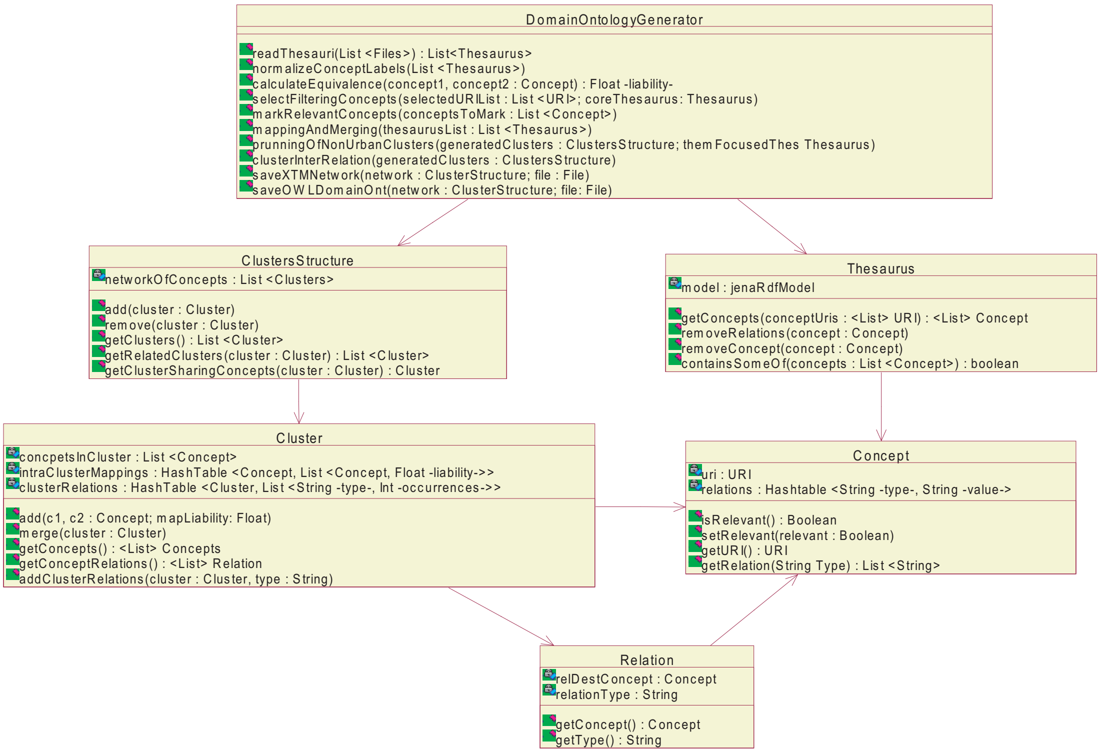

DomainOntologyGenerator
readThesauri(List <Files>) : List<Thesaurus>
normalizeConceptLabels(List <Thesaurus>)
calculateEquivalence(concept1, concept2 : Concept) : Float -liability-
selectFilteringConcepts(selectedURIList : List <URI>; coreThesaurus: Thesaurus)
markRelevantConcepts(conceptsToMark : List <Concept>)
mappingAndMerging(thesaurusList : List <Thesaurus>)
prunningOfNonUrbanClusters(generatedClusters : ClustersStructure; themFocusedThes Thesaurus)
clusterInterRelation(generatedClusters : ClustersStructure)
saveXTMNetwork(network : ClusterStructure; file : File)
saveOWLDomainOnt(network : ClusterStructure; file: File)
ClustersStructure
networkOfConcepts : List <Clusters>
add(cluster : Cluster)
remove(cluster : Cluster)
getClusters() : List <Cluster>
getRelatedClusters(cluster : Cluster) : List <Cluster>
getClusterSharingConcepts(cluster : Cluster) : Cluster
Thesaurus
model : jenaRdfModel
getConcepts(conceptUris : <List> URI) : <List> Concept
removeRelations(concept : Concept)
removeConcept(concept : Concept)
containsSomeOf(concepts : List <Concept>) : boolean
Cluster
concpetsInCluster : List <Concept>
intraClusterMappings : HashTable <Concept, List <Concept, Float -liability->>
clusterRelations : HashTable <Cluster, List <String -type-, Int -occurrences->>
add(c1, c2 : Concept; mapLiability: Float)
merge(cluster : Cluster)
getConcepts() : <List> Concepts
getConceptRelations() : <List> Relation
addClusterRelations(cluster : Cluster, type : String)
Concept
uri : URI
relations : Hashtable <String -type-, String -value->
isRelevant() : Boolean
setRelevant(relevant : Boolean)
getURI() : URI
getRelation(String Type) : List <String>
Relation
relDestConcept : Concept
relationType : String
getConcept() : Concept
getType() : String

Figure 3.13: UML Model of Domain Ontology Creator Tool

model representations to reduce the complexity of the whole process. The objective is to be able to increase or change the knowledge base used for the generation (using other thesauri) without having to modify the used generation software.

The format selected as input has been SKOS. If the glossary or any some of the cross-domain thesauri are not in this format they have to be translated. To do so, a transformation tool has to be developed following the methodology described in section 3.2.

3.4.1.2 Extraction of clusters

Once all the source thesauri (and the glossary) are in the same format, the next step is to enrich the glossary with thematically related concepts and relations extracted from the other models. To identify the thematically relevant concepts in the source thesauri it is needed to calculate their intersection with respect to the selected glossary of suitable terminology (i.e., they have to be aligned). Additionally, to expand the core set of concepts provided by the selected glossary with others of the same theme, the source thesauri have also to be aligned between them. The objective is to find relevant entities that are thematically close to the existent in the glossary. The extraction of clusters step does this task by aligning the different terminological models and aggregating the identified equivalent concepts into clusters (only the clusters with thematically relevant concepts are used). These clusters contain the commonalities between the models (relations, properties and attributes associated to each concept) and they are used in the following steps as a basis for the construction of the target thematic thesaurus.

Terminological ontologies have some particularities in their structure with respect to formal ontologies that have to be taken into account to perform an alignment between them. While formal ontologies represent concepts as classes and the entities that follow the concept definition as instances of a class, terminological ontology terms are usually modeled as instances of a general "Concept" class which define the available types of properties and relations [148, 52].

This way of modeling terminological ontologies simplifies its use for classification, since the terms can be directly used as values of properties in the description of resources. As commented, by Noy [164], representing each concept of a terminological ontology as a different class is problematic because this increases necessarily the complexity of the terminological model. For example, the classes have to be used as property values in the description of the resources, making the model to lose computational completeness (this can be avoided, but elaborated artifacts have to be defined).

The difference in structure of terminological ontologies with respect to formal ones reduces the number of alignment techniques that can be selected from the ones described in section 3.3.1. For example, the matching techniques based on the analysis of the classes, properties, and attributes are not applicable given that all the source models have the same types of properties and attributes (they are all represented using SKOS model). With respect to the techniques based on data analysis, the structure of the terminological ontology inhibit them to be used because the elements to relate in the used terminological ontologies are the instances.

The rest of techniques, such as techniques based on the analysis of entity names, techniques based on the use of external resources or techniques based on semantic interpretation can be applied but on instances and instance property values. For example, instead of using a linguistic technique to find equivalence between class names, the same technique can be used to find equivalences between term labels of different instances. For using these techniques, it is important to note that the multilingual characteristics of the glossary and terminological ontologies to align are of great use as additional resources in the generation of the matchings between the models.

The extraction of clusters process is divided into two different steps. First, the mapping & merging step that relates the thesauri and the glossary between them, merging equivalent concepts into a new one that acts as a cluster of source concepts (it groups concepts from the input thesauri). Second, the pruning of non thematically related clusters step removes those clusters that do not contain clusters containing terms of the glossary or that have a close relation with other one that fulfills this condition.

Mapping & merging In the mapping & merging step, every concept of every used terminological ontology (thesauri and glossary) is aligned to extract the commonalities and generate clusters of equivalent concepts. Each generated cluster represents a group of equivalent concepts and it is identified with one of the URIs of the original concepts. A code oriented view of the process is shown in the algorithm 1.

```
Procedure ClustersStructure mappingAndMerging(List thesaurusList);
begin
    normalizeConceptLabels(thesaurusList);
    ClustersStructure generatedClusters =new ClustersStructure();
    for int i=0;i<thesaurusList.size();i++ do
        List thesConcepts = thesaurusList.get(i).getConcepts();
        for int j=0; j<thesConcepts.size();j++ do
            Concept thesConcept = thesConcepts.get(j));
            Cluster clust= new Cluster(thesConcept);
            for k=i+1;k<thesaurusList.size() do
                List thesConceptsComp = thesaurusList.get(k).getConcepts();
                for int l=0; l<thesConceptsComp.size();l++ do
                    Concept thesConceptComp = thesConceptsComp.get(l));
                    int equivalenceValue = calculateEquivalence(thesConcept,thesConceptComp);
                    if equivalenceValue >0 then
                        clust.add(thesConcept,thesConceptComp,equivalenceValue);
                    end
                end
            end
            Cluster existentCluster=generatedClusters.getClusterSharingConcepts(clust);
            if existentCluster != null then
                existentCluster.merge(clust);
            else
                generatedClusters.add(clust);
            end
        end
    end
    return generatedClusters
end
```

Algorithm 1: Matching process

In this context, two concepts from different thesauri are considered equivalent when at least one of the labels of a concept (preferred and alternatives) is equal to one of the labels from the other concept. To identify these equivalences, from the matching techniques described in section 3.3.1, the described process focuses on the comparison of preferred and alternative labels from concepts in the different available languages (string comparison) leaving the use of relations and other properties such as definition or scope notes as improvements of the basic technique.

To improve results of the matching procedure, the labels are normalized by removing the accents, capital letters and plurals (see character normalization and lemmatization techniques in section 3.3.1). Additionally, the matching could have been enriched with misspellings detection, stemming and word order analysis among others, but given the usual high quality of thesauri content no much improvement would be expected from the use of these techniques.

The use of multilingual thesauri provides additional labels for the matching process increasing the probability of finding an equivalence (two concepts may differ in a language but have a coincidence in other one). Additionally, the number of labels (in multiple languages) that coincide between two matched concepts can be used as measure of the similarity between the concepts and the quality of the matchings. The higher the number of labels two concepts share (from the total they have), the higher their equivalence is. Equation 3.4.1 shows the formula used to measure this similarity. It calculates the percentage of common labels between two concepts, being NMatLab the number of matched labels between two concepts; NLab_1, the total number of different labels of the first concept involved in the matching; and NLab_2 the equivalent for the second one. This formula is somewhat equivalent to the substring similarity formula used in matching systems to calculate the distance between two strings (see [49, chap. 4]).

$$conceptSimilarity = \frac{2 * NMatLab}{NLab\_1 + NLab\_2}. \tag{3.4.1}$$

The concept similarity measure of equation 3.4.1 can be applied to each obtained mapping, but in this process it is only used to analyze the quality of the matching of each source thesaurus with respect to the glossary focused in the selected thematic. This analysis has as objective to show the relevance that each different terminological model gives to the selected theme (see the experiments in section 3.4.2).

Figure 3.14 shows a simplified example of a generated cluster ("Inland Water" cluster). The example does not uses alternative labels for cluster construction and does not fix the plural differences but it is enough to show how the cluster generation works. In the figure, it can be seen that the AGROVOC "Inland Waters" concept is included in the cluster due to the presence of this label and its Spanish translation in the concepts of EUROVOC. The relevance of the matchings has been included to show that they can be seen as a measure of the similarity in the concepts definition.

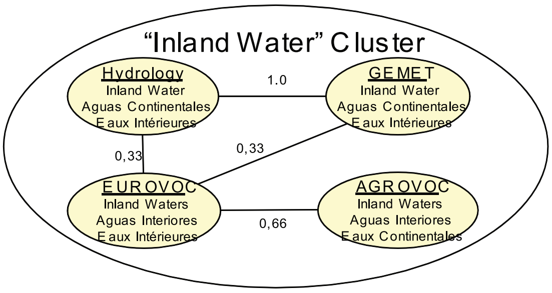


Figure 3.14: Example of a possible cluster

Pruning of non relevant clusters From the clusters obtained in the mapping process, only those containing concepts about the desired theme are required in the following steps to generate the desired thesaurus. Therefore, the non thematically relevant have to be removed from the system. A schema of the pruning process is shown in the algorithm 2. The division between relevant and non relevant clusters is performed using the following rule: A cluster is only considered relevant if it contains a concept from the glossary of selected terminology, or if it is directly related (*broader*, *narrower* and *related* relationships) to a cluster that fits in the first case. The rest of the clusters are considered as not relevant and they are pruned from the system. The reason to include clusters that do not directly contain terminology from the selected glossary is to extend the set of core concepts with the relevant knowledge that different areas of knowledge (the provided by the different used thesauri) consider related to it.

```
Procedure pruningOfNonUrbanClusters(ClustersStructure generatedClusters, Thesaurus themFocusedThes);
begin
    List clusterList = generatedClusters.getClusters();
    for int i=0;i<clusterList.size();i++ do
        Cluster clust = clusterList.get(i);
        List relClustList = generatedClusters.getRelatedClusters(clust);
        relClustList.add(clust);
        boolean themFocusedCluster = false;
        for int k=0;k<relClustList.size();k++ do
            if themFocusedThes.containsSomeOf(relClustList.get(k).getConcepts()) then
                themFocusedCluster=true;
            end
        end
        if not themFocusedCluster then
            generatedClusters.remove(clust);
        end
    end
end
```

Algorithm 2: Cluster pruning process

3.4.1.3 Generation of a domain network of clusters

The clusters generated in the previous step contain the collected terminology about the desired theme (extracted from different knowledge models), but not how this knowledge is inter-related. The next step in the thesaurus generation process is to identify the relations and add them to the clusters. The result obtained is a network of clusters that can be seen as a domain ontology describing the knowledge of the area of interest. Figure 3.15 in the experiments section (section 3.4.2) shows an example of a network obtained for a specific context.

In this context, it is important to note that in addition to the use of the network of concepts as base for the generation of the a new thematic thesaurus, it can also be used for task such as the analysis of the extent in which the studied area of knowledge is represented in the input thesauri or the detection of inconsistencies between the source models.

The generation process of the network of clusters is divided in two different subtasks, generation of inter-cluster relations and pruning of non-relevant relations.

Generation of inter-cluster relations The creation of the relations between the clusters is done by converting the original relations of the concepts contained in the clusters into relations between the clusters that contain them. In addition to the basic relations (*broader*, *narrower* and *related*), *sibling* relations (*narrower* of its *broader*) have been also considered. They have been included to detect those concepts that are not directly related but they are close in meaning. Other more complex relations such as the *grandparent* (*broader* of its *broader*) or *grandchildren* (*narrower* of their *narrower*) could also have been considered to detect clusters relations, but they are left for future functionality expansions.

Algorithm 3 shows a schema of the cluster relation process. For each relation between concepts from two different clusters in their original thesauri, a relation of the same type is added between the clusters containing them. If several relations between two clusters are created, they are aggregated into a single one containing the number and types of the original relations. The number of original relations shows the relevance of the generated inter-cluster relation. The more relations are found in the source models, the stronger the generated relationship is (it has been found as relevant in more different contexts).

```
Procedure clusterInterRelation(ClustersStructure generatedClusters);
begin
    List clusterList = generatedClusters.getClusters();
    for int i=0;i<clusterList.size();i++ do
        Cluster clust = clusterList.get(i);
        List relatedC1 = clust.getConceptRelations();
        for int j=i+1;j<clusterList.size();j++ do
            Cluster clust2 = clusterList.get(j);
            for int k=0;k<relatedC1.size();k++ do
                Relation relR1 =relatedC1.get(k);
                if clust2.contains(relR1.getConcept()) then
                    clust.addClusterRelation(clust2, relR1.getType());
                end
            end
        end
    end
end
```

Algorithm 3: Cluster interrelation

Pruning of non-relevant relations The generated cluster relations can be classified by their relevance (depending on the number of source thesauri that contain them). This classification can be used to reduce the size of the generated network (it may be too complex and/or contain spurious clusters) by selecting only the most relevant relationships.

A specific process has been developed to perform this relationship punning. This process requires the selection by the user of a threshold that is used to determine if a relation is maintained. All the relations with a weight (number of occurrences) below the indicated threshold are pruned. Additionally, after the prune of relations, if some clusters have lost all their relations and do not contain terms from the source glossary they are removed (this is done to obtain a more compact network structure).

3.4.1.4 Generation of a new domain thematic thesaurus

The last step of the defined process is to take the network of clusters describing the content of the decided area of knowledge and transform it into a thesaurus. The structure of the network is reorganized into a hierarchical model. This is possible since the original models are also thesauri; therefore, the properties and relations of the clusters of the network are compatible.

```
Procedure generationOfNewThesaurus(ClustersStructure generatedClusters, String schemaURI, File
fileToSave);
begin
    SKOSWritter writter = new SKOSWritter();
    SKOSModel newThesaurus = writter.createSKOSModel();
    SKOSConceptSchema schema = newThesaurus.createConceptShema(schemaURI);
    List clusterList = generatedClusters.getClusters();
    for int i=0;i<clusterList.size();i++ do
        Cluster clust = clusterList.get(i);
        SKOSConcept concept = SKOSModel.createConcept(clust.getURI(),schema);
        fillConceptProperties(concept, clust);
        List relationList = clust.getClusterRelation();
        List broaders, relateds; getSuitableRelations(relationList,broaders,relateds); if broaders.size()==0
        then
            schema.addTopConcept(concept);
        end
        for int j=0;j<broaders.size();j++ do
            ClusterRelation relation= relationList.get(j);
            SKOSConcept destconcept = SKOSModel.createConcept(relation.getDestinationUri(),schema);
            concept.addRelation(SKOSRelation.broader,destconcept);
            destconcept.addRelation(SKOSRelation.narrower,concept);
        end
        for int j=0;j<relateds.size();j++ do
            ClusterRelation relation= relationList.get(j);
            SKOSConcept destconcept = SKOSModel.createConcept(relation.getDestinationUri(),schema);
            concept.addRelation(SKOSRelation.related,destconcept);
            destconcept.addRelation(SKOSRelation.related,concept);
        end
    end
    List topConceptList = SKOSModel.getTopConcepts();
    for int i=0;i<topConceptList.size();i++ do
        removeCycles(topConceptList.get(i), new ArrayList());
    end
    String skosFile = writter.generateSKOSFile(newThesaurus);
    fileToSave.write(skosFile);
end
```

Algorithm 4: Generation of the new thesaurus

The generation process is shown in algorithm 4. The process transforms the clusters in the network into concepts of the new thesauri. The cluster content (preferred and alternative labels, definitions and scope notes of the source thesauri) is transferred into the new concepts; and the cluster relations are transformed into thesaurus relations modifying them to generate the hierarchical structure of a thesaurus.

The selection of the preferred label of each generated concept (one per language) is performed as follows: If the cluster contains a concept from the thematic glossary its preferred label is the selected; In other case, the selected label is the one in the concept whose source thesaurus has been loaded first in the system (it can be changed by configuration). The rest of the labels in the cluster are left as alternatives.

With respect to the relations structure, each relation between the clusters is marked with the type that has more occurrences (*broader/narrower* or *related* relationships). The *siblings* relationships are used as a concept closeness measure to determine the selection between a *broader/narrower* relationship and a *related* relationship in those situations in which there may be doubts. Algorithm 5 details the selection process of the most suitable relation. This algorithm is based in the application of the following set of rules:

- A relation in the new thesaurus is marked as *broader* if the number of *broader* relations found between the original clusters is greater than zero and the sum of the *broader* and *sibling* relations is greater than the number of *related* relationships. To maintain the consistency of the thesaurus, for each *broader* relationship that is defined an inverse *narrower* relation is also generated.

- In a thesaurus only a *broader* relation should be defined for each concept. Therefore, if more than one *broader* relation is generated using the previous rule, the *broader* relation that has the highest weight (biggest number of relations of this type in the original cluster) is preserved and the rest of them are tagged as *related* relationships. If two or more relations share the same weight, the one with more *sibling* is the selected. Finally, if they also have the same number of *sibling*s the user has to take the final decision.

- If a relation has not been tagged using the previous rules and the number of its inner *related* relationships is greater than zero, the relation is marked as *related* relationship.

- The relationships that are only marked as *sibling* type are discarded.

Having generated the concepts and relations of the thesaurus, it is needed to identify its top terms. This is quite easy, as by definition, a top term is a concept without a *broader* relationship. Therefore, it is only needed to review all generated concepts and mark those without a *broader* relationship as top terms of the thesaurus.

Finally, the last generation step reviews that the *broader/narrower* structure in the obtained thesaurus contains no cycles. Starting from the top concepts, all the *narrower* relationships are recursively analyzed until the end of the structure is reached or a cycle is found. When a cycle is found, the *broader/narrower* relationships that references to a processed concept are transformed into *related* relationships between the concepts (to remark that there is a relation between them).

The new thesaurus is stored in SKOS format to be able to integrate it with the rest of the terminological ontologies used along this thesis. To do this, the SKOS writer component described in section 3.2 has been integrated as part the generation software. The SKOS writer receives the concepts and relations created by the generation software and represents them in SKOS format.

```
Procedure getSuitableRelations(List relationList, List broaders, List relateds);
begin
    ClusterRelation moreRelevantBroader = new ClusterRelation();
    for int i=0;i<relationList.size();i++ do
        ClusterRelation relation = relationList.get(i);
        int weightRelated=relation.getNumberof(SKOSRelation.related);
        int weightBroader=relation.getNumberof(SKOSRelation.broader);
        int weightSibling=relation.getNumberof(SKOSRelation.sibling);
        if (weightBroader>0) && hasBigestBroaderWeight(relation, moreRelevantBroader) then
            Forint i=0;i<broaders.size();i++ ClusterRelation relToTransfer = broaders.get(i);
            if relToTransfer.getNumberof(SKOSRelation.related)>0 then
                relateds.add(relToTransfer);
            end
            broaders.removeAll;
            broaders.add(relation);
            moreRelevantBroader=relation;
        else
            if (weightBroader>0) && hasSameBroaderWeight(relation, moreRelevantBroader) then
                broaders.add(relation);
            else
                if weightRelated>0 then
                    relateds.add(relation);
                end
            end
        end
    end
end
```

Algorithm 5: Selection of the suitable relations between the available

3.4.2 Testing the method

This process has been applied in two different knowledge areas. In a first experiment, the method has been used to create a new thesaurus focused on the urbanism subject. In the second one, it has been applied in the hydrological domain.

3.4.2.1 Testing the method in the urban domain

Urbanism is usually defined as the study of cities including their economic, political, social and cultural environment. This field has attracted much interest within geographical information context by its relevance to the citizens. In this context, the ontologies have begun to be used to facilitate and improve the access to the urban resources (as it has happened in others knowledge areas). An example of the use of ontologies in the urban context is the COST C21 Action[14] project that has been created with the objective of increasing the knowledge and promoting the use of ontologies in the domain of Urban Civil Engineering projects.

Nowadays, there are terminologies containing urban vocabulary but not specifically focused on urbanism. In this context, the definition of stable terminological urban ontologies in the area is needed; but the multidisciplinarity of urbanism makes the selection of the suitable terminology difficult, since it requires a revision of all the cross-domain areas involved in the urbanism to capture the thematically related concepts. As a step in this direction, a draft of an urban model has been created using the generation process described earlier.

[14] http://www.towntology.net/

```
Procedure removeCycles(SKOSConcept concept, List broaders);
begin
    List narrowList concept.getRelation(SKOSRelation.narrower);
    for int i=0;i<narrowList.size();i++ do
        SKOSConcept conceptRel = narrowList.get(i);
        if broaders.contains(conceptRel) then
            concept.removeRelation(conceptRel,SKOSRelation.narrower);
            conceptRel.removeRelation(concept,SKOSRelation.broader);
            concept.addRelation(conceptRel,SKOSRelation.related);
            conceptRel.addRelation(concept,SKOSRelation.related);
        else
            broaders.add(concept);
            removeCycles(conceptRel, new ArrayList(broaders));
        end
    end
end
```

Algorithm 6: Removal of cycles in the generated structure

The urban glossary required for the generation process as core of the urban ontology has been constructed using URBISOC[15] thesaurus as base. URBISOC thesaurus was developed by the Spanish National Research Council to facilitate classification at bibliographic databases specialized in scientific and technical journals on Geography, Town Planning, Urbanism and Architecture. However, URBISOC is not suitable as urban thesaurus since it contains many general terminology not specifically related to urbanism.

In this context, the urban glossary has been constructed using an heuristic filtering process that extracts the urban concepts from URBISOC. Algorithm 7 shows the filtering process. This process requires the manual selection of a small set of initial concepts from URBISOC to work. Recursively, all the *narrower* and *related* concepts of this selected core are added to the set until no more elements are included. The rest of the concepts of URBISOC are considered non thematically relevant and discarded.

In this application example, the "Planificación urbana" (Urban planning) concept has been selected as seed concept(it is used very frequently in urban contexts). The result obtained have been a glossary that has eliminated most of the general terminology and contains 3,091 of the 3,609 concepts of URBISOC.

In addition to the urban glossary, the other input required by the generation system is a set of thesauri containing thematic related terminology. In this case the selected thesauri have been GEMET, AGROVOC, EUROVOC and UNESCO. They have been selected because they contain large sets of urban terminology and each one provide a different view of it. Additionally, they are all multilingual and provide labels for the concepts in Spanish, English, and French (between others). The description of these thesauri and its content is shown in section 2.2.1.4.

Since the used cross-domain thesauri (including URBISOC) were published in completely different representation formats (see section 3.2.3), they had to be translated into SKOS format using the process described in section 3.2.

[15] http://thes.cindoc.csic.es/index_URBA_esp.html

```
Procedure selectFilteringConcepts(List selectedURIList, Thesaurus coreThesaurus);
begin
    List selectedConceptList = coreThesaurus.getConcepts(selectedURIList);
    markRelevantConcepts(selectedConceptList);
    List thesConcepts = coreThesaurus.getConcepts();
    for int i=0;i<thesConcepts.size();i++ do
        Concept thesConcept = selectedConceptList.get(i));
        if not thesConcept.isRelevant() then
            coreThesaurus.removeRelations(thesConcept);
            coreThesaurus.removeConcept(thesConcept);
        end
    end
end
Procedure markRelevantConcepts(List conceptsToMark);
begin
    for int i=0; i<conceptsToMark.size();i++ do
        Concept coreConcept = conceptsToMark.get(i));
        if not coreConcept.isRelevant() then
            coreConcept.setRelevant(true);
            markRelevantConcepts(coreConcept.getNarrowers(coreConcept));
            markRelevantConcepts(coreConcept.getRelateds(coreConcept));
        end
    end
end
```

Algorithm 7: Selection of filtering concepts

The clusters generated in the matching step have been used to analyze the degree of representation of the urban terminology in each of the used cross-domain thesauri. This has been done by counting the concepts of each thesaurus that has been matched to a concept in the glossary, and calculating the mean of the liability of the mappings. Table 3.5 shows the obtained results. It contains the number of concepts of each thesaurus, the number of concepts mapped to the glossary extracted from URBISOC, and the average relevance of the mappings (see equation 3.4.1). These results show that all the thesauri selected to expand and relate the glossary contain a relevant subset of urban related concepts with a quire similar number of concepts (between 358 and 418 urban concepts). Independently, each matched subset only cover a reduced part of the core glossary (12% on average) but combining them the coverage is around a 35%.

Name	Concepts	Mapped Concepts	Liability
GEMET	5244	388	0.765
EUROVOC	6649	418	0.588
AGROVOC	16896	369	0.684
UNESCO	4424	354	0.715

Table 3.5: Relevance of urbanism in cross-domain thesauri

In this use case, the generation of the relations between clusters has been performed a bit different from the described in the general process. The use of URBISOC to generate the core glossary provides a set of extra relations between the concepts of the glossary that have been taken into account for the construction of the network of concepts. These relations have been managed as the rest of relations between concepts from different clusters and used to determine the inter-cluster relationships.

The obtained network contains 4,698 clusters with 2,189 relations of weight 5 or greater, 2,181 of weight 4, 3,137 of weight 3, 12,000 of weight 2 and 44,518 of weight 1. In this context, the weight of the relations have been used to remove gradually the less relevant relations (and associated concepts) obtaining the set of results shown in table Table 3.6. Each row in the table shows the size of the network that includes all the relations of at least "Minimum Weight" weight. It includes the number of clusters, the average size of each cluster, the total number of relations between clusters, and the average number of relations of each cluster.

Minimum Weight	**Clusters**	**Cluster Size**	**Relations**	**Cluster Relations**
1	4698	1.83	64025	13.62
2	2568	2.39	42823	7.59
3	1514	2.90	17570	4.95
4	1082	3.10	11333	4.03
5	681	3.43	5622	3.21

Table 3.6: Size of the network of urban clusters

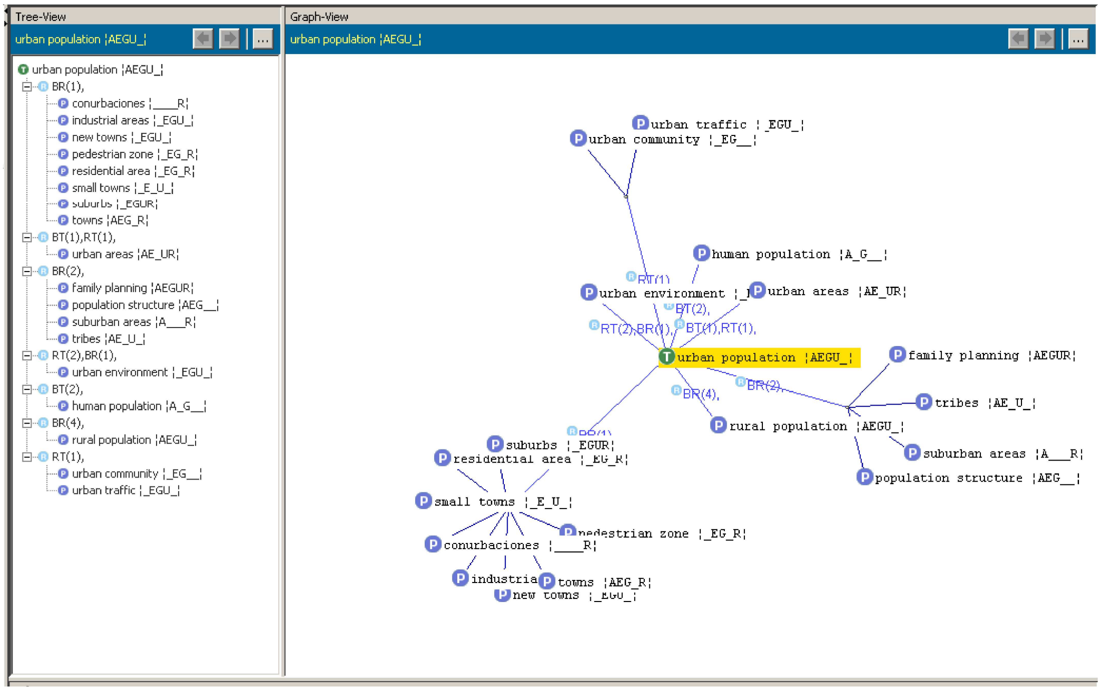
To facilitate the analysis of the structure and content of the different network of clusters obtained for each pruning level, they have been represented using the XTM format [192]. XTM is an XML based format specifically designed to facilitate the visualization and browsing through network of concepts. Nowadays, there is a wide range of tools able to manage XTM, and from them the TMNAV tool created in the TM4J project[16] has been used to obtain the display of the network.

The XTM files have been generated in such a way that the TMNAV tool shows the preferred English label (or the Spanish one if there is no any English label) of the clusters and the initials of the thesauri that have provided a concept to the cluster (i.e., AEGUR means that the concepts exist in AGROVOC, EUROVOC, GEMET, UNESCO and URBISOC; and _E__R means that the concept does only exist in EUROVOC and URBISOC). With respect to the relations, they are labeled with the types and number of the original relations used to define it. BT indicates *broader* relation, NT is *narrower*, RT means *related*, and BR is used for *siblings*.

Figure 3.15 shows a screenshot of TMNAV tool containing all the obtained relations and clusters around the "urban population" cluster from the complete network of clusters (without prune). As it can be seen, the "Urban Population" cluster contains concepts from all the source thesauri except URBISOC and maintains relations with a lot of other clusters. For example, it is related to the "urban areas" cluster through a relation of type *broader* and *related* (both with weight 1).

Each of the generated network of clusters (each one with a different punning level) has been used to create a different version of the thematic thesaurus in SKOS format. Table 3.7 shows the dimensions of each thesaurus generated. It contains the total number of concepts

[16] http://tm4j.org/



- urban population ¦AEGU_¦
  - BR(1),
    - conurbaciones ¦____R¦
    - industrial areas ¦_EGU_¦
    - new towns ¦_EGU_¦
    - pedestrian zone ¦_EG_R¦
    - residential area ¦_EG_R¦
    - small towns ¦_E_U_¦
    - suburbs ¦_EGUR¦
    - towns ¦AEG_R¦
  - BT(1),RT(1),
    - urban areas ¦AE_UR¦
  - BR(2),
    - family planning ¦AEGUR¦
    - population structure ¦AEG__¦
    - suburban areas ¦A___R¦
    - tribes ¦AE_U_¦
  - RT(2),BR(1),
    - urban environment ¦_EGU_¦
  - BT(2),
    - human population ¦A_G__¦
  - BR(4),
    - rural population ¦AEGU_¦
  - RT(1),
    - urban community ¦_EG__¦
    - urban traffic ¦_EGU_¦

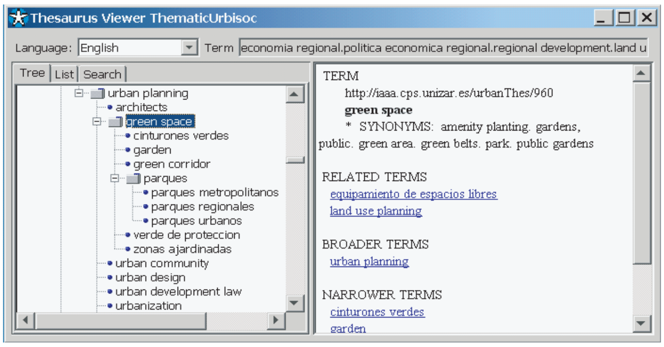
Figure 3.15: Visualization of a part of the generated urban thesaurus

of the generated thesaurus, the number of preferred and alternative labels, the number of *broader/narrower* relations and the number of related relationships. These different alternatives can be manually reviewed to select the most suitable one for the desired application. This revision can be done using any tool able to manage SKOS format. In the context of this thesis, it has been done using the ThManager tool described in section 4.5.

Threshold	Nr Concepts	Nr Pref Labels	Nr Alt Labels	Nr BT/NT Rels	Nr RT Rels
1	4698	9033	16225	4281	14022
2	2568	6347	14598	1482	4146
3	1508	4262	11556	852	1278
4	1069	3087	9135	550	570
5	672	1996	6653	317	302

Table 3.7: Dimensions of the generated urban thesauri

Figure 3.16 shows a subset of the generated thesaurus provided by the ThManager tool to give an idea of the structure of the resulting knowledge model. The obtained results are conditioned by the fact that the used glossary (extracted form URBISOC) is only in Spanish. This characteristic produces that those concepts that have not been matched to any other one do not contain English labels and have to be shown in Spanish. The figure shows the branch of the thesaurus starting with the "urban planning" concept. The generated concepts hierarchy integrates, in a reasonable way, most of the concepts. For example, the generation system relates the “parques” (parks) branch from the original URBISOC, as *narrower* of “green space”. However, there are also some deficiencies. For example, the method has not detected that the concepts ‘cinturones verdes” and “green corridor” should have been grouped into the same cluster because they are equivalent. The reason of this problem is that the Spanish translation of “green corridor” in the source thesauri is “pasillos verdes”. This is a good example of the kind of problems that can be found in the generated model. It is a nuisance because it must be manually corrected. But it is not critical due to the fact that both concepts have been classified as *narrower* terms of “green space” (that facilitates the identification of the problem).



Language: English Term economia regional.politica economica regional.regional development.land u

- urban planning
  - architects
  - green space
    - cinturones verdes
    - garden
    - green corridor
    - parques
      - parques metropolitanos
      - parques regionales
      - parques urbanos
    - verde de proteccion
    - zonas ajardinadas
  - urban community
  - urban design
  - urban development law
  - urbanization

TERM
http://iaaa.cps.unizar.es/urbanThes/960
green space
* SYNONYMS: amenity planting. gardens, public. green area. green belts. park. public gardens

RELATED TERMS
equipamiento de espacios libres
land use planning

BROADER TERMS
urban planning

NARROWER TERMS
cinturones verdes
garden

Figure 3.16: Visualization of a part of the generated urban domain ontology

The translation of the glossary content to the same languages provided by GEMET, EUROVOC, AGROVOC and UNESCO thesaurus (at least English, Spanish, and French) would improve the quality of the result thesaurus because it would reduce the non identified equivalences. This translation could be done manually, or automatically with the help of a multilingual dictionary. Each approach has advantages and disadvantages. On the one hand, a manual translation provides high quality results but requires a lot of effort. On the other hand, an automatic approach is fast but it can introduce errors caused by synonymy problems that would affect to the generation process.

The translation of the glossary has not been included in the present version, but it is believed that an automatic translation process could be used due to the specific character of the urban thesaurus would limit the polysemy problems. Additionally, polysemy problem could be partially detected and tackled by performing crossed translations. For example, each term translated to English could be translated from it to French and compared with the translation obtained through a direct translation. If the obtained translations matches, the French translation is expected to be correct. In other case, there is a problem and the translation has to be manually reviewed (or directly discarded). The inclusion of a translation step would require the modification of the generated process. It should be performed after the step of normalization of the labels in the glossary with the objective of having more homogeneous labels to translate.

3.4.2.2 Testing the method in the hydrological domain

In recent years, there has been a general concern about environmental issues in the European context that has lead to the creation of national and international policies encouraging the development of information infrastructures about this issue. In the environmental field, the hydrology management is one of the most relevant and it has attracted a lot of attention and regulation. Hydrologists must monitor a great variety of features and phenomena that, although initially disconnected, may affect the status of water bodies. Due to this variety, information systems managing hydrological information require of complete well-known domain models containing all the required terminology to classify and retrieve the hydrology related information. As an initial step in the creation of these models, the generation system described earlier has been used to create a draft of a thesaurus focused on hydrology that can be used to facilitate the management and integration of hydrological resources.

The first input required by the generation tool is a glossary of concepts focused on the desired domain (in this case hydrology). This glossary has been manually created using as base the existent terminology provided by the European Water Framework Directive[17] (WFD) for constructing hydrological models [207]. The resulting glossary includes 108 concepts organized as a plain list of 56 concepts and two hierarchies containing the other 52 concepts.

With respect to the set of thesauri required to define the relation structure of the thesaurus to generate, the same ones that were used in the urban example have selected (GEMET, AGROVOC, EUROVOC and UNESCO). This has been done due they were already available in SKOS format and they contain a good deal of hydrology related terminology.

As in the previous example, the clusters generated in the matching step have been used to analyze the degree of representation of the hydrological terminology in each of the used cross-domain thesauri. Table 3.8 shows the obtained results. It contains the number of concepts of each thesaurus, the number of concepts mapped to the glossary, and the average relevance of the mappings according to equation 3.4.1 (section 3.4.1.2).

It can be seen from the results that the source thesauri contain a relevant subset of glossary concepts. Around a quarter of the glossary terms have been matched to AGROVOC, EUROVOC, and UNESCO thesaurus. GEMET performance is much better; it has almost two thirds of the 108 concepts from the glossary matched to it. The combined coverage is of around a 75 percent of the concepts in the glossary.

The great difference of results with respect to the obtained in urban example is caused by two differences in the way that the generation process has been applied. Firstly, the glossary

[17] The European Water Framework Directive is considered to be the most important piece of legislation in the hydrology area. Its main objective is to achieve an accurate management of all water bodies and reach a “good status” for them by 2015.

has been manually selected. Secondly, the glossary content is provided in English, Spanish and French. The manual selection of the glossary is an advantage in the sense that it increases its quality and the thematic closeness of the contained concepts. With respect to the use of a multilingual glossary, as it has been previously described, the existence of multiple labels in the glossary increases the probability of mappings with the thematic thesauri that are also provided in these languages.

Name	Concepts	Mapped Concepts	Liability
GEMET	5244	67	0.704
EUROVOC	6649	21	0.558
AGROVOC	16896	37	0.641
UNESCO	4424	27	0.726

Table 3.8: Relevance of hydrology in cross-domain thesauri

In this context, the weight of the relations have been used to remove gradually the less relevant relations (and associated concepts) obtaining the set of results shown in table Table 3.6.

The obtained network of concepts contains 354 concepts with 250 relations of weight 5, 288 of weight 4, 395 of weight 3, 1387 of weight 2 and 2644 of weight 1. As in the previous example, the weight of the relations have been used to remove gradually the less relevant relations (and associated concepts) obtaining the set of results shown in table Table 3.9. Each row in the table shows the size of the network that includes all the relations of at least "Minimum Weight" weight. It includes the number of clusters, the average size of each cluster, the total number of relations between clusters, and the average number of inter-cluster relations.

Minimum Weight	Clusters	Cluster Size	Relations	Cluster Relations
1	354	2.77	4964	14.02
2	245	3.06	2320	9.46
3	174	3.16	933	5.36
4	116	3.34	538	4.63
5	75	4.11	250	3.33

Table 3.9: Size of the network of hydrology clusters

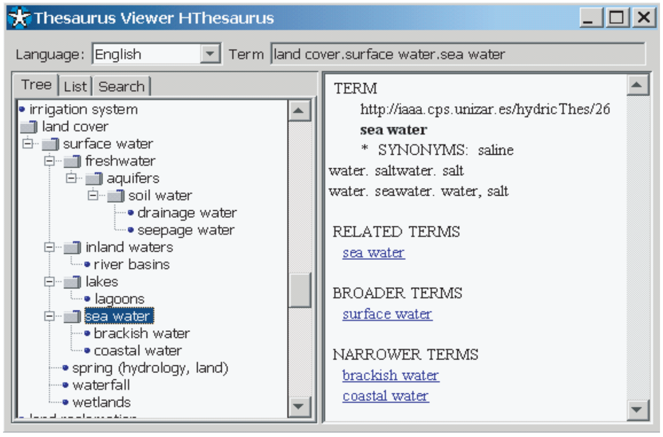
In the same way as in the urban example, each of the generated network of clusters (each one with a different punning level) has been used to create a different version of the thematic thesaurus in SKOS format. Table 3.10 shows the dimensions of each thesaurus generated. It contains the total number of concepts of the generated thesaurus, the number of preferred and alternative labels, the number of *broader/narrower* relations and the number of related relationships.

Figure 3.17 shows a subset of the generated thesaurus using the ThManager tool (described in section 4.5) to give an idea of the structure of the resulting knowledge model. The figure shows a branch of the thesaurus that starts with the "land cover" concept. It shows the

Threshold	Nr Concepts	Nr Pref Labels	Nr Alt Labels	Nr BT/NT Rels	Nr RT Rels
1	354	1062	2622	261	1014
2	245	735	2080	158	352
3	174	522	1585	93	134
4	116	348	1190	56	60
5	75	225	893	29	30

Table 3.10: Dimensions of the generated hydrology thesauri

generated hierarchy containing the different types of water bodies. Although it is true that the generated hierarchy is not the most appropriate (e.g., the "lakes" concept should be a narrower of "inland waters"). However, it only needs a minimal reorganization of the available concepts to provide a much more suitable model.



Language: English

Term land cover.surface water.sea water

- irrigation system
- land cover
  - surface water
    - freshwater
      - aquifers
        - soil water
          - drainage water
          - seepage water
    - inland waters
      - river basins
    - lakes
      - lagoons
    - sea water
      - brackish water
      - coastal water
    - spring (hydrology, land)
    - waterfall
    - wetlands

TERM
http://iaaa.cps.unizar.es/hydricThes/26
sea water
* SYNONYMS: saline water. saltwater. salt water. seawater. water, salt

RELATED TERMS
sea water

BROADER TERMS
surface water

NARROWER TERMS
brackish water
coastal water

Figure 3.17: Visualization of a part of the generated hydrology thesaurus

3.5 Formalization of a terminological ontology

As commented previously, the applicability of thesauri in the classification and information retrieval context has promoted the creation and diffusion of well-established thesauri in many different domains. These simple models are useful for most of classification and retrieval systems where search requisites are not very elaborated; however, in contexts with an information model of great complexity, more elaborated ontologies with formal is-a hierarchies, frame definitions or even general logical constraints are needed to improve the retrieval quality.

Fisher [56] states that the advantage of replacing concept-oriented terminological ontologies with formal ontologies, is that it can cover a spectrum of functionality which, in principle, includes all the traditional services of a classical thesaurus, and it can offer additional ones. Soergel et al. [188] remarks that it is needed to change the use of thesauri into other more formal models when at least one of the following requirements is needed.

- Improved user interaction with the ontology on the conceptual and the term level (query formulation, subject browsing, and user learning about the domain).

- Intelligent behind-the-scenes support for query expansion, both concept expansion and synonym expansion, within one language and across languages.

- Intelligent support for human indexers and automated indexing/categorization systems.

- Support for artificial intelligence and semantic web applications.

However, the required formal model does not always exist. In this context, since the development of ontologies from scratch requires much time and many resources, the activity of knowledge acquisition constitutes one of the most important steps at the beginning of the ontology development process. As its name indicates, this activity is devoted to gather all available knowledge resources describing the domain of the ontology and identify the most important terms in the domain [181].

Nowadays, the use of (semi-)automatic processing of knowledge resources that may alleviate the work of knowledge acquisition is broadly used. This task is known as ontology learning in the literature of ontological engineering [68, 10]. The aim of ontology learning is to apply the most appropriate methods to transform unstructured (e.g., text corpora), semi-structured (e.g., folksonomies, HTML pages) and structured data sources (e.g., databases, thesauri) into conceptual structures. As stated in [78], hierarchical classification standards, thesauri, and such taxonomies are likely the most promising sources for the creation of domain ontologies at reasonable costs, because they reflect some degree of community consensus and contain, readily available, a wealth of category definitions plus a hierarchy. The objective is to combine the terminological richness of good thesauri with the conceptual structure of full-fledged ontologies containing well-structured hierarchies of concepts connected through a rich network of detailed relations.

One of the most basic requirements to jump from terminological to formal ontologies is to be able to classify the *broader/narrower* relationships (that only indicate a general containment meaning) into specific categories such as *is-a*, *instance-of* or *is-part-of*. Other important required transformation is the classification of the abstract *related* relationships into different specific subtypes. Once these basic elements have been correctly transformed, other elements such as value restrictions or cardinalities or relations between others can be included to enrich the transformed ontology. Fisher [56] focuses on the problem of identifying these relations in a thesaurus model describing the issues inherent to the transformation.

This section analyzes the complexity and the viability of the formalization approaches for its use inside the global management context of ontologies in Spatial Data Infrastructures. To do so, a formalization prototype has been developed. This prototype implements some simple formalization rules on some thesauri, and the results are analyzed to see the viability of an approach of this kind.

3.5.1 State of the art

Among the works related to the transformation of thesauri into ontologies, we must cite first a set of works that transform thesauri from its native format into Semantic Web languages such as RDF, OWL or SKOS. The output of these methods cannot be categorized as a formal ontology because the relationships between concepts are still ambiguous, but at least it is a step forward. We move from the term-based approach recommended in ISO standards, in which terms are related directly to one another, into a concept-based approach. In the concept-based approach concepts are interrelated, while a term is only related to the concept for which it stands; i.e. a lexicalization of a concept. Section 3.2.1 shows some format transformation works in which a thesaurus in translated into a formal language. An additional example in this area is the work of Wielinga et al. [220] which describes the transformation of the Art and Architecture Thesaurus (AAT) into an ontology expressed in RDFS. The full AAT hierarchy was converted into a hierarchy of concepts, where each concept has a unique identifier and slots corresponding with the main term and its synonyms.

A second set of works are more ambitious and try to transform the ambiguous *broader / narrower* relationships of thesauri into more formal relationships such as *is-a* or *is-part-of* hierarchies. The ISO 2788 guidelines for monolingual thesauri contain a differentiation of the hierarchical relationship into generic, partitive and instance relationships. However, because the main purpose of thesauri was to facilitate document retrieval, the standards allow this differentiation to be neglected or blurred. But in contrast to thesauri, ontologies are designed for a wider scope of knowledge representation and need all these logical differentiations in relationships [56]. As stated in van Assem et al. [208] a major difference between thesauri and ontologies is that the latter feature logical *is-a* hierarchies, while in thesauri the hierarchical relation can represent anything from *is-a* to *is-part-of.* Fisher [56] identifies several cases where this no differentiation of the *broader / narrower* relationship may be a source of fallacies or problems when transforming a thesaurus into an ontology. In particular this work focuses on the problems of identifying subsumption and instance relationships behind the ambiguous *broader / narrower.*

For instance, Hepp and de Bruijn [78] describes an algorithm called GenTax to derive an RDF-S or OWL ontology from most hierarchical classifications available in the SKOS exchange format. This algorithm, implemented in the tool SKOS2Gentax[18], derives OWL classes from the instances of SKOS concepts and their broader and narrower relationships. The algorithm assumes that SKOS concepts can be used in different contexts with varying semantics of the

[18] http://www.heppnetz.de/projects/skos2gentax/

concepts and their relationships. The algorithm has two main steps. Firstly, it creates two ontology classes per SKOS concept: one for the context of the original hierarchy, and a related second class (subclass of the first one) for the narrower meaning of the concept in a particular context. Secondly, GenTax inserts *subClassOf* relations between the classes in the original hierarchy context. However, since SKOS broader and narrower relationships are translated by default to an *is-a* taxonomy, the output of the algorithm requires many corrections. Another example is the work of Amann et al. [7]. This work uses a preexistent ontology and combine it with a thesaurus to create a metadata schema for classification and querying.

Other works use natural language processing to refine the hierarchical relationship of thesauri. For example, Clark et al. [30] describes the experience of transforming a technical thesaurus (Boeing's technical thesaurus) into an initial ontology. In particular, this work introduces algorithms for enhancing the thesaurus connectivity by computing extra subsumption and association relationships. An important characteristic of technical thesauri is that many concept names are compound (multi-word) terms. They implemented a graph enhancement algorithm for this task that automatically inferred these missing links using word-spotting/natural language processing technology. Additionally, they also used natural language processing to refine the *RT* relationship into finer semantic categories.

Another remarkable work with the aim of automating the refinement of relationships is the described by Soergel et al. [188] and Kawtrakul et al. [111]. It introduces a semi-automatic approach for detecting problematic relationships, especially *broader/narrower* and *use/use-for* relationships, and suggesting more appropriate ones. Upon the experience obtained with the transformation of AGROVOC into an ontology, their approach is mainly based on the identification of patterns and the establishment of rules that can automatically applied. The method is based on three main ideas. Firstly, they try to find expert-defined rules. Assuming that concepts are associated with categories (e.g., geographic term, taxonomic term for animals . . .), experts may define rules that can be generally applied to transform *broader/narrower* relationships of concepts under the same category into *is-a* or *is-part-of* hierarchies. Secondly, they propose noun phrase analysis to detect *is-a* hierarchies. If two terms in a *broader/narrower* relationship share the same headword, this relationship can be transformed into *is-a*. Alternatively, if two terms are in the same hierarchy of hypernyms in WordNet, their relationship is also transformed into *is-a*. Thirdly, in the case of *related* relationships, which usually are under-specified relationships, refinement rules, acquired from experts and machine learning, are applied. If we identify a particular case of conversion of an *related* relationship between two terms, we may derive a general rule for the hypernyms of these two particular terms and apply it again to all their hyponyms related through a *related* relationship.

3.5.2 Proposed formalization process

With the objective of testing the complexity and the viability of the existent formalization approaches for its use inside the global management context of terminological ontologies, a simple formalization prototype has been developed. The prototype focuses on the automatic identification of *is-a* relationships given that their correct identification is the minimum requirement to have a formal ontology.

The purpose of the developed system is not to find all the existent *is-a* relationships in the source terminological model (a thesaurus). The objective is to determine a lower limit (as closest as possible to the real one) of the percentage of the source model structure that can be directly translated into a formal model without additional restructuring. That is, the objective is to obtain a measure of the inherent formalism of the analyzed thesaurus. Additionally, as consequence of this analysis it can be deducted which areas of the thesaurus require little effort and which ones require a complete transformation.

In this context, the identification of a *broader/narrower* relationship between two concepts as a *is-a* relationship has been done using the heuristic described in algorithm 8. According to this algorithm, a *broader/narrower* relationship is transformed into an *is-a* relationship if the narrower concept has the same headword (substantive) in at least one of their labels (preferred or alternatives) in any of the available languages (in this case only Spanish and English have been considered). Here, the language difference is critical because the position of the headword is completely language dependent. Additionally, in order to simplify the identification of equivalences between the headwords, they are processed to remove the plurals. The relations that are non identified as *is-a* relationships are left as *narrower/broader* relationships. Their formalization is left for future work.

Figure 3.18 shows the set of steps that are performed to formalize the source thesaurus. The result of the formalization process is an OWL file that contains the formalized thesaurus. The formalization process transforms each source concept into an OWL class (it is modified to facilitate its visualization by formal ontology management tools such as Protégé[19]). The preferred and alternative labels of each source concept are stored as rdfs:label properties. With respect to the relations, the identified *is-a* relationships are represented as inheritances between OWL classes. The rest of them (non formalized relations) are generated as named *owl:ObjectProperty* (generating a different identifier for each one).

3.5.3 Testing the method

To test the suitability of the formalization for thesauri, two different analysis have been performed. On the one hand, it has been applied to a set of thesaurus (GEMET, AGROVOC, EUROVOC, UNESCO) also used for other experiments in this thesis, in order to detect if the

[19] http://protege.stanford.edu/

```
Procedure boolean detectIsARelations(Resource res1, Resource res2);
begin
    List res1Labels = getPrefAndAltLabels(res1);
    List res2Labels = getPrefAndAltLabels(res2);
    for int i=1;i¡res1Labels.size(); i++ do
        LabelStructure lR1 = res1Labels.get(i);
        String langR1=lR1.getLanguage();
        for int k=1;k¡res2Labels.size(); k++ do
            LabelStructure lR2 = res2Labels.get(k);
            String langR2=lR2.getLanguage();
            String[] campos1 = l1.getLabelString().split(” ”);
            String[] campos2 = l2.getLabelString().split(” ”);
            if lang1.equalsIgnoreCase(”en”) && lang1.equalsIgnoreCase(lang2) then
                if removeEnglishPlural(campos1[campos1.length-1]).
                equalsIgnoreCase(removeEnglishPlural(campos2[campos2.length-1])) then
                    return true
                end
            else
                if lang1.equalsIgnoreCase(”es”) && lang1.equalsIgnoreCase(lang2) then
                    if removeSpanishPlural(campos1[0]).
                    equalsIgnoreCase(removeSpanishPlural(campos2[0])) then
                        return true
                    end
                end
            end
        end
    end
    return false
end
```

Algorithm 8: Identification of *is-a* relationships

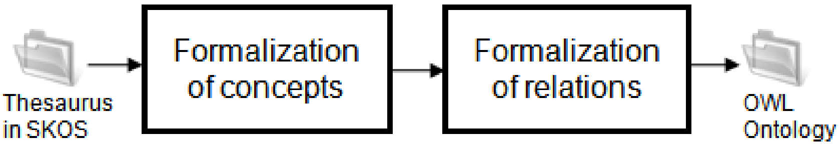


Figure 3.18: Work-flow for the formalization of a terminological ontology

pattern used to identify the *is-a* relationships is effective in general thesauri covering different areas of knowledge. On the other hand, it has been applied to the urban and to the hydrological thesaurus generated in the section 3.4.2 to analyze the structural quality of the resulting models. The process described in section 3.4 allows creating different versions of each generated thesauri by selecting the relations with higher weight (by modifying different parameters of their respective generation processes). This characteristic facilitates the analysis of the formalization suitability in the sense that it allows determining if there is some kind of correlation between the strength of the relationships (number of times they are contained in the generated thesaurus) and the percentage of *is-a* relationships detected in the formalization process (i.e., it allows determining if the most common relations are usually *is-a* relations).

Table 3.11 shows the results of applying the formalization to GEMET, AGROVOC, EUROVOC and UNESCO. It shows for each thesaurus: the number of original *broader/narrower* relationships of each thesaurus; the number of these relations that have been identified as *is-a* relationships (separating those found though the analysis of Spanish labels from those found through the analysis of English ones); and the percentage of the total number of relations that have been transformed into *is-a* relationships.

Thesaurus Name	Nr BT/NT	Nr is-a	% is-a rels	English is-a	Spanish is-a
GEMET	5332	1563	29%	1042	1042
AGROVOC	16579	2498	15%	1379	1722
EUROVOC	7149	2748	38%	1888	1775
UNESCO	14827	699	3%	378	397

Table 3.11: Identification of is-a relations in general thesaurus

These results show that there is an important percentage of relations in the thesaurus (between a 15% and a 38%) that follow the heuristic applied in the formalization process. In this context, the cases of GEMET and EUROVOC are specially relevant because around a third of the concepts in these thesauri can be formalized by applying a simple heuristic. That is, it can be said that they have a better organization and their formalization would be easier.

Table 3.12 shows the results of the formalization of the different urban thesaurus versions described in section 3.4.2. Each version is identified with the weight threshold used to generate it. The rest of the table has the same structure as table 3.11. That is, the number of original *broader/narrower* relationships of each thesaurus; the number of these relations that have been identified as *is-a* relationships (separating those found though the analysis of Spanish labels from those found through the analysis of English ones); and the percentage of the total number of relations that have been transformed into *is-a* relationships.

Weight threshold	Nr BT/NT	Nr is-a	% is-a rels	English is-a	Spanish is-a
1	4281	883	20%	685	625
2	1482	686	46%	535	508
3	852	453	53%	375	325
4	550	307	56%	248	241
5	317	191	60%	157	146

Table 3.12: Identification of is-a relations in the urbanism thesaurus

From the results obtained it can be seen that the percentage of identified *is-a* relationships increases with the weight of the considered relationships. That is, stronger relationships have a much higher chance to be *is-a* relationships than weaker ones.

The comparison of the formalization results of the different version of the generated thesauri allows determining which one of the generated models provides a better balance between the number of generated concepts and the underlying formal structure. This information can be used to improve the selection of the most suitable thesaurus version performed in section 3.4.2. Previously, the only available criteria for the selection of the most suitable thesaurus version were the number of concepts, and a measure of quality obtained though a manual revision of the model. In this context, the measure of the underlying formalism can be used as a third selection criteria. It can be considered that a thesaurus model which a higher percentage of underlying *is-a* relationships is better organized than another one more heterogeneous and therefore it is more suitable for classification use. Additionally, if it is expected to generate on a near future a formal model based on the thesaurus, the better its structure is the easier its latter transform will be.

If this additional criteria is applied to urban thesauri, it can be seen that the most suitable model is the one considering the relations of weight 2. The obtained data shows that using this weight (only the relation that are contained once in the source are removed) the percentage of *is-a* relations jumps for a 20 percent to almost a half of the obtained relationships. A lower weight provides a very heterogeneous structure and a higher one, reduces in a great extent the number of concepts included in the model.

Finally, table 3.13 shows the results of the formalization of the different hydrology thesaurus versions described in section 3.4.2. The structure of table 3.13 is the same as table 3.12. That is, each version is identified with the weight threshold used to generate it and the information provided is the following: the number of these relations that have been identified as *is-a* relationships (separating those found though the analysis of Spanish labels from those found through the analysis of English ones); and the percentage of the total number of relations that have been transformed into *is-a* relationships.

Weight threshold	Nr BT/NT	Nr is-a	% is-a rels	English is-a	Spanish is-a
1	261	135	51%	110	90
2	158	102	64%	85	63
3	93	64	68%	51	54
4	56	42	75%	35	33
5	29	25	86%	23	22

Table 3.13: Identification of is-a relations in the hydrology thesaurus

A similar analysis to the one performed for the urban thesaurus has been done for the hydrology thesaurus. In this case, the results obtained are much better. In the complete thesaurus (without pruning) half of the relations have been identified as *is-a*. And if the less relevant are pruned, this percentage can be increased up to a 86%. In this case, due to the reduced size of the thesaurus and the high number of *is-a* relations identified, the hydrological thesaurus obtained without performing any prune should be the selected.

3.6 Conclusions

This chapter has focused on the problem of obtaining terminological ontologies suitable for the required purpose.

In this context, the first problem identified has been the format heterogeneity of existent terminological models. Ad-hoc formats are very common and extensions to the models are frequently created. Due to the need of a harmonized management of ontologies, this chapter has proposed a methodology to translate input models into a common format (the SKOS based format proposed in section 2.4.1). This process is based on the detailed description of the source and target terminological models. Once these models have been properly described, they are used to construct the translation tool. This construction is simplified thanks to the proposed architectural pattern that divides a translation tool into three separated components: a reader that access to the source format; a matcher that translates the source model to SKOS model; and a writer that writes the target SKOS file. Using this pattern, the task of creation a new translation tool is reduced to the definition of a *reader* component of the source model and the creation of a set of conversion functions to facilitate to a *matcher* component the translation of the model (the writer is common for all the translation tools).

Around 70 different terminological ontologies have been translated to SKOS using this process. From them, most of the translated terminological ontologies are simple controlled vocabularies used for classification purposes, such as the controlled lists contained in ISO-19115, ISO-19119, and CSDGM-FGDC standards. The rest of them are mainly taxonomies and thesauri such as the Spanish and French administrative units models, AGROVOC, EUROVOC, GEMET, URBISOC[20] or UNESCO thesauri. There are also some authority files such as the ISO-639 language code list, and the set of EPSG codes for coordinate reference systems (including datums, ellipsoids and projections).

The second area of interest analyzed in this chapter has been the problem caused by the use of different and overlapping terminological ontologies in collections that must be integrated. The solution proposed in this chapter to manage this issue has been the use of a lexical database as a nexus between the required terminological ontologies. The terminological ontologies are matched to the lexical database to provide its integration. Using the disambiguation algorithm against the WordNet lexical database described in [163] as an initial point, this chapter has proposed a generalization of this method to solve two main problems. Firstly, WordNet cannot be used to relate all kinds of vocabularies given that it is quite general and does not contain many specialized terminology. To solve this problem it has been needed to generalize the described process to be able to relate terminological ontologies with lexical models different to WordNet focused on the subject of the models to relate. Secondly, the original input and the output of the matching system were not reusable. On the one hand, the thesaurus to match had to be provided in an ad-hoc format. On the other hand, there was a lack of representation of the defined matchings. The input problem has been resolved by modifying the disambiguation system to accept SKOS files as input. With respect to the output problem, this issue has been solved by modifying the disambiguation system to establish as output the updated version of SKOS-Mapping described in section 2.4.3. As proposed in the representation framework introduced in section 2.4, this interchange format facilitates the distribution of the generated

[20] http://thes.cindoc.csic.es/index_URBA_esp.html

mappings along an information infrastructure.

The third area of interest has been focused on the analysis of solutions that can be adopted when a terminological ontology with the required structure and content does not exist and has to be created. Due to the costs of creating a new ontology from scratch, the approach has been to take other existent terminological models as base and reuse as much as possible from them. The use of several terminological ontologies (in this case thesauri) covering the same area of knowledge helps to obtain a more general interpretation of the desired domain. The different views provided by each terminological model complement each other. If a set of concepts and relations between them can be found in most of the source models, it indicates that the organization of these subsets of knowledge is generally recognized and it is quite stable. The main steps of the generation process are the harmonization of the input formats, the mapping between the concepts to generate clusters of equivalent concepts using linguistic similarity measures, the establishment of relations between the clusters on the basis of the original relations between the concepts contained in different clusters, the selection of the set of relations that conform a thesaurus structure, and the storage of the new generated model in SKOS format. This process facilitates the creation of different thesaurus versions through the selection of different prune conditions for the cluster relationships.

The process has been tested to generate a thesaurus of urbanism and another one in the context of hydrology. The domain thesauri obtained as a result of the method proposed has several advantages in comparison with the models used as source in the following areas:

- Consensus and focus: The concepts of the resulting network have been selected by consensus thanks to the mappings among the different sources, removing those concepts that are neither common nor focused on urbanism.

- Relations: With respect to the relation structure, the total number of available relations is bigger than the existent ones in each of the original sources. Besides each relation has a weight that indicates its relevance. As future work, the semantics of these relations should be enriched. The information provided by definitions, examples, and naming patterns in the properties of the original concepts should help to refine the current relations (e.g., broader relations could be refined as *is-part-of*, *instance-of* or *generalization* relations).

- Multilingual support: Thanks to the combination of different sources of knowledge with multilingual support, the output network is enriched with alternative terminology in different languages.

Finally, the last area analyzed is related to the need of using of formal ontologies with formal *is-a* hierarchies, frame definitions or even general logical constraints. There are systems that require improved user interaction with the ontology on the conceptual, intelligent behind-the-scenes support for query expansion, improved indexing/categorization systems, or support for artificial intelligence. And for these purposes, the use of terminological ontologies is not enough. However, since the creation of formal ontologies is even more complex than the use of terminological ones, the use of simpler models for its construction is expected to save a lot of effort.

This chapter has proposed a method to determine the feasibility of constructing formal models through the automatic formalization of terminological ontologies. The method focuses on the deriving of *is-a* relationships from the *broader/narrower* relationships of thesauri. The objective is to find how much the original model structure directly fit into a formal model. This prototype has been tested with the set of thematic thesauri used along the thesis (see section 2.2.1.4), and the different versions of urban and hydrology thesauri generated in the previous merging. The results obtained for the thematic thesaurus are quite heterogeneous: they range from a minimum 13% of UNESCO thesaurus to a maximum 38% of EUROVOC. These results show that the complexity of the formalization depends greatly on the structure of each processed model (it is easier to formalize the EUROVOC than the UNESCO thesaurus). With respect to the urbanism and hydrology thesaurus, the comparison of the results obtained with each one of the generated versions (according to different prune levels) clearly show that the most common relations have a much higher chance to be an *is-a* relationships. Additionally, the obtained results can be used as a measure of the structural quality of each thesaurus. They can be used as an additional factor in the decision of the thesaurus to select for each purpose.

Chapter 4

Access to terminological ontologies

4.1 Introduction

Typical information infrastructures, and SDIs are not an exception, manage different terminological ontologies such as glossaries, taxonomies or thesauri integrated in the different components. Due to the multidisciplinary character of SDIs and its applicability to a wide range of application domains, there is a great variety of terminological models with very different levels of specificity, language coverage, i.e. from monolingual list of terms to multilingual thesauri covering more than 20 languages; formalization, i.e. from simple glossaries to well-structured thesauri; or size, e.g. AGROVOC thesaurus [130] contains more than 16,000 concepts.

These different ontology models have been traditionally stored, managed, used and updated independently from the rest of them. There is no coordinate management, making very difficult to determine which models are used in which service and with which purpose. Additionally, this lack of coordination leads to the replication of the terminological ontologies along the infrastructure. Ensuring that all the copies of each ontology are the same requires a lot of maintenance effort each time there is an update of version. Moreover, the lack of coordination makes the update process very prone to errors. For example, it is very easy to forget to update one of the copies; and if this happen, the service using the obsolete terminology would produce deficient results. Additionally, the lack of knowledge of the used ontologies leads to a dispersion in their use. For instance, different but similar terminological ontologies are used for situations where a single one would be a better selection.

When ontologies are provided to the public by their creators, they are distributed through ad-hoc services developed for the institution providing each ontology. This may be useful for applications requiring access to a single ontology. However, in systems such as SDIs, which use a great amount of them, it is not viable to access a large amount of different incompatible services to retrieve the needed vocabularies. It is needed to collect all the required models obtained from external and internal sources and provide them homogeneously, using a single inner model, storage system and access procedures.

Section 4.2 reviews the existent works in this area to determine if the proposed solutions cover all the required management needs. The revision has shown that existent solutions tackle specific management problems independently of the rest. In this context, with the objective of managing terminological ontologies and having control of their life cycle in a homogeneous way, section 4.3 describes the general architecture that has been proposed for the management of terminological ontologies. Each one of the following sections describes in detail a specific component of this architecture. Firstly, section 4.4 focuses on the need of a common repository to store all the required terminological ontologies and makes a proposal for such a repository. Secondly, at management level, section 4.5 describes the management needs for terminological models and proposes solution for an efficient management. Finally, section 4.6 focuses on access and describes the requirements and structure of a Web service to facilitate the distribution of the most suitable terminological models to the components of an information infrastructure. An additional requirement that has been considered in this section has been the desired full integration within a typical SDI. Therefore, the defined elements have been designed to follow the standard interfaces used in the geospatial community.

The components described in this management architecture have been implemented and tested to verify their suitability for working with terminological ontologies. Section 4.7 shows the experiments performed and the results obtained.

4.2 State of the art in ontology creation, management and access

Because of the variety and large number of terminological models used in an information infrastructure, the harmonization of their management is a priority. The interest in creating terminological ontologies in the digital libraries field and other related disciplines has led to an increasing number of software packages for the construction of different types of terminological models. The web site of Willpower Information[1] offers a detailed revision of more than 40 tools, most of them designed for thesauri edition. Some tools are only available as a module of a complete information storage and retrieval system, but others also allow the possibility of working independently of any other software. Among these creation tools, one may highlight the following products:

- BiblioTech[2]. This is a multi-platform tool that forms part of BiblioTech PRO Integrated Library System and can be used to build an ANSI/NISO standard thesaurus [9].

[1] http://www.willpower.demon.co.uk/thessoft.htm
[2] http://www.inmagic.com/

- Lexico[3]. This is a Java-based tool that can be accessed and/or manipulated over the Internet. Thesauri are saved in a text based format. It has been used by the U.S. Library of Congress to manage vocabularies and thesauri such as: the "Thesaurus for Graphic Materials", the "Global Legal Information Network Thesaurus", the "Legislative Indexing Vocabulary" and the "Symbols of American Libraries Listing".

- MultiTes[4]. This is a windows based tool that provides support for ANSI/NISO relationships plus user defined relationships and comment fields for an unlimited number of thesauri (both monolingual and multilingual).

- TermTree 2000[5]. TermTree is a windows based tool that uses Access, SQL Server or Oracle for data storage. It can import/export TRIM thesauri (format used by the Towers Records Information Management system[6]) as well as a defined TermTree 2000 tag format.

- WebChoir[7]. WebChoir is a family of client-server web applications that provide different utilities for thesaurus management in multiple DBMS platforms. TermChoir is a hierarchical information organizing and searching tool that enables to create and search varieties of hierarchical subject categories, controlled vocabularies, and taxonomies based on either pre-defined standards or a user-defined structure and exported to an XML based format. LinkChoir is another tool that allows indexers to describe information sources using terminology organized in TermChoir. And SeekChoir is a retrieval system that enables users to browse thesaurus descriptors and their references (broader terms, related terms, synonyms . . .).

- Synaptica[8]. Synaptica is a client-server web application that can be installed locally on a client's intranet or extranet server. Thesaurus data is stored in a SQL Server or Oracle database. The application supports the creation of electronic thesauri in compliance with ANSI/NISO standard. The application allows the exchange of thesauri in CSV (Comma-Separated Values) text format.

- SuperThes [13]. SuperThes is a windows based tool that allows the creation of thesauri. It extends the ANSI/NISO relationships allowing many possible data types to enrich the properties of a concept. It can import/export thesauri in XML and in tabular format.

- TemaTRES[9]. TemaTres is a web application specially oriented to the creation of thesauri, but it also can be used to develop web browsing structures or to manage the documentary

[3] http://www.pmei.com/lexico.html
[4] http://www.multites.com/
[5] http://www.termtree.com.au/
[6] http://www.towersoft.com/
[7] http://www.webchoir.com
[8] http://www.synaptica.com/
[9] http://www.r020.com.ar/tematres/

languages in use. The thesauri are stored in a MySQL database. It provides the created thesauri in Zthes format [10] or in SKOS format.

In addition to the tools specifically designed for editing terminological ontologies such as thesauri, general ontology editors can also be used. In this context, Denny [40] describes a detailed survey of general ontology editors. Other edition alternatives are the family of RDF edition tools such as SWOOP [108], Protégé [165] or Triple20 [219]. However, these general editing ontology tools do not provide specific adapted interfaces to create ontologies following a specific model. Additionally, they are too complex as they provide too many options and capabilities not needed in terminological ontology models.

Having all the required terminological models available, the homogenization of their management becomes a priority. The Canadian Geospatial Data Infrastructure project [11] advanced in 1999 that an SDI would need a centralized ontology service with the objective of providing a mechanism to maintain terminological ontologies when the number to manage would increase. In 2004 they published a prototype of a web service, the Multilingual Geospatial Ontology (M3GO)[12], with some limitation in the relations that it could manage and the ways to identify ontologies. However, the main standardization organizations such as ISO or OpenGIS have not provided any specification to develop this kind of service.

Uniform management of terminological models, is not a simple task, it is needed to be able to share them to other components of the infrastructure, and identify them adequately to be able to locate the required one. Additionally, the size of this kind of models creates the need to provide additional communication protocols to access to the required area instead to the whole ontology.

Traditionally, in the information community, the classical approach to share terminological ontologies has been to create different ad-hoc web services that provide access to a particular ontology. Some examples of this kind of service are the General Multilingual Environmental Thesaurus[13] (GEMET), the Agriculture vocabulary[14] (AGROVOC) of the Food and Agricultural Organization of the United Nations (FAO) or the Alexandria Digital Library Feature Type Thesaurus[15].

Other protocols and systems that provide access to a knowledge model are CERES[16], provided by the California Resources Agency, or ADL Thesaurus protocol[17] created by the Alexandria Digital Library project. The work described by Binding and Tudhope [19] and Tudhope

[10] http://zthes.z3950.org/schema/index.html

[11] http://www.geoconnections.org/CGDI.cfm

[12] http://www.geoconnections.org/projects/geoinnovations/2002/INTELEC/Index_f%20OGM3%20v1.0.html

[13] http://www.eionet.europa.eu/gemet

[14] http://www.fao.org/aims/ag_intro.htm

[15] http://www.alexandria.ucsb.edu/

[16] http://ceres.ca.gov/thesaurus/

[17] http://www.alexandria.ucsb.edu/ gjanee/thesaurus/

and Binding [202], that shows a web service to provide access to a thesaurus, providing access to concepts and stored relations.

These services provide access to a single terminological model through their own ad-hoc communication protocol. Each protocol is different from the rest making nonviable its direct integration in systems using multiple terminological ontologies. Integration problems are partially solved by using standardized approaches for providing access to the models. An example of a standardized access protocol is Zthes[18], a Z39.50 [8] profile for thesaurus representation, access and browsing.

The use of a standardized access protocol for terminological models can be a solution for systems using only a few ontologies, but systems using lots of models cannot afford creating different services for each one. A homogeneous solution that integrates the access into a single service is required. The most recent works in this area go in this direction. The Simple Knowledge Organization System (SKOS) project [19] created by World Wide Web Consortium (W3C) Semantic Web Activity [20] has published a prototype of a web service [21] to provide access to a repository of terminological ontologies. Other access service for terminological models is the created in the STAR Project[22] to provide term look up in vocabularies known to the system, browsing and semantic concept expansion.

In the area of formal ontologies there is also the same concern with respect to the need to reuse existent ontologies. Swoogle [42] is a crawler-based indexing and retrieval system for the Semantic Web that allow searching for Web documents in RDF or OWL (that is, ontologies). It extracts metadata for each discovered document, and computes relations between documents. Watson [36] goes in the same direction by collecting and giving access to ontologies and semantic data available online. Oyster [172] is a bit different in the sense that it focuses on the description of the ontologies as the way to provide a better access. The access is not provided through a traditionally web service but through a Peer-to-Peer network. Other different approaches for accessing formal ontologies are KAON [212] or Ontolingua [50]. They are complex infrastructures which have been designed to share ontologies in general contexts. They include different components such as editors or reasoners that access to the repository containing the stored the ontologies through a predefined API,

The problem with all these solutions is the lack of search capabilities to find the desired terminological ontology. SKOS server does not provide any facility to locate the desired ontology, and the same happens in the KAON tool that identifies the desired ontology by means of a URI. A bit better is Ontolingua and the STAR project served where users have a list of ontologies with the name plus a short description.

[18] http://zthes.z3950.org/z3950/zthes-z3950-1.0.html

[19] http://www.w3.org/2004/02/skos/

[20] http://www.w3.org/2001/sw/Europe/reports/thes/

[21] http://www.w3.org/2001/sw/Europe/reports/thes/skosapi.html

[22] http://hypermedia.research.glam.ac.uk/kos/terminology_services/

All these projects have identified that the use of a centralized service is a step forward to facilitate access and management of ontologies in complex information infrastructures; however, the lack of discovery services for the contained models (e.g., search for the ontology name, creation date or description) limits its integration possibilities in systems requiring a dozens of terminological models. In these systems, it is required to provide the most suitable model for each context; therefore, they have to be detailed described to be able to locate the required ones. In addition, the use of these ontologies to the SDI context add new requirements not needed in more general services, such as searches of ontologies focused on a specific geographic area, application domain or even data creator. Moreover, the integration of the service in an SDI imposes structural restrictions needed to fit with the general architecture of the infrastructure.

4.3 Architecture of the management and access proposal

In order to advance in the improvement of the management of terminological ontologies in infrastructures with an important need of access to these types of models, the architecture of components shown in figure 4.1 is proposed.

The system consists of a core component called *ontology manager* that provides the storage for the terminological models and a software components that provide the access to the *ontology repository*, allowing the addition, removal and update of the desired ontologies. The manager provides a metadata oriented access to the terminologies. Each model has to be described and identified using metadata. The metadata are then used to provide a search system in the repository simplifying the location of the required ontology. Additionally, the manager provides an interrelation functionality to provide equivalences between the ontology concepts. These equivalences can then used, for example, to expand the user queries with similar concepts in a search system with access to the repository of terminological ontologies. To provide access to the repository, the *ontology manager* provides an API with all the required access functionality. The *ontology repository* stores the terminological ontologies, their metadata and additional components used for interrelation of models.

With the objective of controlling the contents of the repository and making the required changes when needed, on top of the *ontology manager* a desktop editor (ThManager) that allows visualizing, adding, updating, deleting and modifying the terminological ontologies and their metadata descriptions has been designed. This desktop editor follows an architectural pattern that simplifies the creation and composition of different graphical components for the visualization and modification of terminological models.

Finally, to provide access to terminological ontologies to components requiring it, a web ontology service (WOS) that provides the access to the ontologies stored in the repository through the *Ontology Manager API* has been designed. It offers different interfaces to cover a broad range of applications with different functionality requirements. Some possible client applications (e.g., query expansion, browsing) are described in chapter 5.

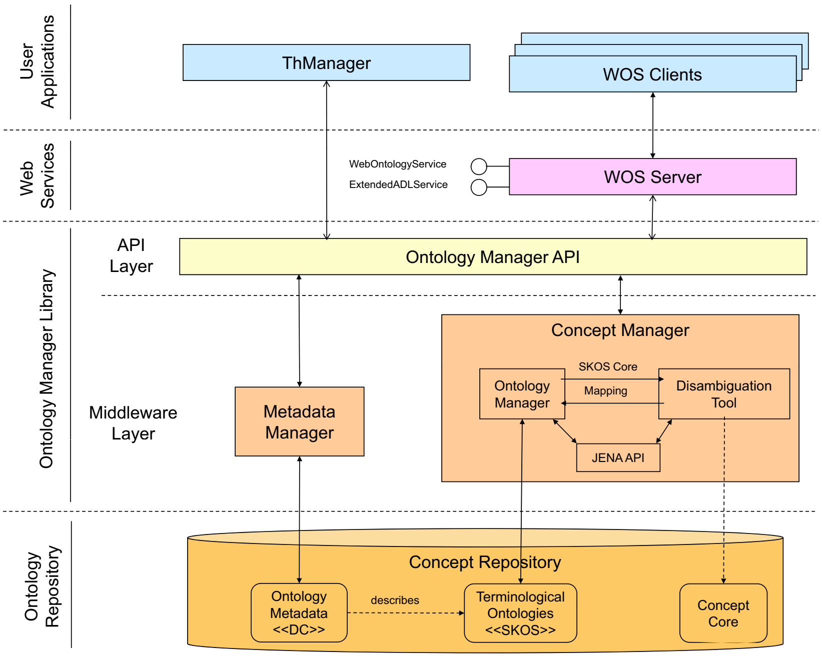


Figure 4.1: Architecture of the management and access proposal

4.4 Ontology repository and management proposal

The first step to provide a harmonized access to terminological models has been to create a repository for the ontologies and a management component able to provide access to the stored models. The repository layer stores not only the terminological ontologies (concepts and relations) but also the metadata describing them and a concept core used for the interconnection of ontologies. With respect to the *Ontology Manager*, its main function is to provide access to a repository. It is used for the *ThManager* and the *Web Ontology Service* to harmonize the access to the ontologies. As it is shown in figure 4.1, the *Ontology Managemer* consists of two main layers: the middleware layer that is in charge of accessing the repository and providing the interrelation capability; and the API layer that provides the external interface for tools requiring access to its functionality.

4.4.1 Ontology repository

As proposed in the representation framework presented in section 2.4, the format selected for terminological ontologies is based on SKOS. As stated in section 2.3.1, SKOS is a RDF based model created specifically to manage simple Knowledge Organization systems for the W3C Semantic Web project. Widely accepted within the digital library community, SKOS provides a very reach machine readable language for representing terminological ontologies such as subject heading lists, taxonomies, classification schemes, thesauri, folksonomies, and other types of controlled vocabularies.

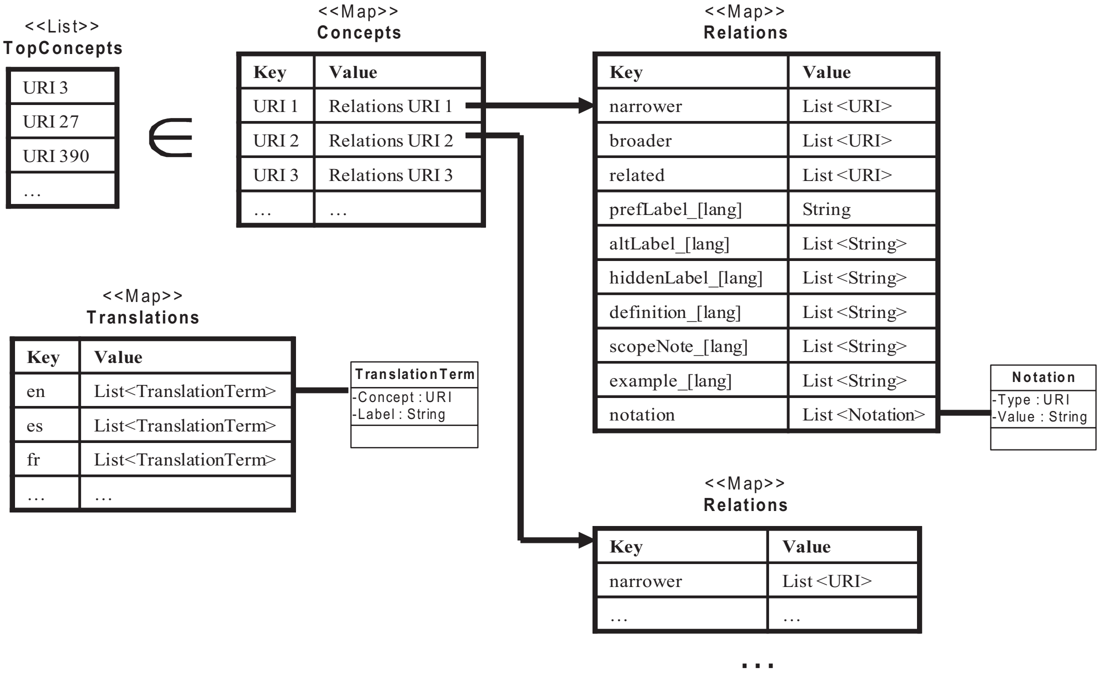
The access to RDF SKOS documents storing terminological ontologies is provided in the application layer through *Jena*[23]. *Jena* is a popular library that simplifies the manipulation of RDF documents, storing them in text files or in a relational database. One important advantage of using *Jena* is that it has an open model that can be extended with specialized modules to provide other ways of storage such as the *Jena-Sesame adapter*[24], which provides access to *Sesame*[25] databases.

Terminological models vary enormously in size, ranging from hundreds of concepts and properties up to hundreds of thousands. Therefore, the time spent on load, browsing and search processes are a functional restriction in their management. The use of Sesame as storage model provides an acceptable performance. However, for those situations where no external database access is required (e.g., simple desktop tools), the use of the default Jena component to directly load the SKOS files with the terminologies is very inefficient. SKOS is RDF based, and reading RDF and extracting the content is a slow process for terminological models of big size. To provide better access time, an alternative storage format has been created. Instead of storing the ontologies in SKOS, they are transformed into a binary format when a new ontology (in SKOS format) is added to the repository.

The data structure used in the developed storage model is shown in figure 4.2. The model is an optimized representation of the RDF triplet structure. The *Concepts* map contains the concepts and their associated relations in the form of key-value pairs: the key is a URI identifying a concept; and the value is a *Relations* object containing the properties of the concept. A *Relations* object is a map that stores the properties of one concept in the form of <property type, property values> pairs. The keys used for this map are the names of the property types in the SKOS model (e.g., narrower, or broader). The only special cases for encoding these property types occur when the property has a language attribute (e.g., *prefLabel*, *definition*, or *scopeNote*). In those cases we propose the use of a [lang] suffix to distinguish the property type for a particular language. For instance, *prefLabel_en* indicates a *prefLabel* property whose content is in English. Additionally, it must be noted that the data type of the property values assigned to each key in the *Relations* map varies upon the semantics given to each property type. The data types used fall into the following categories: a string for a *prefLabel* property type; a list of strings for *altLabel*, *hiddenLabel*, *definition*, *scope note*, and *example* property

[23] http://jena.sourceforge.net/
[24] http://sjadapter.sourceforge.net/
[25] http://www.openrdf.org/

types; a list of URIs for *narrower*, *broader* and *related* property types; and a list of *Notation* objects for a *notation* property type. The data type used for *notation* values is a complex object because there may be different *notation* types. A *Notation* object consists of type and value attributes. The type attribute is a URI which identifies a particular notation type and qualifies the associated *notation* value. Some other elements of the SKOS model such as *collections* and different types of *notes* have been omitted given its lack of use in the managed ontology models, but they could be easily added to the model and supported in the repository if required.



TopConcepts
URI 3
URI 27
URI 390
…

Key	Value
URI 1	Relations URI 1
URI 2	Relations URI 2
URI 3	Relations URI 3
…	…

Key	Value
narrower	List <URI>
broader	List <URI>
related	List <URI>
prefLabel_[lang]	String
altLabel_[lang]	List <String>
hiddenLabel_[lang]	List <String>
definition_[lang]	List <String>
scopeNote_[lang]	List <String>
example_[lang]	List <String>
notation	List <Notation>

Key	Value
en	List<TranslationTerm>
es	List<TranslationTerm>
fr	List<TranslationTerm>
…	…

Key	Value
narrower	List <URI>
…	…

Figure 4.2: Persistence Model

With the objective of increasing the speed of some operations (e.g. browsing or search), some additional optimizations have been used. Firstly, for terminological models with a hierarchical structure such as taxonomies or thesauri, the URIs of the top concepts are stored in the *TopConcepts* list. This list contains redundant information, because those concepts are also stored in the Concepts map, but it makes immediate the location of the top concepts. Secondly, to simplify and speed up the process of search of concepts, and to reduce the time required to provide the concepts in an alphabetical order, the *Translations* map has been created. This map contains, for each language in which the terminological ontology is, a list of pairs URI, *prefLabel* in that language ordered by the *prefLabel*. *Translations* map also contains redundant information, but the improvements obtained compensate the small overhead in load time (as it is shown later). If no alphabetic viewer and search are needed, this structure can be removed without any collateral effect.

The storage model includes some additional elements not described here, used to represent the interrelations between terminological ontologies. Also related to the interrelation of model is the *concept core* depicted in figure 4.1. All this elements are described in detail in section 4.4.2 where the functionality provided for interrelation of terminological ontologies is described.

This storage solution has proven to be useful to manage the kind of terminological models used in the infrastructures considered in this thesis (the ontology models used do not surpass the size of 50,000 concepts and about 330,000 properties). They can be loaded in a reasonable time allowing an immediate browsing and search. See section 4.7 for a detailed performance analysis.

A fundamental aspect in the repository is the description of ontologies. Metadata for describing ontologies are considered as basic information to be facilitated to clients. These metadata are depicted in figure 4.1 as *Ontology Metadata*. The reason for this metadata-driven interface is that centralized ontology storage is not enough to manage them efficiently. Ontologies must be described and classified to facilitate the selection of the most adequate ontology for each situation. The purpose of these metadata is not only to simplify the terminological ontology location to a user, but also to facilitate the identification of models useful for a specific task in a machine-to-machine communication. The lack of metadata describing them makes very difficult the identification of ontologies provided by other services, producing a low reuse of them in other contexts. Metadata are used in search processes to facilitate ontology retrieval, allowing users to search them not only by an agreed *name*, but also by the *application domain* or the associated *geographical area* among other descriptors. The metadata profile used is the described in section 2.4.1, that is the used in all the components described in this thesis to describe the terminological ontologies. It is a metadata profile base on Dublin Core that adds some metadata elements to the basic Dublin Core Element Set to adjust it to the specific requirements of terminological ontologies.

4.4.2 Ontology manager

The purpose of the components contained in the ontology manager shown in figure 4.1 is to simplify the access to the information stored in the repository to the applications constructed on top of it.

In this context, the ontology manager component (in the middleware layer), has as main objective separate the upper layers from the complexity of the repositories. This simplifies to a great extent the interchange of repositories, allowing the selection of the most suitable one for each situation.

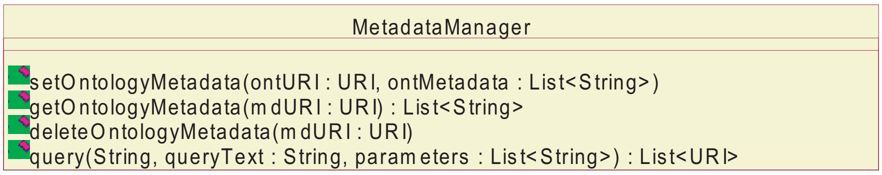
Nowadays, the vast choice of terminological models that are available implies an undesired effect of content heterogeneity. Terminological ontologies usually overlap in their content with other ones defined for different application domains. In order to facilitate cross-domain classification of resources, users would benefit from the possibility of knowing the connections of the stored terminological models. For example, they can be used to extend the query system with equivalent concepts from other models.

In this context, an additionally functionality of the repository layer has been the support of storage and management of interrelation among the stored terminological models. The interrelation is indirectly performed by the *disambiguation tool* and the concept core described in figure 4.1. Its storage is supported through the methods *addConceptCoreInterrelation* and *exportConceptCoreInterrelation*.

The disambiguation component enables the alignment of terminological ontologies with respect to a core upper-level ontology (the concept core displayed in figure 4.1). The algorithm used to establish these alignments is the described in section 3.3. With this relation established, the concept core can be used as a bridge to jump from concepts of one terminological model to equivalent concepts in other ones. At the moment, WordNet [52] has been used as upper-level ontology. As described in section 2.2.1.5, WordNet is a large English lexical database that groups nouns, verbs, adjectives and adverbs into sets of cognitive synonyms (synsets), each expressing a distinct concept. Those synsets are interlinked by means of conceptual-semantic and lexical relations. It is structured in a hierarchy of synsets, defining a synset as a set of strict synonyms representing one underlying lexicalized concept. We have used the name "disambiguation" for this alignment method because the label of a concept in the ontology may be polysemic with respect to the possible synsets that may contain this label in WordNet. Thus, the objective of this disambiguation tool consists in determining which one of the synsets of WordNet can be aligned to the real concept in the terminological ontology. The process is specialized on using WordNet as concept core but it could be easily adapted to use other knowledge repositories such as EuroWordNet [213]. The use of EuroWordNet (A multilingual upper-level ontology) would allow an improved disambiguation of terminologies described in multiple languages.

In addition to the repository of terminological models, it is needed to provide access to the repository storing the metadata that describe the ontologies. This functionality is provided by the *metadata manager* component in the middleware layer. The methods provided are shown in figure 4.3. A query method to search the ontologies for the metadata content is provided, additionally methods, to retrieve, update and delete metadata elements have been included. The metadata contain a reference to the ontology URI in the field *identifier*; therefore, accessing to the metadata allows a user to locate the desired terminological ontology in the ontology repository.

The upper layer in the *ontology manager* is the *API layer* containing the *ontology manager API*. The objective of this API is to unify the functionality provided by the *metadata manager* that gives access to the metadata describing the terminologies; and the *concept manager*



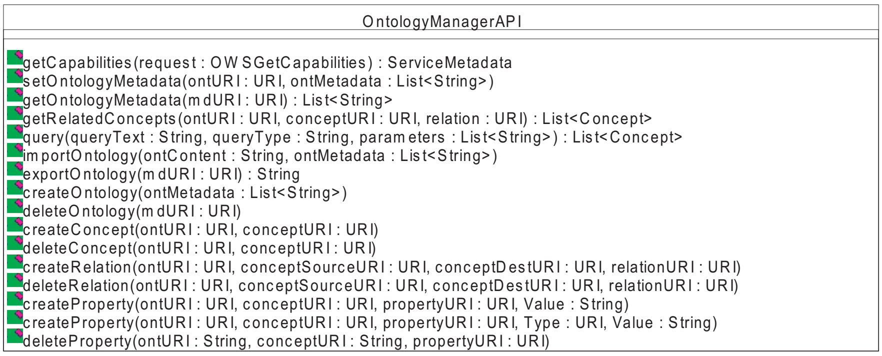
```
MetadataManager

setOntologyMetadata(ontURI : URI, ontMetadata : List<String>)
getOntologyMetadata(mdURI : URI) : List<String>
deleteOntologyMetadata(mdURI : URI)
query(String, queryText : String, parameters : List<String>) : List<URI>
```

Figure 4.3: Metadata manager

that aggregates the *ontology manager* providing access to the ontology repository, and the *disambiguation tool* that interrelates terminological ontologies. The resulting set of procedures includes all the required methods to allow other components to access and manage the stored terminological ontologies managed. These methods, displayed in figure 4.4, can be classified in two categories: query and administration.

- With respect to query methods, *query* and *getRelatedConcepts* methods allow users to browse through the relations between concepts and to search concepts by their label in different languages. The *query* method uses the disambiguation mechanism described before to expand the results returned, providing equivalent terms from the same or different ontologies.

- As regards to administration methods, they allow users to create a new ontology given its metadata, modify its metadata, delete it, and import or export it in SKOS format. Additionally, the API includes methods to update concept properties and relations between concepts from different ontologies.



```
OntologyManagerAPI

getCapabilities(request : OWSGetCapabilities) : ServiceMetadata
setOntologyMetadata(ontURI : URI, ontMetadata : List<String>)
getOntologyMetadata(mdURI : URI) : List<String>
getRelatedConcepts(ontURI : URI, conceptURI : URI, relation : URI) : List<Concept>
query(queryText : String, queryType : String, parameters : List<String>) : List<Concept>
importOntology(ontContent : String, ontMetadata : List<String>)
exportOntology(mdURI : URI) : String
createOntology(ontMetadata : List<String>)
deleteOntology(mdURI : URI)
createConcept(ontURI : URI, conceptURI : URI)
deleteConcept(ontURI : URI, conceptURI : URI)
createRelation(ontURI : URI, conceptSourceURI : URI, conceptDestURI : URI, relationURI : URI)
deleteRelation(ontURI : URI, conceptSourceURI : URI, conceptDestURI : URI, relationURI : URI)
createProperty(ontURI : URI, conceptURI : URI, propertyURI : URI, Value : String)
createProperty(ontURI : URI, conceptURI : URI, propertyURI : URI, Type : URI, Value : String)
deleteProperty(ontURI : String, conceptURI : String, propertyURI : URI)
```

Figure 4.4: Web Ontology Service Implementation

4.5 Design of a terminological ontology editor

Having defined the structure, functionality and access methods to the repository of terminological ontologies, the next step has been to design a editor able to create the required ontologies, to fill the repository, to load the existent ones, and to export them in a suitable format for interchange.

The required terminological ontologies have to be created in a suitable format, updated with the latest changes and described appropriately with their metadata to allow the clients to find them. As commented in the representation framework described in section 2.4, the most accepted format to define simple knowledge models such as terminological ontologies is SKOS. Although SKOS has been recently proposed, the number and importance of organizations involved in its creation process (and that publish their knowledge models in this format) indicates that it will probably become an extended standard for representation or terminological models. SKOS provides a rich machine readable language, but nobody would expect to have to create it just using a general purpose RDF editor (SKOS is RDF based).

Section 4.2 has reviewed different tools for the edition of terminological models. However, none of them is suitable here. Some of them are unable to manage properly structures with tens of thousands of concepts. Others are deeply integrated in bigger systems and cannot easily be reused in other environments because they need specific software components to work (as DBMS to store terminologies). Those that are independent terminological editors have an architecture difficult to integrate within other information management tools. Moreover, they use incompatible interchange formats and do not provide support for SKOS (the format selected in this thesis work for interchange of terminological models). Finally, they are not focused on providing an integrated management of collection of ontologies and their metadata descriptions as is required in this context.

To fill this gap, an editor that facilitates the construction of SKOS based terminological ontologies has been designed. It has been called ThManager and it has been thought to manage thesauri. But it is also appropriate to create and manage any other terminological models that can be represented using SKOS format. It provides a dual functionality; first, an ability to visualize the terminological models in the repository to validate that their content are correct; and second, to facilitate the creation/update of the required ones. To facilitate its use in different computer platforms, ThManager has been developed using the Java object oriented language. Additionally, the tool is distributed as Open Source software accessible through the SourceForge platform[26].

This section presents the architecture of ThManager editor. The main features that have guided ThManager design have been the following: a metadata-driven design, an efficient management of terminological ontologies, the possibility of interrelating them, and the reusability

[26] http://thmanager.sourceforge.net/

of ThManager components.

ThManager has been constructed on top of the *Ontology Manager* described in section 4.4.2 (it provides the access to a repository of terminological ontologies). The complete architecture structure is shown in figure4.5.

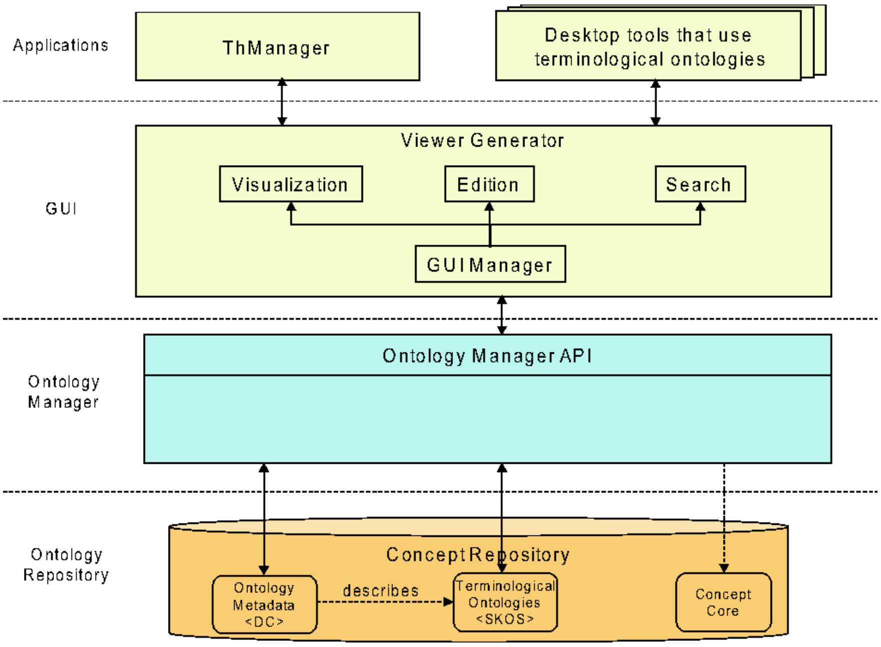


Figure 4.5: ThManager layer architecture

ThManager has a layered architecture to facilitate its integration in desktop tools requiring the access to terminological models. On top of the repository layer (constituted by the *Ontology Manager*), a GUI layer that offers different graphical components has been placed in order to visualize terminological models (specifically thesauri), search by their properties, and edit them in different ways. Among the graphical components, there is a hierarchical viewer, an alphabetic viewer, a list viewer, a searcher and an editor, but more components can be built if needed. The GUI layer is designed as factory of reusable graphical components that makes possible to create tools able to manage terminological models with a minimum effort. Additionally, it also allows the integration of this technology in other applications that need controlled vocabularies to improve their functionality. For example, in a metadata creation tool, it can be used to provide the graphical component to select values from controlled vocabularies and automatically insert them in the metadata.

Figure 4.6 shows the integration process of a thesaurus visualization component in an external tool. The provided thesaurus components have been constructed following the Java Beans philosophy (reusable software components that can be manipulated visually in a builder tool), where a component is a black box with methods to read and change its status that can be reused when needed. Here, each thesaurus component is a *ThesaurusBean* that can be directly inserted in a graphical application to use its functionality (visualize or edit thesauri) in a very simple way. The *ThesaurusBeans* are provided by the *ThesaurusBeanManager* that, given the parameters of the thesaurus to visualize and the type of visualization, returns the most adequate component to use.

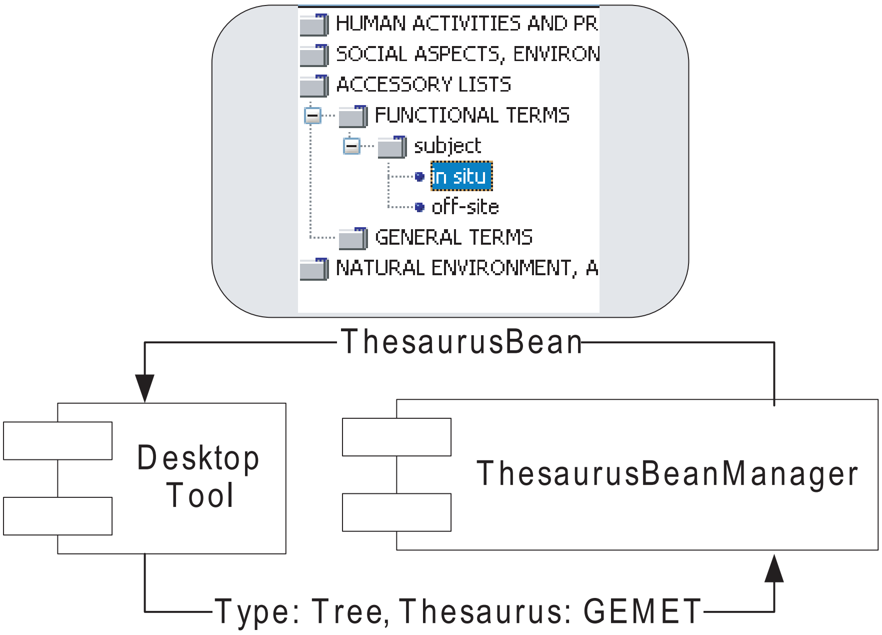
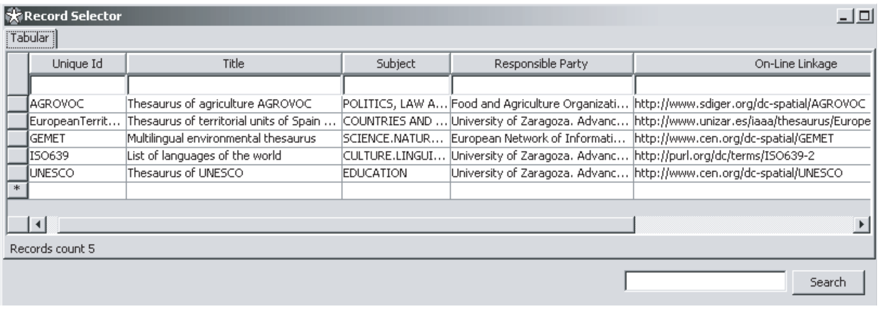


Figure 4.6: GUI component integration

The ThManager editor is an application component that uses the GUI layer elements to provide the user the management and edition of terminological ontologies. The tool groups a subset of the provided components, relating them to obtain a final user application that allows the management of the stored thesauri, their visualization (browsing by the concept relations), their edition and their importation/exportation using SKOS format. In the same way as ThManager has been constructed other tools requiring access to terminological models can use the visualization components or the method provided by the persistence layer to provide access to stored models.

The management of the models stored by the ThManager tool, as it can be deduced by the structure of the repository described in section 4.4, is metadata oriented. The first window visualized by the graphical interface of the editor shows a table including the metadata records describing all the terminological models stored in the system (figure 4.7). The selection of a record is used for the rest of the components of the application to know the model they have to work with.

The creation and deletion of the models is also provided in the same window. The only operation that can be performed when no record is selected is to import a new thesaurus stored in SKOS. To import it, the name of the SKOS file has to be provided. The import tool also contains the option to interrelate the imported terminological model to the concept core (it uses the disambiguation tool described in section 4.4.2). The metadata of the ontology are extracted from inside of the SKOS if they are available, or they can be provided in an associated XML metadata file. If no metadata record is provided, the application generates a new one with minimum information, using the name of the SKOS file as base. Once the user has selected a model, it can visualize and modify its metadata, content, export it to SKOS or, as commented before, delete it.



Unique Id	Title	Subject	Responsible Party	On-Line Linkage
AGROVOC	Thesaurus of agriculture AGROVOC	POLITICS, LAW A...	Food and Agriculture Organizati...	http://www.sdiger.org/dc-spatial/AGROVOC
EuropeanTerrit...	Thesaurus of territorial units of Spain ...	COUNTRIES AND ...	University of Zaragoza. Advanc...	http://www.unizar.es/iaaa/thesaurus/Europe
GEMET	Multilingual environmental thesaurus	SCIENCE.NATUR...	European Network of Informati...	http://www.cen.org/dc-spatial/GEMET
ISO639	List of languages of the world	CULTURE.LINGUI...	University of Zaragoza. Advanc...	http://purl.org/dc/terms/ISO639-2
UNESCO	Thesaurus of UNESCO	EDUCATION	University of Zaragoza. Advanc...	http://www.cen.org/dc-spatial/UNESCO

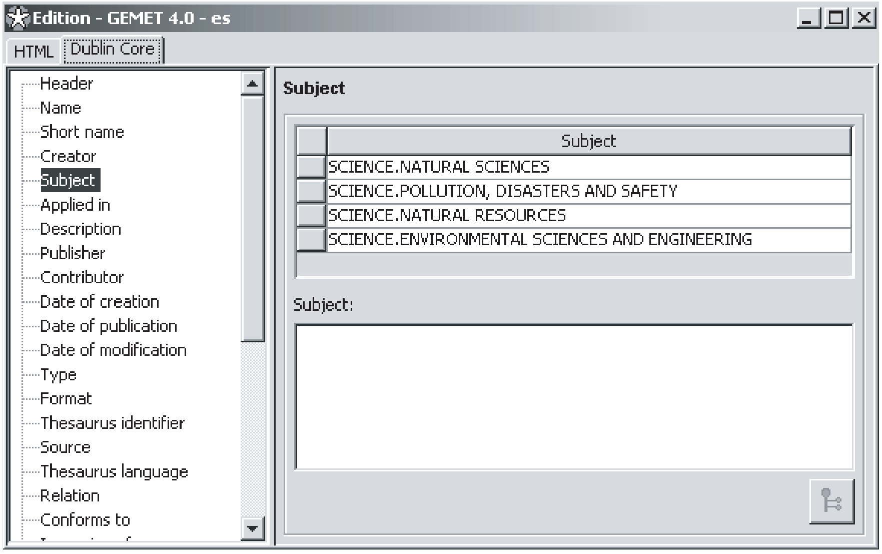
Records count 5

Figure 4.7: General metadata based selector

The metadata describing a terminological ontology is visualized by a metadata viewer in an HTML based view. Different HTML views can be provided by adding different CSS presentation files to the application. The edition is performed by a customizable metadata editor. To add or delete metadata elements to the metadata edition window, it is only needed to modify the description of the used IEMSR profile of the metadata (described in section 2.4.1) included in the application.

One of the basic functionalities of the tool is to visualize the terminological ontologies structure, showing all properties of concepts and allowing the browsing by relations (Figure 4.9). To do this, different read-only viewers are provided. There is an alphabetic viewer, which shows all the concepts ordered by the preferred label in one language. There is also a hierarchical viewer that provides navigation by broader and narrower relations (used for taxonomies and thesauri). And finally, a hypertext viewer shows all properties of a concept and provides navigation by all its relations (e.g. broader, narrower and related) by means of hyperlinks.

To simplify the location of concepts, there is a search system that allows the typical searches needed for terminological modes such as thesauri (equals / starts with / contains). Currently,



Subject
SCIENCE.NATURAL SCIENCES
SCIENCE.POLLUTION, DISASTERS AND SAFETY
SCIENCE.NATURAL RESOURCES
SCIENCE.ENVIRONMENTAL SCIENCES AND ENGINEERING

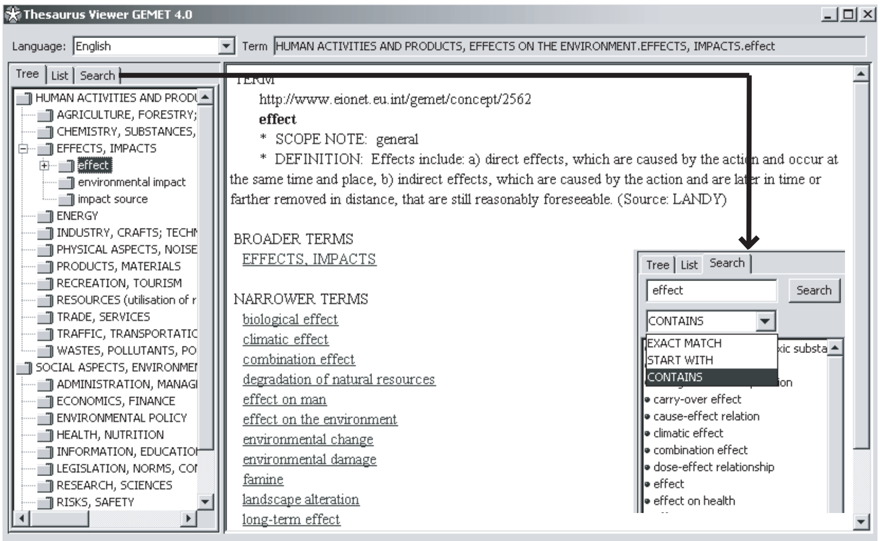
Figure 4.8: Metadata editor

search is limited to preferred labels in the selected language, but it could be extended to allow searches by other properties such as synonyms, definitions or scope notes. All these viewers are synchronized, so the selection of a concept in one of them produces the selection of the same concept in the others.

The third available operation is to edit the model structure. Here, to create terminological ontologies following the SKOS model, an edition component for the ontology concepts is provided (figure 4.10). The graphical interface provided shows a list with all the concepts created in the selected model, allowing the creation of new ones (providing their URIs) or deletion of selected ones. Once a concept has been selected, its properties and relations to other concepts are shown, allowing their edition. To facilitate the creation of relations between concepts, a selector of concepts (one of the previously described terminological ontology viewer) is provided, allowing the user to add related concepts without the need of manually typing the URI of the associated concept. Also, to see if the created thesaurus is correct, a preview of the terminological model can be shown, allowing the user to easily detect problems in the defined relations.

With respect to the interrelation functionality, at the moment, the mapping obtained is shown in the thesaurus viewers but the browsing between equivalent concepts of two thesauri has to be manually done by the user. The development of a browsing component would allow the user to jump from a concept in a terminological ontology to concepts in others that are mapped to the same concept in the common core.

As mentioned in section 4.4, for efficiency, the format used to store the thesauri in the



TERM
http://www.eionet.eu.int/gemet/concept/2562
effect
* SCOPE NOTE: general
* DEFINITION: Effects include: a) direct effects, which are caused by the action and occur at the same time and place, b) indirect effects, which are caused by the action and are later in time or farther removed in distance, that are still reasonably foreseeable. (Source: LANDY)

BROADER TERMS
EFFECTS, IMPACTS

NARROWER TERMS
biological effect
climatic effect
combination effect
degradation of natural resources
effect on man
effect on the environment
environmental change
environmental damage
famine
landscape alteration
long-term effect

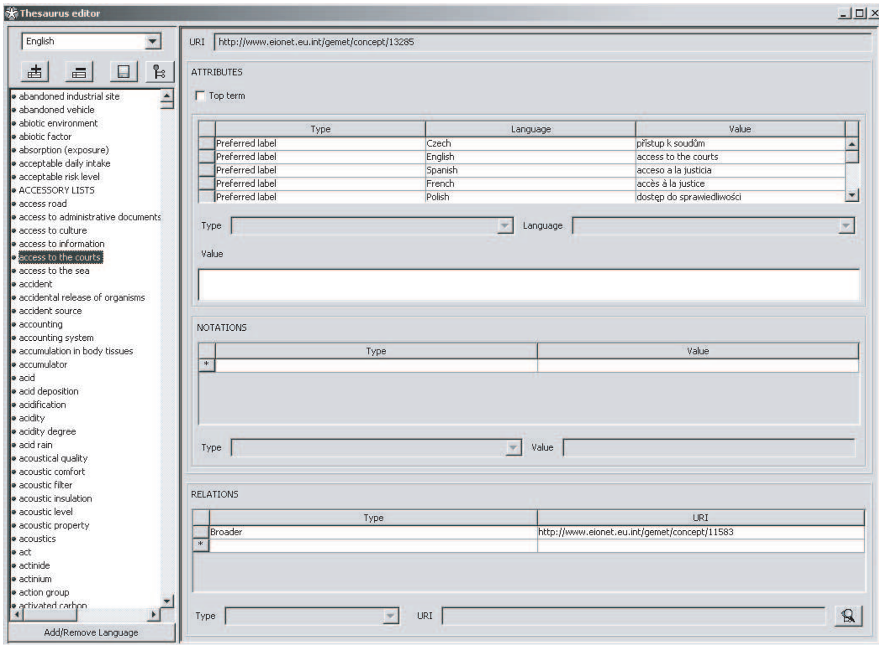
Figure 4.9: Concept selector

repository may change, but the interchange format used is SKOS. So, a module for importation/exportation of the terminological is provided. This module is able to import from SKOS and export to SKOS, but also the interrelation with respect to the concept core (if it has been created) using the format described in section 2.4.3.

The layered architecture described before allows all the different described components to be reused in other contexts. ThManager tool itself reuses some of the components in different areas. For example, the terminological ontology viewer is used in the editor of terminological ontology metadata to facilitate the selection of values for the subject section of metadata. Additionally, it is used in the concept editor to simplify the selection of concepts, providing the list of all the created ones, and a preview of the created terminological model (to help to detect errors in the creation process).

4.6 Accessing terminological ontologies through a web service

Once the repository of terminological ontologies described in section 4.4.1 has been filled with the required models, and once they have been properly described and updated to fit with the required functionality, the last step is to provide these ontologies to the services requiring them. The access needs can be multiple: they can range from the need of a list of possible values for specifying criteria in a web search system, to a thesaurus based browsing of a collection of resources in an exploratory search system. This section describes a web service specification



English

- abandoned industrial site
- abandoned vehicle
- abiotic environment
- abiotic factor
- absorption (exposure)
- acceptable daily intake
- acceptable risk level
- ACCESSORY LISTS
- access road
- access to administrative documents
- access to culture
- access to information
- access to the courts
- access to the sea
- accident
- accidental release of organisms
- accident source
- accounting
- accounting system
- accumulation in body tissues
- accumulator
- acid
- acid deposition
- acidification
- acidity
- acidity degree
- acid rain
- acoustical quality
- acoustic comfort
- acoustic filter
- acoustic insulation
- acoustic level
- acoustic property
- acoustics
- act
- actinide
- actinium
- action group
- activated carbon

URI http://www.eionet.eu.int/gemet/concept/13285

☐ Top term

Type	Language	Value
Preferred label	Czech	přístup k soudům
Preferred label	English	access to the courts
Preferred label	Spanish	acceso a la justicia
Preferred label	French	accès à la justice
Preferred label	Polish	dostęp do sprawiedliwości

Type Language

Value

Type	Value

Type Value

Type	URI
Broader	http://www.eionet.eu.int/gemet/concept/11583

Type URI

Figure 4.10: Concept editor

called *Web Ontology Service* (WOS) that provides the access functionality focusing on the communication protocol and the underlying architecture.

Figure 4.1 places the server providing *Web Ontology Service* in the web services layer. It access to the ontology repository through the *Ontology Manager API* contained in the *Ontology Manager* described in section 4.4. It provides the access to the external components requiring access to terminological ontologies through two different HTTP based interfaces. Each of the interfaces is focuses on providing access to a different family of clients.

For applications in general environments having to access to terminological models such as thesauri, an interface following the service architecture proposed in the Alexandria Digital Library (ADL) project [101] has been developed. An interface providing the ADL Thesaurus protocol [102] (a protocol designed for the distribution of thesauri through the Web) has been created. The *ADLService* interface at the bottom of figure 4.11 contains the methods provided by this protocol. Given the limitations of ADL Thesaurus protocol to provide access to a single monolingual thesaurus, an extension of the *ADLService* interface called *ExtendedADLService* has been developed to provide access to multiple thesauri, whit those thesauri able to represent properties in multiple languages. As depicted in figure 4.12, the *ExtendedADLService* is implemented by the *MultilingualServiceImpl*, which provides the bridge to the *OntologyManagerAPI* to access to the repository.

For SDI conformant components requiring access to terminological ontologies, a specific

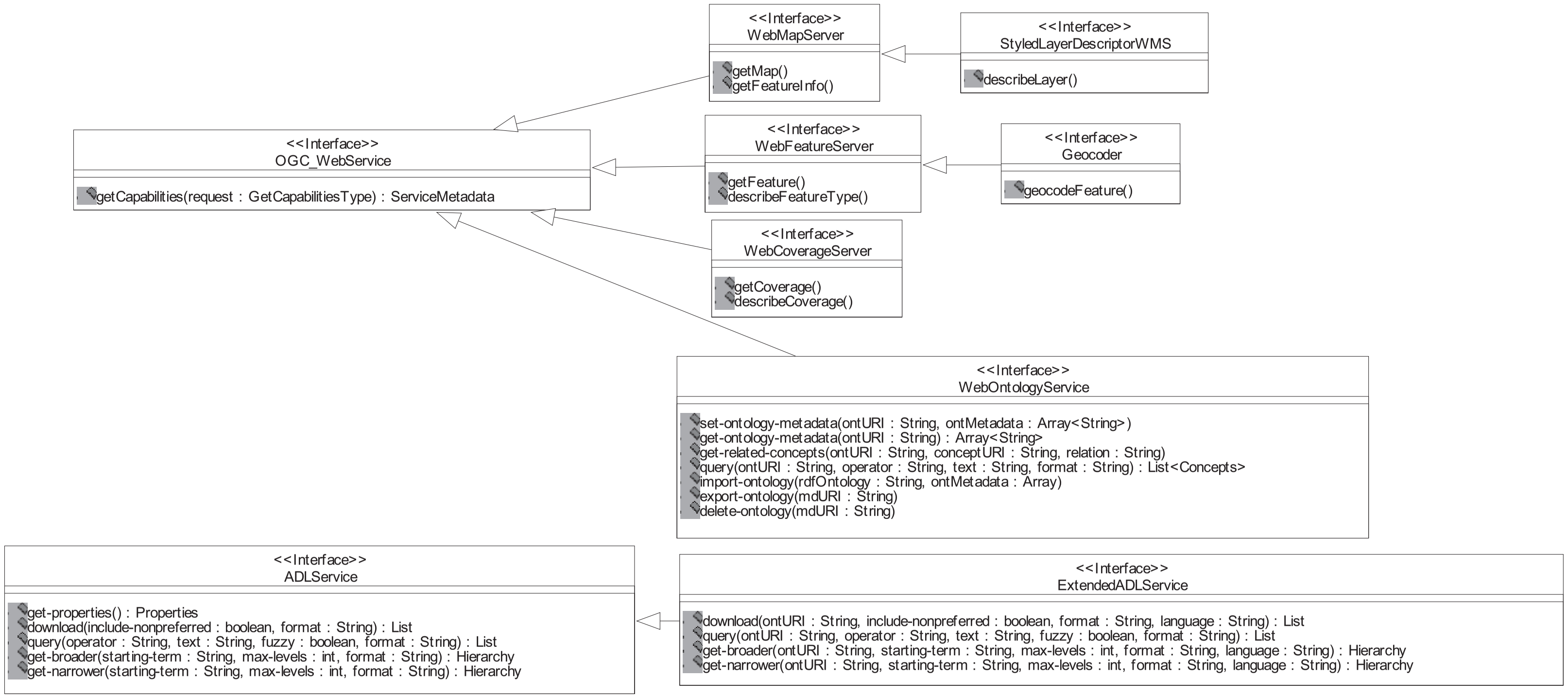

<<Interface>>
StyledLayerDescriptorWMS
describeLayer()
<<Interface>>
WebMapServer
getMap()
getFeatureInfo()
<<Interface>>
Geocoder
geocodeFeature()
<<Interface>>
WebFeatureServer
getFeature()
describeFeatureType()
<<Interface>>
WebCoverageServer
getCoverage()
describeCoverage()
<<Interface>>
OGC_WebService
getCapabilities(request : GetCapabilitiesType) : ServiceMetadata
<<Interface>>
WebOntologyService
set-ontology-metadata(ontURI : String, ontMetadata : Array<String>)
get-ontology-metadata(ontURI : String) : Array<String>
get-related-concepts(ontURI : String, conceptURI : String, relation : String)
query(ontURI : String, operator : String, text : String, format : String) : List<Concepts>
import-ontology(rdfOntology : String, ontMetadata : Array)
export-ontology(mdURI : String)
delete-ontology(mdURI : String)
<<Interface>>
ExtendedADLService
download(ontURI : String, include-nonpreferred : boolean, format : String, language : String) : List
query(ontURI : String, operator : String, text : String, fuzzy : boolean, format : String) : List
get-broader(ontURI : String, starting-term : String, max-levels : int, format : String, language : String) : Hierarchy
get-narrower(ontURI : String, starting-term : String, max-levels : int, format : String, language : String) : Hierarchy
<<Interface>>
ADLService
get-properties() : Properties
download(include-nonpreferred : boolean, format : String) : List
query(operator : String, text : String, fuzzy : boolean, format : String) : List
get-broader(starting-term : String, max-levels : int, format : String) : Hierarchy
get-narrower(starting-term : String, max-levels : int, format : String) : Hierarchy

Figure 4.11: External Interfaces of the WOS

interface has been proposed. Geospatial community aims at facilitating the adoption of open, spatially enabled reference architectures in enterprise environments worldwide. In this context, the Open Geospatial Consortium (OGC) has defined an architecture of components called OGC Web Services Architecture (WSA) [217, 134] that all the components integrating an SDI must fulfill. The objective is to promote interoperability among OGC service specifications by increasing commonality and discouraging non-essential differences. According to this API, every OGC service inherits from a general service whose unique operation is *getCapabilities* [217]. The *getCapabilities* operation provides a description of the service, its operations, parameters and data types. It is used for clients to identify whether a service provides the needed functionality and how to access it.

Although OGC has developed numerous specifications for SDI web services, they have not created a specification for a service to manage ontologies yet. The *WebOntologyService* interface has been developed to fulfill this gap. As it shown in figure 4.12, the interface is implemented by the *WebOntologyServiceImp* that uses the *OntologyManagerAPI* to provide the required functionality.

The top elements of figure 4.11 show the integration of *WebOntologyService* with the rest of OGC services. The *WebOntologyService* extends the standard *OGC_WebService* interface (as the rest of services in the OGC WSA) with methods that provide the functionality to access terminological ontologies. And thanks to the compliance with the general OGC architecture, it can be integrated with the rest of OGC services in an SDI.

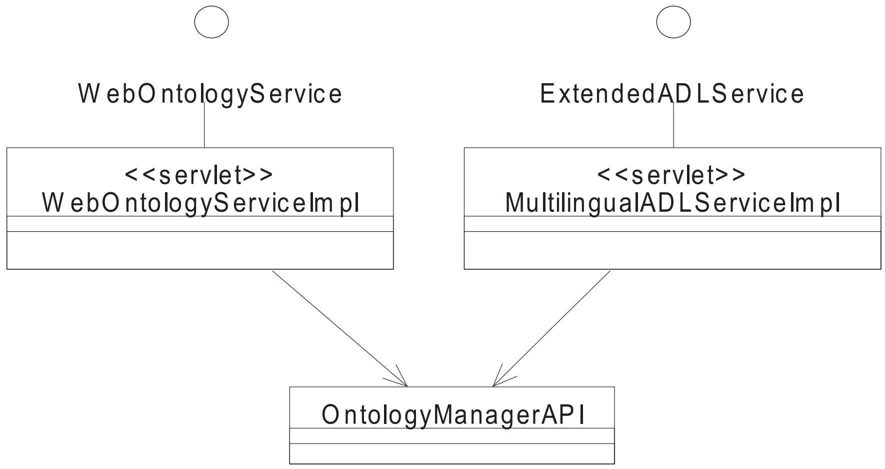


Figure 4.12: Implementation of WOS interfaces

It must be noted that none of the service interfaces include update methods for concepts and relations. With respect to the clients accessing the models, each ontology is considered as a whole, managing their changes as different versions of the whole ontology.

4.7 Performance tests

In the context of this thesis, 70 terminological ontologies among thesauri and other types of controlled vocabulary have been tested. All those models have been managed and distributed using the components and tools described in this chapter.

Some of the managed terminological models have a considerable size; therefore, to test that the developed components are able to work with them without problems, a performance test has been done. For the test, a set of terminological ontologies including thesauri, authority files and classification schemes commonly used by the geographic information community have been selected. Table 4.1 shows the selected models: the ADL Feature Types Thesaurus[27] (ADL FTT), the ISOC thesaurus of Geography[28] (ISOC-G), the ISO-639 codes for the representation of names of languages [84], the UNESCO Thesaurus[29], the OGP Surveying and Positioning Committee Code Lists[30] (EPSG) , the Multilingual Agricultural Thesaurus[31] (AGROVOC), the European Vocabulary Thesaurus (EUROVOC)[32], the European Territorial Units (Spain and France) (ETU), and the GEneral Multilingual Environmental Thesaurus[33] (GEMET). They have been selected because they are very different in sizes and can be used to show how the load time evolves with the terminological ontology size. The size of these models varies greatly from a few hundred of concepts to thousands of concepts and/or relations. All of them, except the ADL Feature Types have been transformed to SKOS using the processes described in chapter 3 and then imported into the ontology repository. The ADL Feature Types described here is an extension of the original one that was manually created using the ThManager tool to include a more detailed classification of features types.

Table 4.1 columns indicate: the name of the ontologies (Name column), their number of concepts (NC column), their total number of properties and relations (NP and NR columns), and the number of languages in which concept properties are provided (NL column). To give an idea of the cost of loading these structures, the sizes of SKOS and binary files (SS and SB columns) are provided in Kilobytes (KB).

Table 4.1 includes the performance in load time of the analyzed terminological models. The times have been measured using a 3 Ghz Pentium IV processor. The measured obtained are used to compare the difference in time of sing the Jena library to load the SKOS files as RDFs, and the time when they are stored using the binary storage model described in section 4.4. This can be used to determine the size of the models that can be managed for each type of storage model. Three different load times (in seconds) have been computed. The BT column

[27] http://www.alexandria.ucsb.edu/gazetteer/FeatureTypes/ver070302/
[28] http://thes.cindoc.csic.es/index_esp.html
[29] http://www.ulcc.ac.uk/unesco/
[30] http://www.epsg.org/
[31] http://www.fao.org/aims/ag_intro.htm
[32] http://europa.eu/eurovoc/
[33] http://www.eionet.europa.eu/gemet

Name	**NC**	**NP**	**NR**	**NL**	**LT**	**BT**	**JT**	**SS**	**SB**
ADL FTT	210	210	408	1	0.4	0.047	0.062	103	41
ISOC-G	5136	5136	1026	1	2.4	1.063	1.797	2796	1332
ISO-639	7599	16247	0	6	5.1	1.969	2.89	3870	3017
UNESCO	8600	13281	21681	3	2.1	1.406	2.984	4034	2135
EPSG	4772	9544	0	1	1.8	0.969	1.796	2935	1682
AGROVOC	16896	103484	30361	3	7.5	4.953	14.75	15859	5089
EUROVOC	6649	196391	20861	15	11.1	9.266	15.828	18442	11483
ETU	44991	89980	89976	2	13.3	10.625	17.844	23828	10412
GEMET	5244	326602	12750	21	13.7	11.828	25.61	28010	15048

Table 4.1: Sizes of some thesauri and other types of vocabularies

contains the load time in seconds of binary files without the cost of creating the GUI for the thesauri viewers (load time for the WOS). The LT column contains the total load time in seconds of binary files (including the time of GUI creation and drawing for load time in the ThManager). The JT column contains the time in seconds spent by Jena library to load the SKOS into memory (it does not include GUI creation). The difference between the BT and LT column shows the time used to draw the GUI once the thesauri have been loaded in memory. The difference between BT and JT columns shows the gain in terms of time of using a binary storage instead of a RDF based one.

Figure 4.13 depicts the comparison of the different load times shown in table 4.1 with respect to the size of the RDF SKOS files. The order of the terminological models in the figure is the same as in the table 4.1. It can be seen that the time to construct the model using a binary format is almost half the time spent to create the model using a RDF file. This is the only time required to load the terminological ontology for the WOS.

Once the binary model is loaded, the time to generate the GUI for the ThManager is not very dependent on thesaurus size. This is possible thanks to the redundant information added to accelerate the load of the alphabetic viewer and to facilitate the access to top concepts for hierarchical terminological models. The time to generate the top terms and the alphabetic list of the concepts in all the available labels is very costly; therefore it has to be previously calculated to obtain a reasonable drawing time of the GUI. This redundant information produces an overhead in the load of the model but is compensated by the reduction the graphical drawing time. The import time is increased (the redundant information is generated then), but that is only done once, while the visualization is done frequently.

Despite using the binary representation model described in section 4.4, load time of the biggest terminological models shown in figure 4.13 is still high. However, once it is loaded, future accesses are immediate; negligible for WOS and less than 0.5 seconds for ThManager. The ThManager time accesses include, opening it again, navigating by thesaurus relations, changing

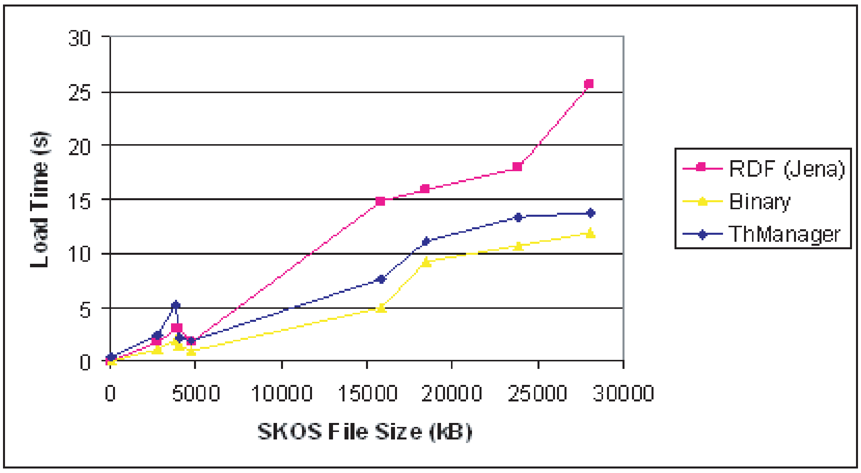


Figure 4.13: ThManager Load Times

the visualization language, and searching concepts by their preferred labels. To minimize the load time, in the ThManager the terminological models can be loaded in background when the application is launched, reducing in that way the user perception of the load time. In the WOS it has to be waited until all the models are loaded, but it is only done at start-up time.

Another interesting aspect shown in figure 4.13 is the peak of the third element. It corresponds with the ISO-639 classification scheme. It has the special characteristic of not having hierarchy and having many notations. These two characteristics produce a little increase in the load time of the model, given that the top concepts list contains all the concepts and the notations are more complex than other relations. However, most of the time is used to generate the GUI of the tree viewer. The tree viewer gets all the concepts that are top terms, and for each one it asks for its preferred labels in the selected language and sorts them alphabetically to show the first level of the tree. This process is fast for a few hundred of concepts but not for the 7599 of the ISO-639.

This problem can be solved using the *type* field in the metadata describing the terminological ontology (see section 2.4.1) to distinguish between terminological models and perform the different optimized tasks for each type of model. If the tool knew that the terminological ontology does not have broader/narrower relations, it could use the structures used to visualize the alphabetic list, which are optimized to show all the concepts of the ontology very fast, instead of trying to load it as a tree.

The use of the persistence approach based on binary files has the advantage of not having to use external persistence systems such as a DBMS, and it provides a very fast access after the load, but it has the drawback (in time and space) of loading all terminological ontologies in memory (up to 512 MBs for the analyzed terminological models). For the context considered in this thesis it has been found to be enough: it can be used to provide a version of the *ThManager* independent of any DBMS (easier to install and use); and with respect to the WOS, the performance is adequate.

If much bigger thesauri where used, the use of some kind of DBMS would be necessary, and some of the elements in the architecture (e.g., alphabetic viewer) would have to be updated to be able to work properly.

4.8 Conclusions

This chapter has focused on the problems of managing terminological ontologies in infrastructures that require the access to a great amount of them. To manage these models appropriately, this chapter analyzes the need of providing terminological ontologies to all the components of the infrastructure in a coordinated way to avoid the use of several copies of the same ontology or very similar ones in different components. As part of the access problems, it has been found important to tackle the storage problem, the status management, and the relations with other models.

In order to provide a single access to the terminological ontologies an architecture for an *ontology manager* has been proposed. It facilitates the access to different types of storage repositories for the ontologies, allowing the selection of the most appropriate for each situation. The stored terminologies are described through metadata to facilitate their thematic location and access through search services. Additionally, it provides a matching mechanism to relate the different stored terminological models with respect to a common upper lever ontology (in this case WordNet), which acts as a bridge to be able to jump from concepts of a model to other equivalent ones of other models stored in the repository (the matching mechanism is described in section 3.3). The *ontology manager* acts as an independent layer and on top of it different management applications can be built.

To fill the repository and manage the stored terminological ontologies, a editor called *ThManager* has been designed. It provides the functionality to create, describe, update and delete terminological ontologies stored in a repository (accessed through the *ontology manager*) and it is distributed as open source software. Additionally, *ThManager* architecture facilitates the design and implementation of different edition and selection components to provide access to terminological models.

In order to provide the access to the ontologies contained in the repository to the components requiring it a centralized web service called *Web Ontology Service* (WOS) has been designed. In the same way as the *ThManager*, the service access to the ontologies using the *ontology manager library*. Designed as a centralized service, its architecture aims at reducing the need of creation of new terminological ontologies, using existent ones, improving reusability and avoiding duplicities and inconsistencies. To facilitate access two different interfaces are provided, one for general access and the other one compliant with the OGC Web Services Architecture specification for its integration in an SDI.

Nowadays, tools and systems managing terminological model are not very effective in loading terminological models with hundreds of thousands concepts and relations without the use of a DBMS, so the storage model developed for situation when the use of a DBMS is discouraged has been tested calculating the load times of a set of the terminological models used in the context of this thesis. The performance of the tool is proved through a series of experiments on the management of a selected set of thesauri. This work analyzes the features of this selected set of thesauri and compares the efficiency of this tool with respect to their load directly from a RDF file. The results obtained show that the designed storage is good enough for the required terminological ontologies, providing an acceptable load time and an immediate access. The fast access to typical operations such as browsing, sorting or changing the visualization language increases the usability of *WOS* and *ThManager*.

As future work, it is planned to extend the *WOS* functionality to give support for non-terminological ontologies expressed in formal ontology languages such as OWL. This would not imply a complete redesign of the described architecture because both SKOS is RDF/OWL based, but it requires a further analysis of the implications derived from the inclusion of non terminological models and how this would affect the present API.

Future work could be oriented as well towards the enhancement of *ThManager* functionalities. On the one hand, the ergonomics to show the connections among different thesauri could be improved. Currently, these connections can be computed and annotated but the created GUI does not allow the user to navigate them. The base technology has been already developed, only a graphical interface is needed. On the other hand, *ThManager* could be extended to support data types different from texts (e.g. images, documents, or other multimedia sources) for the encoding of property values of concepts. This and the possibility to add user defined relations would increase the flexibility to support more complex models. Finally, improvements in usability could be added thanks to the component-based design of *ThManager* widgets. New viewers or editors can be created with little effort to meet the needs of specific users.

Chapter 5

Applicability of terminological ontologies to SDI discovery

5.1 Introduction

The previous chapters have focused on describing different strategies, processes, components and systems to improve the management of terminological ontologies. This chapter takes all those elements and applies them to improve information discovery in spatial data infrastructures.

Section 5.2 puts into context the information retrieval needs of spatial data infrastructures. It reviews the terminology, applicability and challenges involved in the improvement of SDI discovery systems. After this revision, it analyzes the roles that terminological ontologies can play in these discovery scenarios to improve their performance. The rest of the sections use the frameworks and architectural patterns described in previous chapters to improve resource classification, information retrieval systems and information browsing.

In order to help in the classification of resources, section 5.3 proposes the integration of the terminological ontology management component described in section 4.5 with a tool for the creation of geographical metadata. The objective of the integration is simplifying the creation of semantic annotations for the geographical resources.

Another basic element in the discovery process is the development of search components. In order to improve the recall in SDI search systems, section 5.4 shows a retrieval model that uses terminological ontologies for query expansion. These terminological ontologies are accessed through a web service following the architecture described in section 4.6.

In addition to query interfaces, browsing systems are also usually used to locate information in many different contexts. As mentioned in [11, p. 20], the user task in a discovery scenario might be one of browsing instead of performing blind searches. These systems are based on providing an ontology more or less complex that is used as a core structure for the browsing process. However, the direct use of ontologies is not appropriate because usually neither a unique ontology nor their complete structure have been used in the collection. Somewhat related to the information browsing problem is the need to identify the content of different geographic data catalogs that contribute from the different nodes of the SDI. Having an overview of the content of a geographic catalog helps to identify if the content may be suitable for the required purpose. This is very important in an SDI where usually there is not a unique geographic data catalog but hundred of them, and where an indiscriminate search can consume a lot of processing time only to obtain that many of them contain nothing about the desired subject. This problem can be attended by using suitable content summary descriptions for the collection contents that help to reduce the scope of the searches; however, given that manual generation is costly because the information gets quickly, an automatic approach is more effective. Section 5.5 describes two approaches to provide adapted categorizations of a collection. On the one hand, it proposes a method to generate a topic map adapted to the collection content that can be used for browsing. On the other hand, it describes a set of methods to generate a collection classification that can improve the efficiency of distributed discovery systems.

5.2 Information retrieval in the context of Spatial Data Infrastructures

The SDI concept was firstly used by the USA government in the executive order 12906 titled “Coordinating Geographic Data Acquisition and Access: the National Spatial Data Infrastructure” [206] with the objective of grouping the technology, policies, standards and human resources needed to acquire, process, store, distribute and improve the utilization of geospatial resources under the name of Spatial Data Infrastructure. The term has become more popular and has been redefined since their origins. Coleman and Nebert [32] remarks that it should include data providers (source of spatial data); databases with data descriptions; networks; technologies to manage, search and represent large collections of information; institutional agreements; policies; standards; and final users. The Global Spatial Data Infrastructure Association Cookbook [152] defines an SDI as “the relevant base collection of technologies, policies and institutional arrangements that facilitate the availability of and access to spatial data, providing a basis for spatial data discovery, evaluation, and application for users and providers within all levels of government, the commercial sector, the non-profit sector, academia and by citizens in general”. As Nebert [152] remarks, the Spatial Data Infrastructures (SDI) provide the framework for the optimization of the creation, maintenance and distribution of geographic information at different organization levels (e.g., regional, national, or global level) and involving both public and private institutions. Nowadays, the European Committee for Standardization (CEN) has redefined the SDI concept as a platform-neutral and implementation-neutral technological infrastructure for geospatial data and services, based upon non-proprietary standards and specifications [173]. Using this last definition, an SDI can be seen as a specific kind of information infrastructure specialized in geographical data.

The increasing relevance of geographic information for decision-making and resource management in different areas of government has promoted the creation of geo-libraries and spatial data infrastructures to facilitate distribution and access of geographic information [163]. Step by step, SDIs have gained relevance and nowadays many governments are considering seriously SDIs as basic infrastructures for the development of a country (as relevant as electricity, water, gas, transport or telecommunication infrastructures), and they are developing policies to provide to the general public as much geographical information as possible[159]. Among the different initiatives to create SDIs, the National Geospatial Data Clearinghouse project[1] can be considered as one of the most relevant driving forces that encouraged this new concept of infrastructure. This project was developed by the Federal Geographic Data Committee (FGDC) as a key component of the U.S. National Spatial Data Infrastructure. It defines a distributed network of catalogs that enable enterprise and technological independence by using a standardized mechanism for catalog querying. The nodes of this network conform to the ANSI/NISO Z39.50 information and retrieval protocol, which has been widely used since the beginning of the 1990s for the construction of OPACs (Online Public Access Catalogs). And although the Clearinghouse project had originally a national character, many servers from other countries (e.g. Canada, Australia, South Africa or Uruguay) have adhered to the initiative. Other relevant initiatives are *"Inspire"*[2], promoted for the European commission for the development of an SDI in Europe; and *"GeoConnections"*[3], launched by the Canada government to develop the Canada SDI.

According to the INSPIRE initiative, an SDI consists of four main different groups of components depicted in figure 5.1 (extracted from [187]). These components are: content repositories, data-service catalogs, access and geo-processing services, and user applications [187, 173].

The content repositories (bottom part of figure 5.1) store the geo-spatial data and provide them to the rest of the infrastructure components. In addition to this, depending on the system, they can contain other data such as photographs or multimedia content as long as they contain some spatial referencing information.

With respect to data catalogs (left area of figure 5.1), they contain descriptions (metadata) of the elements stored in the content repositories. Very similar are the service catalogs that contain descriptions of the set of services offered by an SDI. The differences between these two types of catalogues are the types of data described and the metadata model used to describe them. The structure of the used metadata is not arbitrary, to provide interoperability, they follow one of the several existing international standards used to describe geographical information or

[1] http://clearinghouse1.fgdc.gov/

[2] http://www.ec-gis.org/inspire/

[3] http://www.geoconnections.org/CGDI.cfm

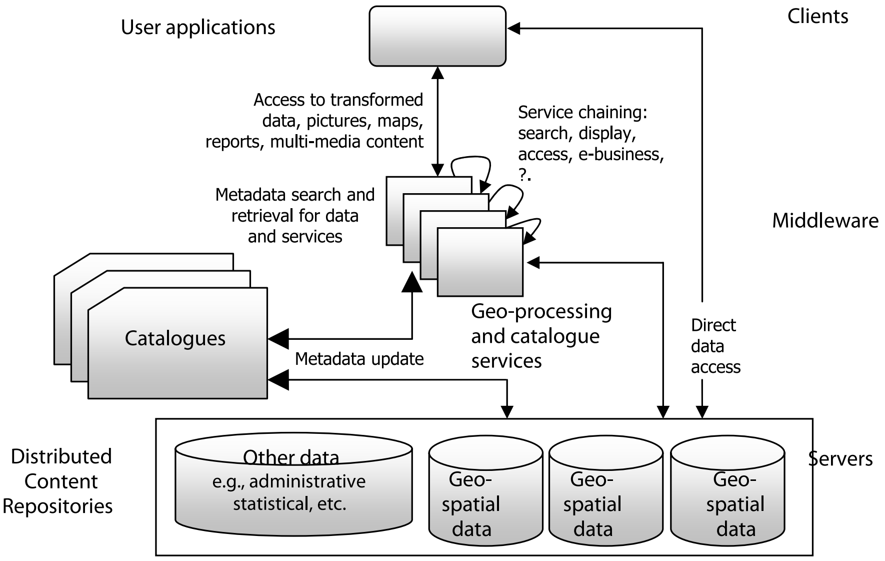

User applications
Clients
Access to transformed data, pictures, maps, reports, multi-media content
Service chaining: search, display, access, e-business, ?.
Metadata search and retrieval for data and services
Middleware
Catalogues
Metadata update
Geo-processing and catalogue services
Direct data access
Distributed Content Repositories
Other data
e.g., administrative statistical, etc.
Geo-spatial data
Geo-spatial data
Geo-spatial data
Servers

Figure 5.1: General architecture of an SDI

services. For metadata, the ISO-19115, the CSDGM-FGDC, and the Dublin Core Metadata Element Set [88] are the most used. For services, the *"ISO 19119 for Geographic Services"* [94] is the more accepted.

The data and the data descriptions stored in the repositories and catalogues of the SDI are used for a wide range of services integrated in the infrastructure (center of figure 5.1). Some of the services are focused on providing management / administration / coordination of the stored information; others on providing access to the resources, facilitating interfaces to search data / services, and to retrieve maps / coverages / features / geographical names; and others in processing the stored information by transforming coordinates or merging different sets of geo-spatial data. In this area, the Open Geospatial Consortium[4] has focused in the development of standard interfaces for these services. Some examples are the document of the common elements used along a SDI [217], or the documents describing the interfaces of some SDI components such as the services providing access to data/service catalogs [153], features [214], coverages [218], maps [38] or gazetteers [57].

Finally, there are user applications (top of figure 5.1) using these SDI services. They are desktop or web tools that combine the information obtained from the different SDI services to provide the functionality required by the final user. In general, what they do is to provide a human oriented interface to the services in the infrastructure, allowing to a human user manage, search, locate, visualize and analyze, the geo-spatial information stored in the SDI.

[4] http://www.opengeospatial.org/

One of the main barriers in a distributed systems such as the SDI described by this architecture is the heterogeneity in their construction. SDI-based initiatives must deal with the challenge of overcoming the syntactic and semantic heterogeneities that may arise in the systems (system services and data accessed through these services) participating in these distributed scenarios.

The use of standards and recommendations proposed by different standardization organizations (ISO-TC211[5], CEN-TC287[6]) and other community consortiums (Open Geospatial Consortium, W3C . . .) has supposed a very significant step for the foreseen interoperability, at least to solve the most basic problems of syntactic interoperability in the definition of common interchange formats and service interfaces. However, as the implementation of standards and specifications is still open for the interpretation of developers, important semantic differences still remain. Each component defines its own models, properties and possible values, independently of the ones used in the rest of the SDI components increasing in that way the interoperability and communication problems. Moreover, the geospatial community, as other communities, expects to make profit of the resources developed in other domains not necessarily using same specifications and standards.

In order to reduce this semantic heterogeneity within the Spatial Data Infrastructures (SDIs), the use of ontologies as knowledge representation mechanism is acquiring an increasing relevance. Ontologies have been traditionally used in other knowledge areas such as digital libraries to improve knowledge organization and information retrieval [11, 16]. Some specific examples of ontology based information retrieval applications are the Alexandria digital library[7] that shows the applications of ontologies in a collection of geo-referenced materials; and OntoSeek [75] and Ontalk [112], which use ontologies to provide conceptual searches. Focusing on the SDI context, the use of ontologies can facilitate the interoperability in the different scenarios involved in the resource access paradigm shown in figure 5.2[8].

As concerns **resource discovery**, some of the most remarkable problems that affect the interoperability and cooperation of discovery systems are metadata schema heterogeneity and content heterogeneity [163].

As regards the problem of metadata schema heterogeneity [161], given that a metadata schema is a model that contains a set of concepts with properties and relations to other concepts, their structure can be modeled as an ontology, where metadata records are instances of this ontology [17]. This kind of ontologies may be used to profile the metadata needs of a specific geospatial resource and its relationships with metadata of other related geospatial resources,

[5] International Organization for Standardization (ISO), technical committee for Geographic information/Geomatics

[6] European Committee for Standardization (CEN), technical committee for Geographic Information

[7] http://www.alexandria.ucsb.edu/

[8] To illustrate the Geospatial Resource Access paradigm, the figure shows the process initiated by a user (e.g., citizen, or local administration) to discover, evaluate and finally have access to a "National Topographic Map" (distributed by a National Mapping Agency).

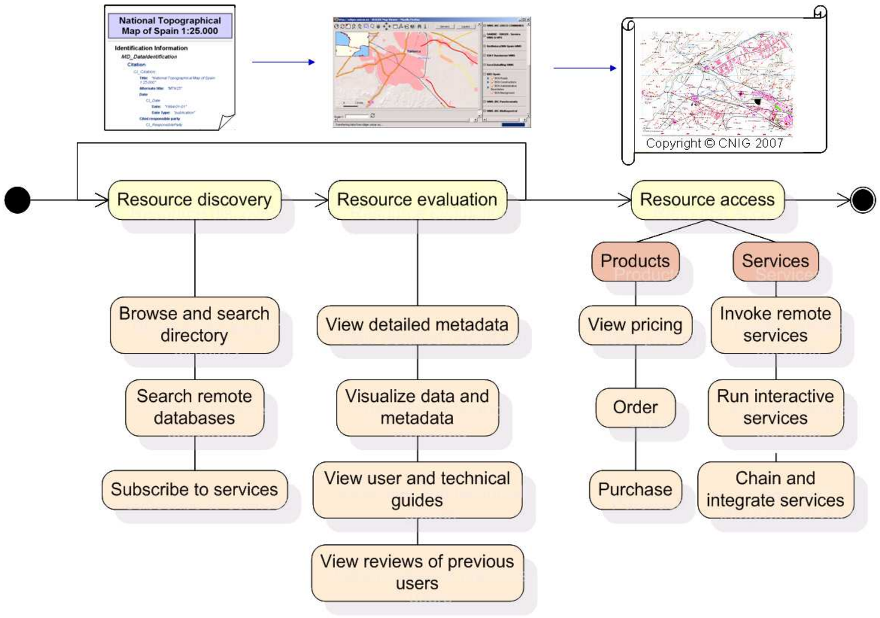


Figure 5.2: Geospatial Resource Access Paradigm, modified after Nebert [152].

or to provide interoperability across metadata schemas. Transformations of metadata between two different standards could be solved by systems that observe commonalities of two ontologies and automatically detect the metadata element mappings. An example of this kind of mappings can be seen in Weißenberg and Gartmann [216], where different metadata standards are used to describe geo-services. Another approach is the one shown in Bowers and Ludäscher [23] presenting a system for transforming heterogeneous data through a formalized ontology. There, metadata standards are modeled as ontologies using F(rame)-Logic and matched to enable semantic queries.

For the problem of metadata heterogeneity, ontologies facilitate classification of resources and information retrieval [158]. Metadata try to exactly describe information resources to enhance information retrieval, but this improvement depends greatly on the quality of metadata content. One way to enforce the quality is the use of selected terminology for some metadata fields in the form of lexical ontologies. These ontologies are used to describe contents but also allow computer systems to reason about them.

Bechhofer and Goble [14] show that thesauri are useful in bridging the gap between the metadata describing the resources and the concepts used by the searcher, remarking that thesauri should have an underlying formal ontology to reason about concepts. Another classification project (but not GI related) is Healthcybermap [21], which combines metadata and ontologies to provide new ways of finding health information resources. There, explicit concepts in the resource metadata are mapped onto an ontology (e.g. a clinical terminology or classification or a collection of merged ontologies) allowing a search engine (Semantic Web agent) to infer implicit meanings not directly mentioned in either the resource or its metadata.

Regarding **resource evaluation**, an SDI must facilitate the task of viewing detailed metadata, and must provide enough means to visualize the data appropriately. In this scenario, one could consider multilinguality and resolution level as main problems for system interoperability.

In the case of viewing metadata in a specific language required by the user, one may face the problem of having to translate it. Once again, metadata ontologies and terminological ontologies may facilitate the work in two important aspects. Firstly, a metadata ontology may provide the labels, in the appropriate language, for the elements of the metadata schema. Secondly, terminological ontologies may be used in the task of automatic translation of metadata to increase accuracy of translations.

Regarding the case of portrayal services for data visualization, one must face as well the problem of resolution level and "culture and linguistic adaptability". On the one hand, the resolution level affects portrayal of data because not all the features are meaningful at a particular zoom level. For instance, at a city scale level, it is worth visualizing the features of the urban transport network (streets, avenues, squares . . .). However, these urban network features are not meaningful for a road network at national level. On the other hand, culture and linguistic adaptability may influence the results offered by portrayal services. Although the visualization of data seems language independent, SDI developers must consider the internationalization of legends and the display of internationalized attribute information if necessary. For instance, the BALANCE project [171] uses external XML files to provide the translations of Web Mapping Services (WMS) capability documents, which are used by the client for the translation of WMS data layers names. Moreover, during the phase of resource evaluation, other multilingual and multinational issues must be taken into account; for instance, the selection of the correct Spatial Reference System, or the appropriate symbology according to cultural traditions of each country. Thus, one could seriously consider the creation of an ontology of features visualized through portrayal services defining for each feature: the range of scales most appropriate for visualization, its textual label in every language, the most appropriate reference system for a geographic area, or the appropriate symbol (image) for rendering this feature on a map.

Finally, the **resource access** and further processing may benefit as well from the use of ontologies to facilitate data sharing and system development [176, 211]. Once again, ontologies help to define the meaning of features contained in geo-spatial data and they can provide a "common basis" for semantic mapping; for example, to find similarity between two features that represent the same object but that have been defined using different languages, such as the standardization items defined by ISO/TC211 (technical committee for Geographic Information/Geomatics) to create data dictionaries defining features and attributes that may be of interest to the wider international community (ISO-19109 [93], ISO-19110 [91], ISO-19126 [90]).

In this context, Lutz and Klien [139] describe a system to interrelate features provided by different GI services to give a unified view to the final user. Also Shafiq et al. [185], provides communication between web services using an ontology based infrastructure. Other works like Fonseca et al. [59] and Fonseca [58] even propose the creation of software components from diverse ontologies as a way to share knowledge and data. These software components are implemented as classes derived from ontologies, using an object-oriented mapping. This use of an ontology, translated into an active information system component, leads to ontology-driven information systems, in this case ontology-driven geographic information systems. Furthermore, it is also usual in GI context to hear about extending the metaphor of Spatial Reference Systems (i.e., referencing things to some point on the ground) with the definition of Semantic Reference Systems [117]. The idea is that apart from spatial reference systems commonly used in maps and Geographic Information Systems (GIS), non-spatial components of geographic information should conform to some kind of semantic referencing.

From the different ontology models, terminological ontologies such as controlled vocabularies, authority lists, taxonomies of thesauri are frequently used in the digital library context for these described tasks. They are so used because they are easier to create than more formal models and, in the context they are used, the semantic they provide is usually enough. Nowadays, there is a great variety of terminological models with a high quality and extent in many diverse areas, including the field of geographical information. This fact, makes the use of terminological models in the SDI context very appropriate to improve classification and retrieval of resources. The need to create new models is reduced because the existent ones can be adapted to fit to the specific requirements of each system.

From these three different areas of work (discovery, evaluation and access) inside of the resource access paradigm of an SDI where the ontologies can be used to improve their performance, this chapter focuses on the discovery area. Discovery process is the most relevant of the three since is the first one in the resource access paradigm. If the required resources are not found by the user, it is not possible to evaluate or access to them.

The discovery process in an SDI combines the knowledge provided by two different research areas. On the one hand, to locate the required geo-spatial information it is required to use information retrieval techniques adapted to the geographical context. On the other hand, the continuous nature of the geographical information creates the need of managing information generated by different countries, and therefore described using different languages.

The field of geographical information retrieval (GIR) can be considered as a specialization of the general information retrieval knowledge area that uses the specific characteristics of the geographical information to improve the general solutions. Due to the fact that a high percentage of the information handled by institutions and public or private companies have, to some extent, relation to spatial data, the field of Geographic Information Retrieval has gained a lot of interest. The creation of specific sessions in reputable international congresses such as the

ACM SIGIR (Special Interest Group on Information Retrieval) or the CIKM (Conference on Information and Knowledge Management) devoted exclusively to research and development of GIR shows its increasing importance. Figure 5.3 shows the geographical information retrieval framework used in this chapter. It is based on the one proposed by Baeza-Yates and Ribeiro-Neto [11] but adapted to geographical information. Three different areas of this framework have been tackled: first, the improvement of the resource classification system, i.e. the creation of the metadata records describing the resources (right-bottom of figure 5.3); second, the expansion of classical query systems with additional terms (left of figure 5.3); and finally, the development of information browsing strategies (upper area of figure 5.3);

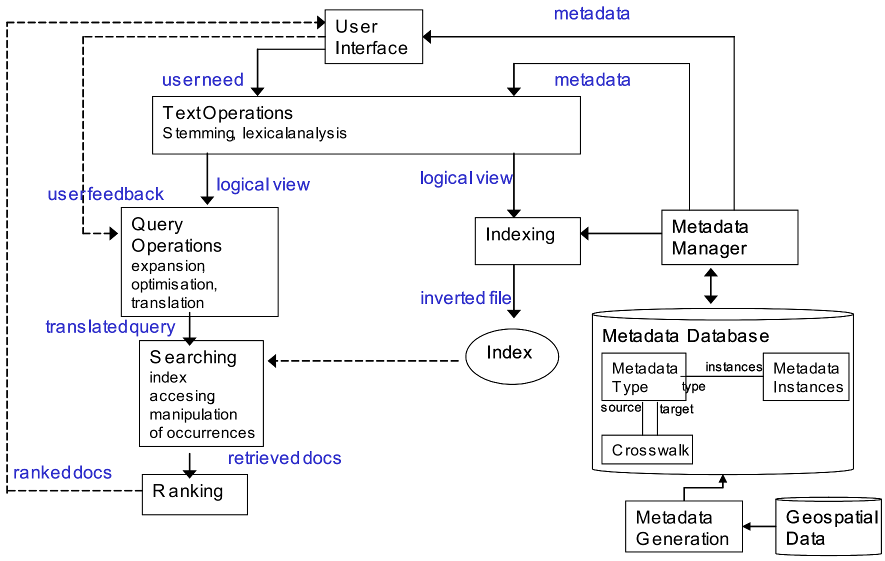


Figure 5.3: Geospatial Framework of geographical information retrieval. Modified version of the proposed by Baeza-Yates and Ribeiro-Neto [11]

In addition to the management of the special geographical features contained in this kind of resources, it is important to remark the multilingual aspects usually associated to geospatial resources. In contexts, such as the establishment of a SDI (encouraged by the INSPIRE initiative), the support for an increasing number of official languages represent an additional barrier to interoperability. Multilingual aspects, often discussed under the concept of cultural and linguistic adaptability (CLA). Geography do not understand of frontiers or administrative limits and therefore most of geographical repositories feed from data from areas corresponding to different countries (e.g., the data about the Tajo hydrographical basin comprises areas from Spain and Portugal).

The consideration of multi-lingual/cultural and linguistic adaptability aspects depends on the scope and audience/users of (new) geo-systems and geo-data; the processes (e.g. information, data transfer, online database access); and the age of the data, database or system. Although, not all data need be modeled supporting multi-lingual aspects, it is difficult to know if a system will be needed one day under multi-lingual conditions. In the case of SDI discovery multilingual aspects are quite relevant because the metadata used to describe the geographical resources are basic to locate the resources and they are completely language dependent.

During the last decade, multilingual information retrieval has attracted a lot of attention as an area of multidisciplinary research that combines aspects of Natural Language Processing, Information Retrieval and Digital Libraries. The origins of multilingual information retrieval can be found in 1996 with the organization of the first workshop specifically for the systematic comparison of multilingual retrieval systems within the Special Interest Group on Information Retrieval (SIGIR) ACM. Since then, numerous activities have been organized regularly at the international level: the Text Retrieval Conference (TREC) established in 1997 a new theme (track) specialized in multilingual information retrieval; the group NII NACSIS-Text Collection for IR workshop (NTCIR), established in 1998, also includes as special thematic (track) comparing multilingual systems working with English and Asian languages; and the Cross-Language Evaluation Forum (CLEF) created in 2000 for the evaluation of multilingual information retrieval systems in Europe.

In the geographical area, multilingual aspects are a concern of many geographical related organizations and committees. Some examples in the arena of standardization are JTC1-SC32/WG1 "Open-EDI" and SC32/WG2 "Metadata". ISO/TC 211 has also dedicated resources to CLA and some of its GI related standards analyze the multilingual problem. For instance, ISO-19135 [92] "Geographic information – Procedures for item registration", ISO-19126 [90] "Geographic information - Feature concept dictionaries and registers", ISO-19112 [86] "Geographic information - Spatial referencing by geographic identifiers". In this field, specially relevant are the recommendations for multilingual support of the technical specification ISO-19139 [95] for the encoding of multilingual geographical metadata in XML (see clause "7.3 Cultural and Linguistic adaptability extensions").

Specifically devoted to management of multilingual issues in geographical information retrieval systems, it is the GeoCLEF discussion forum (Evaluation of cross-language Geographic Information Retrieval Systems). It aims to establish an infrastructure that enables the assessment of spatial and multilingual characteristics of information retrieval systems and proposes tasks such as translating locations, detection of ambiguous geo-references (e.g. "Jack London" is a person not a place, "Islas Baleares" and "I. Baleares" are the same place), detection of spatial ambiguity (e.g., "Zaragoza" in Spain o in Mexico), and combination of textual and spatial information.

The role of ontologies for information retrieval is even more significant in the cases where a strategy for cross-language information retrieval is required. For example in an European

SDI it is not expected that the member states provide translation for each metadata record they produce. Therefore, a European SDI catalog must tackle the problem of finding resources independently of the language used for metadata and data creation (i.e., the multilingual issues have to be tackled).

Usually, there are three different approaches to manage the general cross-language information retrieval problem: the translation of queries, the translation of documents (metadata in GI context), and the conceptual indexing of documents and queries in a language independent manner Oard [168]. In any of these cases, terminological ontology resources play a significant role for implementing these strategies [63, 162]. In the context of this thesis, the metadata describing the geographical resources that have been available were usually in a few set of languages such as Spanish, English, and French. Therefore, the second approach (translation of documents) is directly provided (between the existent set of languages).

Additionally, for those situations where no translation of the metadata exist, the approach followed is the translation of the user queries. The translation is performed by using the multilingual terminological ontologies available in the system. Fortunately, there is a large number of multilingual models that can be used for this purpose. For example, the main thesauri used along this thesis have the following multilingual characteristics: GEneral Multilingual Environmental Thesaurus[9] (GEMET) provides terms in 27 languages, the Agriculture vocabulary [10] (AGROVOC) is in 19 languages, and the European vocabulary[11] (EUROVOC) is provided in 23 languages.

5.3 Resource classification through semantic annotation

SDIs are characterized by integrating information from many different sources, which may range from individuals (e.g., concerned citizens or graduate students in geography) and nonprofit institutions (e.g., universities or non- governmental organizations for humanitarian help) to large remote sensing companies or governmental institutions (e.g., national mapping agencies, cadastres or environmental agencies). This great variety of sources implies a consequent heterogeneity in the classification process, both in the wide choice of metadata models used and in the different expertise of metadata creators.

According to the different resources and organizational procedures of the institutions contributing to the SDI, metadata may be created by scientific spatial data producers, by library cataloguers, or by administrative staff. Therefore, it is important to provide users with metadata edition tools that facilitate the content creation, i.e. generating those metadata elements that can be automated, and guiding in the edition of descriptive elements that must be typed

[9] http://www.eionet.europa.eu/gemet

[10] http://www.fao.org/agrovoc

[11] http://europa.eu/eurovoc/

manually. Moreover, since typing errors in metadata creation can imply not finding a resource, control of content quality is very important. Being homogeneous in the selection of the terms used to describe a resource is another important issue. If two resources have similar characteristics, they should be described with the same terms. Otherwise, a query system will only return a subset of the records it should return.

The use of different metadata standards can be solved by relating the metadata models and establishing crosswalks between them. Previous to this thesis work, a lot of effort has been done in this area; works such as Lacasta et al. [124], Zarazaga-Soria et al. [222] and Nogueras-Iso et al. [161] show the problems that affect to the relation of metadata models and propose approaches for the creation of crosswalks between them, that facilitate the transformation between models. However, even if all the metadata describing the resources have been transformed to a single model, the differences in their content still persist.

Tolosana-Calasanz et al. [197] describes a process to identify those metadata elements that affect to a greater extent to the overall quality of the metadata. The use of controlled vocabulary for the most relevant of these fields can help to reduce the time of creation, the number and impact of human errors, and increase the homogeneity. In order to reuse these vocabularies in different SDI services, it becomes essential to manage them uniformly by means of services such as the proposed by the Web Ontology Service in section 4.6 or the ThManager tool shown in section 4.5.

The OGC catalogue service specification [153] (standardizing the interface of discovery systems in SDIs) recommends the use of ISO-19119 [94] for service description and ISO-19115 [87] or Dublin Core [88] for geographic information description. All these standards define a large number of metadata elements, and many of them must or may contain terms from terminological ontologies such as controlled vocabularies or thesauri. Some examples in ISO-19115 are the *descriptive keywords*, the *topic category*, the *distribution format* or the *spatial representation type*.

The values for these elements are usually facilitated by the used metadata edition tool; however, most of these tools manage them independently in a non coordinated way, making difficult to update them and add new ones. To facilitate the use of the terminologies in the metadata creation process improving its quality and homogeneity, and with the objective of managing them uniformly, it is needed to provide to the metadata edition tools with the capability to access to a repository of terminological models in the same ways as it has been recommended along this thesis for all the SDI components requiring to use terminological models.

The solution adopted has been to integrate the management components provided by ThManager (see section 4.5) into a metadata edition tool for the documentation of geographic information resources (metadata compliant with ISO19115 geographic information metadata standard) called *CatMDEdit*. *CatMDEdit* [223] is an Open Source cross-platform desktop tool (distributed as Open Source software[12]), developed in Java and internationalized to six different languages, whose main aim is to promote metadata creation as a mechanism that facilitates the processing of data in a more effective way. Data and metadata management cannot be understood as separate tasks. Thus, *CatMDEdit* facilitates semi-automatic mechanisms for the generation of metadata in order to minimize the interaction with users and hide the complexity of metadata standards. Additionally, it provides alternative mechanisms for integrating Geographic Information System (GIS) tools that enable the access to datasets through their associated metadata. In the same way as other content management tools do, metadata are the means that allow users to describe a resource and launch associated tools for its visualization or modification.

Figure 5.4 shows the architecture of CatMDEdit. It has been designed using a three-layered architecture. The upper-layer contains the three main functional components of the application: a *Resource Browser* (to manage the metadata records), a *Metadata Editor* (to modify the metadata), and a *Resource Viewer* (to visualize the resources associated to the metadata). The second layer contains the middleware software libraries that support the development of the functional components. And the data layer includes the different repositories needed for the configuration of the tool and the storage of data and metadata, which are managed or produced by the application. In this architecture, the thesaurus tool component uses the ThManager visualization components to access to a local repository of terminological ontologies. The *ThesaurusBeans* provided in ThManager library have been used to facilitate the selection of keywords for some metadata elements. Moreover, it must be noted that thanks to this integration *CatMDEdit* not only facilitates the selection of terms in different languages, but also gives access to their definitions, synonyms, narrower-broader concepts and related concepts to help the user to select the most suitable one.

Some more complex ontology models such as those used to describe *spatial reference systems* (datum, ellipsoid and projection) or *citations* (name, organization, address, ...) can also be managed in the same way, but given that the SKOS model is not adequate for these structures and the storage model is based on it, special interfaces to create and to visualize them would be required for each one of them.

If stronger coordination of terminological models were required, the use of a local repository for the ontologies would not be adequate. For these situations, a remote access to a central repository could be provided by the Web Ontology Service - WOS (described in section 4.6), where the ThManager components integrated in CatMDEdit would act as WOS clients.

[12] http://catmdedit.sourceforge.net/

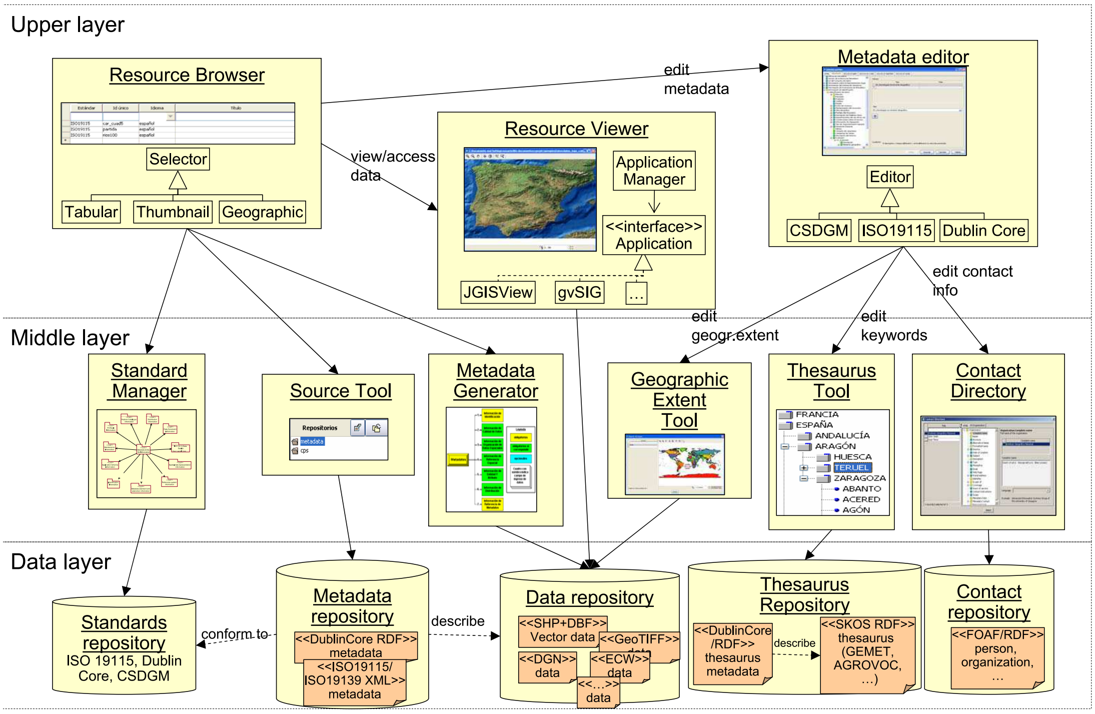


Figure 5.4: Integration of Ontology Management with CatMDEdit

5.4 Information retrieval system enhanced with query expansion

As stated in Nebert [152], SDIs aim at being a basic infrastructure for all kind of users and providers of spatial data within all levels of government, the commercial sector, the non-profit sector, academia and citizens in general. However, in many situations SDI users (and applications built on top of these SDIs) do not have a clear understanding about which keywords they should introduce in their queries. Sometimes the users are professionals with a high level of expertise, but other times it is also usual to find citizens and novice users just exploring for the first time the possibilities offered by SDI services. Thus, the keywords used to express the concepts behind the user queries may differ from the keywords used by metadata creators.

This problem is partially solved by offering search interfaces that guide the user through a thesaurus or other type of linguistic/terminological ontology that contain the more appropriate terms, ideally the same terms also used by metadata creators (see section 5.2). However, the ideal situation of having created metadata by selecting terms from a unique terminological ontology does not occur very frequently. Quite the opposite, an SDI project that implies the cooperation of different institutions usually derives in a collection of metadata records using a wide range of thesauri and other classification schemes. Content creators from different organizations and application domains apply their own criteria for the classification of resources, generating very diverse terminology even for the description of similar resources. This situation is even more problematic when the catalog system stores metadata records written in different languages. In that case, the terminological differences between users and metadata creators become a really difficult barrier for information retrieval.

The heterogeneity of metadata content and the great variety of SDI users' expertise has been shown as a factor that may reduce the retrieval performance, despite guiding the user in the construction of queries by means of a terminological ontology. In this context, the initial query formulation can be improved with techniques such as query expansion [11, Chap.5] to obtain better quality results. The user query terms are used as an initial core set that is enriched with additional elements derived from the original ones that are expected to improve the query specification and rise in that way the recall.

An information retrieval model can be defined as the specification for the documents (in our case, metadata records), queries and the comparison algorithm to retrieve the relevant documents. The integration of query expansion techniques into a retrieval model helps to understand the sense of user vocabularies and to link this meaning to the underlying concepts expressed in metadata records. The objective behind its incorporation is to enhance its capabilities, moving from data retrieval strategies to information retrieval ones.

This section describes an information retrieval model for SDIs that uses query expansion techniques to improve the results. The Web Ontology Service (WOS) described in section 4.6 is integrated to provide access to a repository of related terminological ontologies. The information provided by the terminological models (e.g., translations, synonyms and related concepts in similar terminological ontologies) is used to create an automatic approach for the expansion of user queries. Subsection 5.4.1 reviews the query expansion issues analyzed along all the section. Next, subsection 5.4.2 describes the information retrieval model. Finally, section 5.4.3 shows the results of applying the proposed method for the retrieval of a metadata collection.

5.4.1 State of the art in query expansion

A classical way to expand queries is to use the semantic relations of the natural language. Relations such as *synonymy*, *hypernymy-hyponymy* or even *holonymy-meronymy* and *gender* relations can be used to improve the query. However, adding new elements comes to a cost, if *polysemic* or *homonymic* terms (they have several meanings) are included in the queries the precision of the results obtained is reduced (not desired resources are returned).

The most basic semantic relationship between terms is the *synonymy*. Two terms are synonyms if they have the same meaning. It should be able to use any of the synonyms indistinctively in a query system and obtain the same results (with records classified according to one of them). However, prefect synonymy is scarce, most of synonymy relations are partial-synonyms that only share part of their meanings. For query expansion, perfect synonyms are ideal (they mean the same, so all the available should be included in the expansion), but also close synonyms can be used if there is no possibility of meaning confusion in the context of the application. In this context, it is very important the identification of the intended meaning of each term in the user query because it can help to add additional close synonyms.

An special case of *synonymy* is the relation between terms from different languages with equivalent meaning. Multilingual terminological models have an important applicability in multilingual systems where it is needed to retrieve results independently of the language of the user query and the language used to classify the resources makes. The expansion of query terms to those equivalents to other languages greatly simplifies this task.

In addition to *synonymy*, *hypernymy-hyponymy* relationships between terms are also interesting to expand the queries. A term is hyponym of another one if its meaning is contained in the meaning of the other, being the *hypernymy* the inverse relation. In most situations, hyponyms can be directly used for expansion given that their meaning is completely included in the original one and therefore are relevant to improve the query formulation. This can even be done recursively adding the hyponyms of the hyponyms to obtain all the terms whose meaning is contained in the original one.

Hypernyms, on contrary, are discouraged for query expansion. They have a more general meaning and can include non desired items to the result set. However, in some specific contexts were it is known that the hypernym has been only used for classification with the same intended meaning than the hyponym (i.e., no records classified with other of the hypernym meanings exist in the collection), they can be used in for expansion.

More specific than general *hypernymy-hyponymy* relations, there are the *holonymy-meronymy* relationships. A term is meronym of another one if it denotes a constituent part of, or a member of something (e.g. finger is meronym of hand). The *holonymy* is the *meronymy* inverse relationship. For example, a tributary *is part of* a river, a specific mountain *is part of* a mountain chain, and a specific administrative division of the territory *is part of* another bigger one.

Meronymy relationship is especially useful to expand queries in the GIS context by the nature of the information that is mainly based on spatial earth surface divisions. For example, in a query about the “flora in Aragón” (a sub-division of Spain), the term “Aragón” could be expanded to include additional subdivisions such as “Jaca” that can be used to increase the number of suitable records returned.

Gender relation is another semantic relation to deal with. From the terminological models, usually only the dictionaries contain this relationship. However, it is very interesting for query expansion given than searching for a gender usually means an interest on searching for the other one (a search for “horses” in a specific place, also should include specific results about "mares"). This relationship is completely dependent of the language, and although in many languages, most of the times, the gender differences in the terms are reduced to small changes in the ending of the words that can be removed trough a lemmatization process in a similar way as it is done with plurals, in other situations the used words are completely different (e.g., uncle-aunt).

There are many expansion related works in the field of geographical information. For example, Jones et al. [103] combines a hierarchical distance measure (using a toponym thesaurus) with Euclidean distance between place centroids to create a hybrid spatial distance measure. This measure is integrated with a thematic distance, based on classification semantics, to create the semantic closeness measure used for relevance ranking in the retrieval system. Other works are the proposals of Alani et al. [3] and Tudhope et al. [201], which illustrate how hierarchical spatial relationships can be used to provide more flexible retrieval for queries incorporating place names in applications employing online gazetteers and geographical thesauri. Additionally, they explore how to combine spatial with associative relationships by filtering on the context of the associative link and their subtypes. Having into account the multilingual issues, query expansion can be used to provide language independent query systems by using multilingual terminological ontologies to translate the query terms to other languages [70, 162].

5.4.2 Query expansion through terminological ontologies

As it has been described previously, user queries can be expanded by using the concepts contained in terminological ontologies and the relations that hold among them to improve the results obtained. To be able to perform this expansion, the information retrieval model must have a suitable access to the required terminological models. This subsection describes an approach to provide this integration. It describes an information retrieval model for SDIs metadata catalogs, the integration with the Web Ontology Service (WOS) proposed in section 4.6 to provide access to the terminological models, and an expansion model for the user queries based on this terminology.

5.4.2.1 General context

An information retrieval process implies a series of typical operations such as text processing, indexing of documents, query processing, searching and ranking of retrieved documents. Figure 5.5 shows a schema of these operation interactions based on the model proposed by Baeza-Yates and Ribeiro-Neto [11], but customized to the special characteristics of metadata management. Additionally, the figure remarks where the interaction for query processing with the WOS component is placed.

As regards the specific decisions taken in the operations involved in this information retrieval process, this work proposes the use of CatServer for storage of the resource descriptions. This

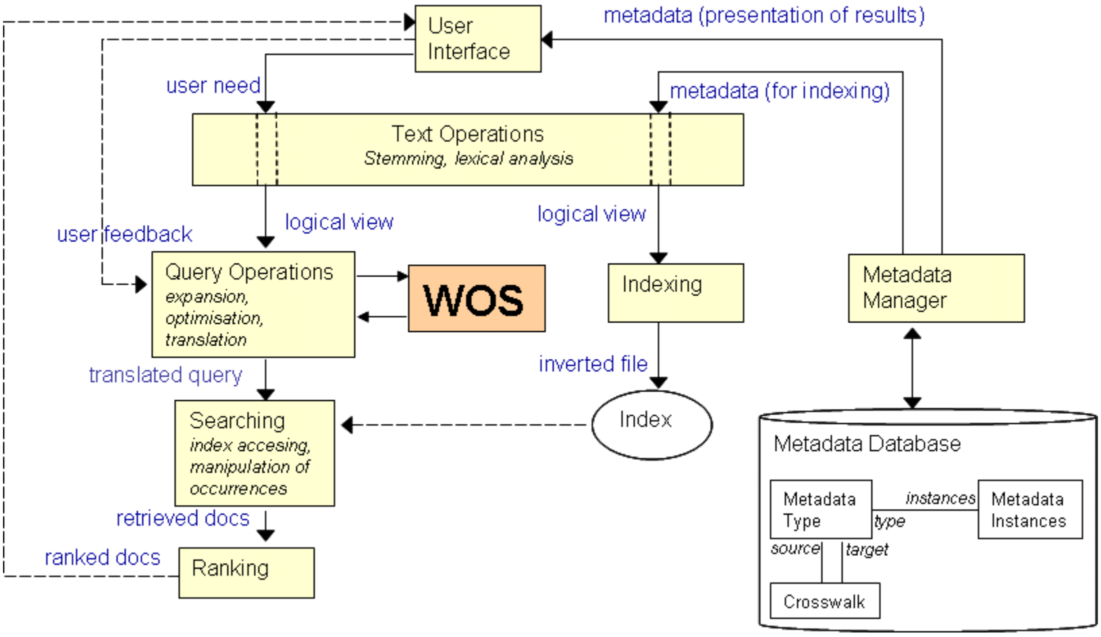


Figure 5.5: Structure of an information retrieval system (IRS) [11]

catalog system, described in Tolosana-Calasanz et al. [199], provides a functional kernel for catalog services handling XML-encoded metadata. With respect to the information retrieval model applied, CatServer is based on the Extended Boolean Model [11], i.e. it combines the simplicity of the Simple Boolean Model with the slightly more sophisticated ranking of results supplied by the Extended Model. Additionally, it is worth noting that this catalog system fulfils two main requirements. On the one hand, the system is independent from the metadata standards or schemas followed by the metadata inserted in the catalog. The idea behind this requirement is to use CatServer as a basis for the implementation of different metadata-driven services such as geographical data catalogs, service catalogs, or even Web Feature Servers (including its gazetteer variant). On the other hand, CatServer is able to manage large amounts of metadata records and be efficient enough in response time.

In order to be independent from metadata standards, two design decisions have been taken in the development of CatServer:

- Firstly, metadata are directly stored in XML at CatServer. This modus operandi is significantly different from other catalogs which convert the XML into a persistent object model. The great advantages of the adopted approach are its retrieving speed (since it only has to retrieve the XML) and its independence from metadata standards. Otherwise, as it happens with the persistent object model approach, the inclusion of new standards involves code rewriting.

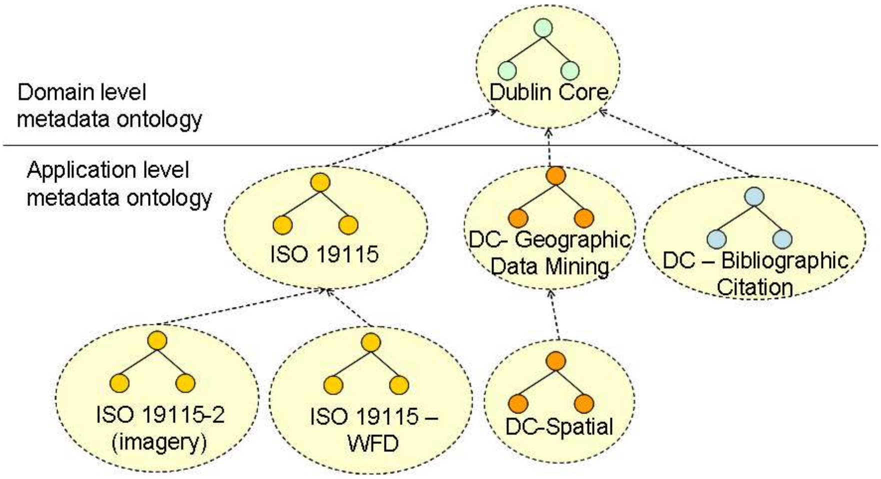
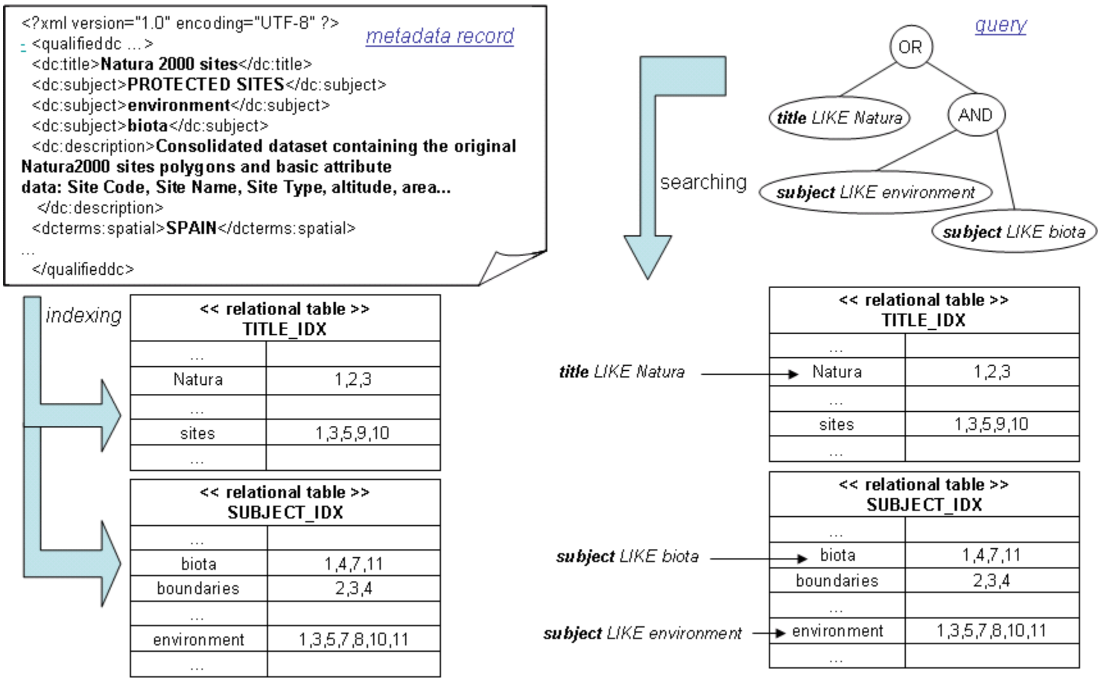

Domain level metadata ontology
Dublin Core
Application level metadata ontology
ISO 19115
DC- Geographic Data Mining
DC – Bibliographic Citation
ISO 19115-2 (imagery)
ISO 19115 – WFD
DC-Spatial

Figure 5.6: A Hierarchy of Metadata Ontologies

- Secondly, apart from the storage in XML format, the independence from metadata standards is fulfilled thanks to the fact that the different metadata schemas share a common core [198]. This common core is needed if the system wants to provide the user with the functionality of querying all the metadata instances stored, independently of the metadata schema used (e.g. we need a common set of queryable properties). As depicted in figure 5.6, the only prerequisite of the standards supported by our system is to provide their XML Schema and their mapping to the common core of Dublin Core. That is to say, as it is shown in figure 5.5, the metadata database maintains a knowledge base of the supported *metadata types* (schemas) and the *crosswalks* between them (at least a crosswalk towards the Dublin Core common core).

With respect to the need related to the efficiency and the management of huge amounts of metadata records, it must be noted that the Inverted Index structure [11] was chosen and adapted to speed up queries. This structure could be defined as a sequence of *(key, pointer)* pairs where each pointer refers to a record in a database which contains the key value in some particular field. The index is sorted by the key values to provide fast searching for a particular key value (e.g. using binary search). The index is "inverted" in the sense that the key value is used to find the record rather than the other way around. For catalog systems enabling searches with filters on more than one database field, multiple indexes (sorted by those keys) may be created.

The index structure of CatServer is slightly different. It consists of a pair *(key, array)* where the key has the same meaning, but there is an array instead of a pointer to a register. The array is a metadata identifier array which represents those metadata records that contain the word in a specific XML metadata element tag. The index structure has been implemented by means of a relational database table. The usual way of working is to build an Inverted Index for every XML metadata element tag for which the clients need to search. Figure 5.7 (left) shows two Inverted Indexes built over the Dublin Core elements *title* and *subject* (the examples uses an excerpt of metadata describing the *Natura 2000 sites* dataset, a set of areas of special interest for biodiversity protection across Europe).



```
<?xml version="1.0" encoding="UTF-8" ?>
- <qualifieddc ... >
  <dc:title>Natura 2000 sites</dc:title>
  <dc:subject>PROTECTED SITES</dc:subject>
  <dc:subject>environment</dc:subject>
  <dc:subject>biota</dc:subject>
  <dc:description>Consolidated dataset containing the original
Natura2000 sites polygons and basic attribute
data: Site Code, Site Name, Site Type, altitude, area...
   </dc:description>
  <dcterms:spatial>SPAIN</dcterms:spatial>
...
  </qualifieddc>
```

<< relational table >>
TITLE_IDX

...	
Natura	1,2,3
...	
sites	1,3,5,9,10
...	

<< relational table >>
SUBJECT_IDX

...	
biota	1,4,7,11
boundaries	2,3,4
...	
environment	1,3,5,7,8,10,11
...	

Figure 5.7: Retrieval example: XML tags and Inverted Index implementation correspondence (left); querying process (right)

Once the indexes are built, the system can retrieve the information with only the tag name, which determines the index to examine, and the key. For instance, let us consider the query represented in figure 5.7 (right). This query aims at retrieving those metadata records whose *title* contains *Natura* or whose *subject* contains *biota* and *environment* (*title LIKE '%Natura%' OR (subject LIKE '%biota%' AND subject LIKE '%environment%')*). Thus, CatServer would obtain three arrays of metadata identifiers: one for *Natura*, one for *biota*, and another for *environment*. The next step in the process is to combine these arrays as sets of metadata records. The *AND* implies an intersection operation between the *biota* array and the *environment* array. The *OR* implies a union operation between the *Natura* array and the subset obtained in the previous step.

Evidently, not all the results are equally important. As mentioned at the beginning of this subsection, the ranking process is based on the Extended Boolean Model. Therefore, the subset of metadata is in fact a list of metadata records ordered by relevance. Following with the example, metadata records satisfying both operands of the *OR* logic expressions are more relevant than those which only satisfy one of them, i.e. they appear before in the ranked list.

5.4.2.2 Proposed expansion model

The use of any of the semantic relations described in section 5.4.1 for query expansion requires its identification in the specific application area of the system and its combination. In an equivalent way, instead of queries indexes can be expanded with equivalent terminology. It increases the indexes size but provide faster query systems because expansion is executed off-line.

Terminological ontology can provide the semantic relationships needed for the expansion. However, depending on the area of interest it may be difficult to find a suitable ontology model. In this case, the combination of the knowledge provided by different models can be used to gather enough terminology to properly enrich the queries.

In this context, a method to expand user queries by making profit of the knowledge behind the terminological ontologies managed by WOS has been developed.

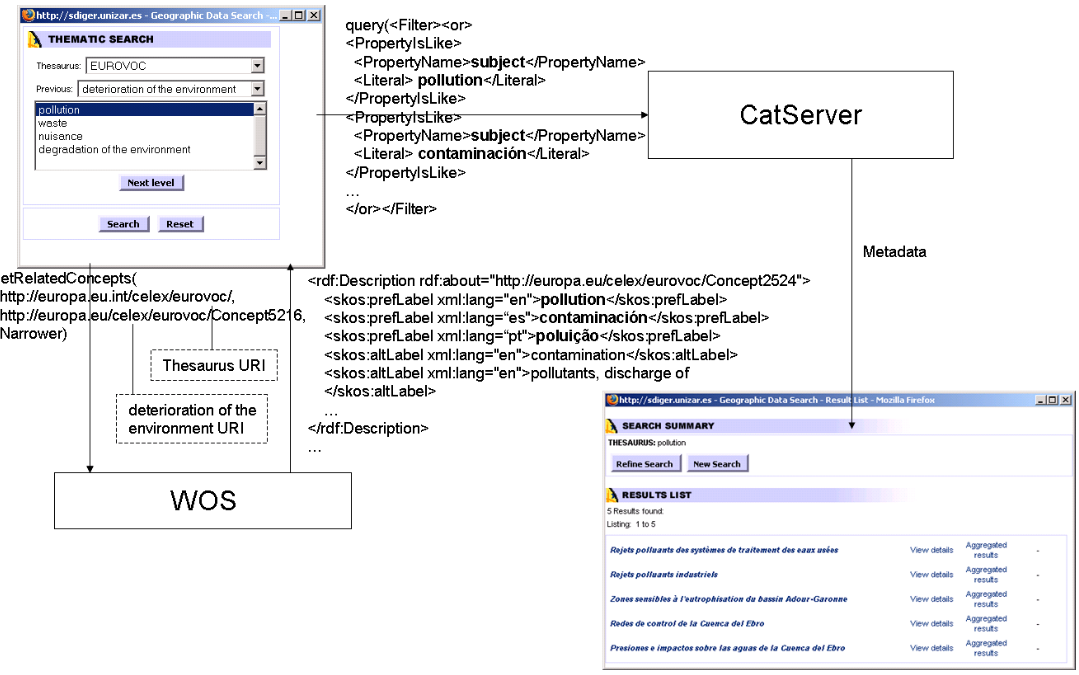
As it is described in section 4.4.2, the WOS provides access to the stored terminological ontologies but additionally, provides a disambiguation mechanism to relate the terminological models stored in the repository. The provided mappings can be used as an additional semantic base to produce better query expansions.

This query expansion model is similar to works like Tudhope et al. [203] or Clark et al. [30], which present systems where thesauri are used as the basis for discovery services, and the thesaurus hierarchical structure helps to find resources either directly related to the "concepts" found in user queries or "closely" related to "the user's concepts of interest".

As depicted in figure 5.5 the basic functionality provided by CatServer is extended with a module that processes the terms included in the user query in order optimize and expand them with related terms obtained through a WOS service. Assuming that the user is guided by an initial terminological ontology, the *Query Operations* module will expand the user queries in two directions:

- *Expansion through the initial terminological ontology.* Firstly, the concepts selected by the user through an initial terminological ontology (and displayed in a particular language) are expanded with all the existing alternative labels in the different languages supported by this initial terminological ontology. By alternative labels of a concept we mean the preferred labels of this concept in all the languages supported by the ontology and all the synonym labels of this same underlying concept in those languages.

- *Expansion through disambiguation (related terminological ontologies).* Secondly, the *Query Operations* module tries to expand the query with the labels corresponding to related concepts in other terminological ontologies managed by WOS. Using the disambiguation component, described in section 4.4.2, it is possible to interrelate ontologies thanks to the connection with an upper-level ontology. If the user selects a very specific concept in the initial terminological ontology, this strategy will not probably find similar concepts in other ontologies. But in the case of searching more general concepts, this strategy will help to find synonyms or translations existing in related ontologies, which may have a richer vocabulary or support more languages than the initial one.



http://sdiger.unizar.es - Geographic Data Search -...

THEMATIC SEARCH

Thesaurus: EUROVOC

Previous: deterioration of the environment

pollution
waste
nuisance
degradation of the environment

```
query(<Filter><or>
<PropertyIsLike>
  <PropertyName>subject</PropertyName>
  <Literal> pollution</Literal>
</PropertyIsLike>
<PropertyIsLike>
  <PropertyName>subject</PropertyName>
  <Literal> contaminación</Literal>
</PropertyIsLike>
...
</or></Filter>
```

```
getRelatedConcepts(
  http://europa.eu.int/celex/eurovoc/,
  http://europa.eu/celex/eurovoc/Concept5216,
  Narrower)
```

```
<rdf:Description rdf:about="http://europa.eu/celex/eurovoc/Concept2524">
  <skos:prefLabel xml:lang="en">pollution</skos:prefLabel>
  <skos:prefLabel xml:lang="es">contaminación</skos:prefLabel>
  <skos:prefLabel xml:lang="pt">poluição</skos:prefLabel>
  <skos:altLabel xml:lang="en">contamination</skos:altLabel>
  <skos:altLabel xml:lang="en">pollutants, discharge of
  </skos:altLabel>
  ...
</rdf:Description>
...
```

http://sdiger.unizar.es - Geographic Data Search - Result List - Mozilla Firefox

SEARCH SUMMARY

THESAURUS: pollution

RESULTS LIST

5 Results found:

Listing: 1 to 5

Rejets polluants des systèmes de traitement des eaux usées	View details	Aggregated results	-
Rejets polluants industriels	View details	Aggregated results	-
Zones sensibles à l'eutrophisation du bassin Adour-Garonne	View details	Aggregated results	-
Redes de control de la Cuenca del Ebro	View details	Aggregated results	-
Presiones e impactos sobre las aguas de la Cuenca del Ebro	View details	Aggregated results	-

Figure 5.8: Example of query expansion for a thematic catalog

Figure 5.8 shows an example of the first type of query expansion (*expansion through the initial terminological ontology*). This example can be described in a sequence of three main steps:

- Firstly, a thematic search interface allows the user to browse the concepts contained in a terminological ontology (left side of figure 5.8). Although the search interface only shows the preferred labels in the language the user has selected, it can be assumed that the terminological ontology is multilingual, i.e. it gives support for several languages[13]. Whenever the user browses the narrower concepts of a first concept (e.g., the concept *deterioration of the environment* identified by the URI *http://europa.eu/eurovoc/Concept5216* in

[13] This search interface belongs to the set of search services offered by the SDIGER project - http://sdiger.unizar.es- (see section 5.4.3 for more details).

the EUROVOC terminological ontology), the thematic search interface interacts with the WOS service to retrieve all the preferred and alternative labels of the narrower concepts and in all the available languages (e.g., *pollution* in English but also *contaminación* in Spanish). Figure 5.8 shows an excerpt of the *getRelatedConcepts* request sent to the WOS service and the response in SKOS format returned by WOS.

- Secondly, a click on the *search* button represents that the user has stopped browsing the terminological ontology and has decided the final concepts to be included in the query. At this moment the search interface constructs the query that will be sent to the catalog system (CatServer). This query is compliant with the OGC Filter encoding specification [215] and contains an expression that includes all the possible alternatives of preferred and alternative labels in different languages obtained from the WOS service.
- And thirdly, the CatServer system launches the searching and ranking processes to obtain the metadata records that satisfy the expanded user query. Thanks to the fact that WOS provides preferred terms in different languages, the returned metadata records may have been written in multiple languages. For instance, the results shown on the right side of figure 5.8 include records in French (*Rejets pollutants des systèmes ...*) and Spanish (*Presiones e impactos sobre ...*).

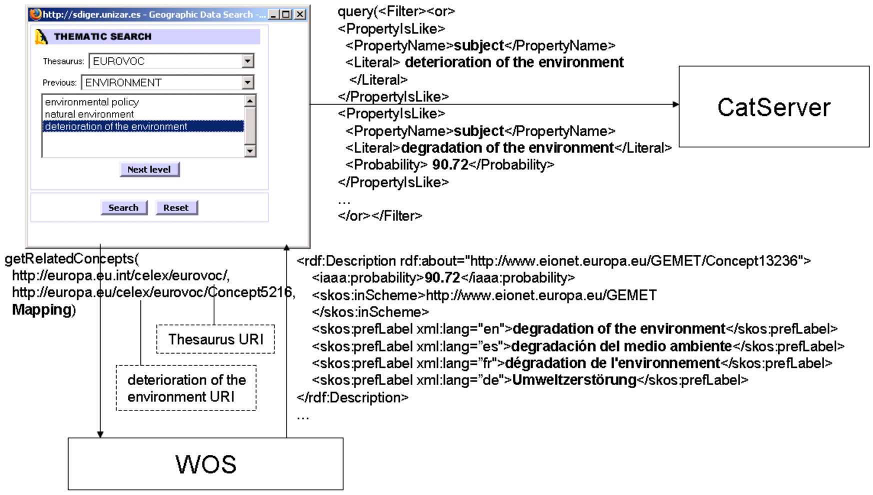
With respect to the second strategy for query expansion (*expansion through disambiguation*), the *Query Operations* module applies a basic routine to estimate the reliability of expanding an original set of keywords with a new term belonging to a new different ontology, not used in the original set. This basic routine consists of four steps:

- The first step is the collection of more liable mappings of the concepts in the original query with respect to the upper-level ontology used by the WOS service. From now on we will use the name *synset* for these major mappings because this is the name given to the concepts in WordNet, which is the upper-level ontology used for the disambiguation functionality described in section 4.4.2. As a result of this first step, we obtain for each concept in the query an initial collection of *synsets.*
- Secondly, we will also collect the *synsets* corresponding to a concept from a different ontology, which may be a candidate for query expansion. Initially, all the concepts of the ontologies stored in the WOS are considered as candidates.
- Thirdly, we will compute the reliability of a new candidate concept as the number of *synset* coincidences with the *synsets* of the original query concepts divided by the number of *synsets* of the new concept and multiplied by 99:

$$reliability = \frac{|synset\ matches\ of\ new\ concept|}{|synsets\ of\ new\ concept|} \times 99 \tag{5.4.1}$$

The reason to use a final factor of 99 and not 100 in equation 5.4.1 is to obtain a maximum reliability percentage of 99 for automatically expanded concepts, reserving uniquely a 100-reliability percentage for the concepts which were originally in the query.

- Finally, the reliability of a new candidate concept is compared with a *threshold* reliability. If the reliability percentage is greater than a *threshold* reliability, the query is expanded with this new concept. This means that the query expression will include as alternatives the preferred and alternative labels of this new concept in the different languages available. A *threshold* of 50% is considered as an appropriate value to detect suitable concepts related to the initial set.



http://sdiger.unizar.es - Geographic Data Search -...
THEMATIC SEARCH
Next level
Search
Reset
CatServer
Thesaurus URI
deterioration of the environment URI
WOS

Thesaurus: EUROVOC

Previous: ENVIRONMENT

environmental policy
natural environment
deterioration of the environment

```
query(<Filter><or>
<PropertyIsLike>
  <PropertyName>subject</PropertyName>
  <Literal> deterioration of the environment
   </Literal>
</PropertyIsLike>
<PropertyIsLike>
  <PropertyName>subject</PropertyName>
  <Literal>degradation of the environment</Literal>
  <Probability> 90.72</Probability>
</PropertyIsLike>
...
</or></Filter>
```

```
getRelatedConcepts(
  http://europa.eu.int/celex/eurovoc/,
  http://europa.eu/celex/eurovoc/Concept5216,
  Mapping)
```

```
<rdf:Description rdf:about="http://www.eionet.europa.eu/GEMET/Concept13236">
   <iaaa:probability>90.72</iaaa:probability>
   <skos:inScheme>http://www.eionet.europa.eu/GEMET
   </skos:inScheme>
   <skos:prefLabel xml:lang="en">degradation of the environment</skos:prefLabel>
   <skos:prefLabel xml:lang="es">degradación del medio ambiente</skos:prefLabel>
   <skos:prefLabel xml:lang="fr">dégradation de l'environnement</skos:prefLabel>
   <skos:prefLabel xml:lang="de">Umweltzerstörung</skos:prefLabel>
</rdf:Description>
...
```

Figure 5.9: Expansion through disambiguation

The core of this expansion technique is integrated in the WOS service. The disambiguation functionality is provided through the *getRelatedConcepts* operation, using *Mapping* as relation type. Figure 5.9 shows an example of this type of query expansion. The concept *deterioration of the environment* belonging to the EUROVOC vocabulary is expanded with the concept *degradation of the environment* of GEMET. This new concept of GEMET has been mapped to the original concept of EUROVOC with a reliability of 90.72%.

5.4.3 Testing the retrieval model

In order to quantify the retrieval effectiveness of an information retrieval system, performance measures such as precision (number of relevant hits divided by the number of hits) and recall (number of relevant hits divided by the number of relevant documents) must be computed upon the results obtained from evaluation experiments, which are conducted under controlled conditions. This requires a testbed comprising a fixed number of documents, a standard set of queries, and relevant and irrelevant documents in the testbed for each query.

For the case of testing the retrieval model and verifying the influence of WOS in the improvement of information retrieval performance, this model has been applied within the context of the SDIGER project. This project includes a thematic catalog searcher (see left side of figure 5.8) that makes use of a WOS instance to access multilingual terminological models such as thesauri and to help in the construction of user queries, which are automatically expanded with cross-language terminology by means of the strategies explained in section 5.4.1. The multilingual models managed by the WOS instance integrated within the SDIGER project are the Multilingual Agricultural Thesaurus (AGROVOC), the European Vocabulary Thesaurus (EUROVOC), the GEneral Multilingual Environmental Thesaurus (GEMET), and the UNESCO Thesaurus. All of them have been defined by well-known organizations and give support to several European languages, at least the three ones required for the project: English, French and Spanish. Other projects were the same model has been applied (although they are not used for the test described here) are the Spanish SDI[14], and the SDI-EBRO prototype[15].

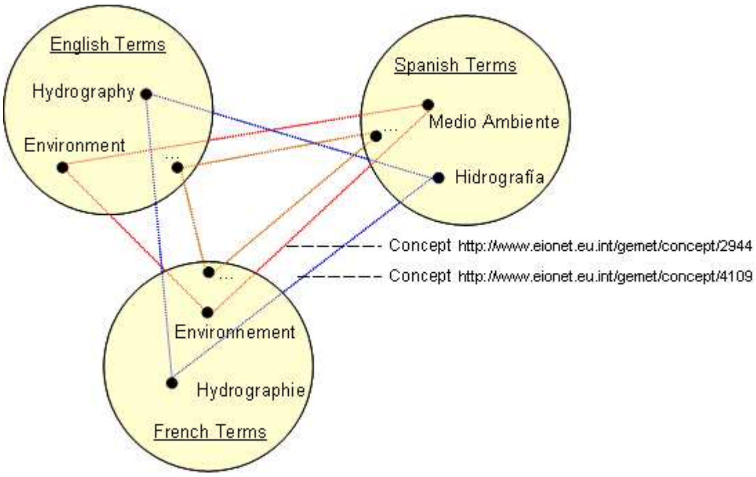


Figure 5.10: Mapping between terms in different languages

The SDIGER metadata corpus consists of around 26,000 metadata records in Spanish, English and French, which contain about 350 different keywords (in Spanish, English and French) to describe their associated data. Many keywords in such metadata records have been extracted from different thesauri but others have been randomly typed by metadata creators. In addition, each metadata record is written only in one language and this includes the terms used as keywords. All these characteristics make the corpus appropriate to analyze the impact

[14] http://www.idee.es/

[15] http://80.255.113.15/portalIDE-Ebro/Default.vm

of multilingual dispersion in the information retrieval performance.

Previous to the performance analysis, it was necessary to obtain a series of topics and their relevance with respect to metadata records. This way, it would be possible to compare different retrieval (and query expansion) strategies. The topics were selected upon an analysis of the concepts behind the 350 different keywords found in the metadata records. After mapping terms in different languages, identifying synonyms, and eliminating redundancies introduced by plurals and other derived lexical forms, 204 different concepts were obtained.

To extract the topics, the following semi-automatic process was developed. Firstly, the language of each of the 350 keywords was identified by means of the *language* descriptor found in the metadata records containing these keywords. Additionally, the language assigned to each element was verified with a multilingual dictionary. Secondly, as shown in figure 5.10, a manual mapping between terms in different languages was applied to identify the concepts that would be used later as topics for the experiments. Thirdly, the identification of synonyms and elimination of related lexical forms was applied as well with the aid of a multilingual dictionary. Finally, spatial data experts from the institutions contributing to the SDIGER project assigned manually the relevance of metadata records with respect to each topic.

For the sake of facilitating topic relevance assignment, experts were provided with the *Inverted Indexes* automatically created by CatServer and an initial pre-assignment of relevance according to the following rule: *"a metadata record is relevant to a topic if it contains one of the possible terms (labels) that represent the concept in that topic"*. The experts only had to revise this initial pre-assignment for possible mistakes due to word-sense ambiguity. However, in most cases the initial pre-assignment was accurate. In contrast to text information retrieval, where full documents are indexed, in this case we are indexing metadata records, which are short summary texts created by experts. This has two important advantages in comparison with classical text information retrieval. On the one hand, the texts are short and there are few noise words, reducing the possibilities of mistaking a noise word for a label representing a real topic of the resource described. And on the other hand, most terms found in metadata are quite specific, reducing the possibilities of polysemy. In fact, a search system just based on word-matching of topic terms would yield a high precision. The main problem that affects the performance of search systems over this metadata corpus is the problem of detecting the correspondence among translation of terms and some synonymy issues. That is to say, a simple word-matching strategy for retrieval yields a low recall.

Once the corpus was fully established, a series of experiments were conducted using CatServer in order to compare different alternatives for query expansion. These experiments can be classified into three categories according to the query expansion strategy applied:

- *No query expansion.* The first three experiments consisted in selecting a particular language (e.g., English, French or Spanish) and sending queries to CatServer using topic terms in that particular language without applying any strategy for query expansion. In other words, these three experiments were oriented to study the three original languages separately (used in metadata records) and the problems derived from the multilingual dispersion.

- *Expansion through the initial ontology.* A second series of three experiments was oriented to analyze the effect of expanding queries thanks to the knowledge stored in the terminological ontologies. This strategy matches with the first heuristic described in section 5.4.1 for query expansion. In these experiments, it is assumed that the user is browsing a terminological ontology (GEMET, AGROVOC, or UNESCO) for the definition of user queries. When the user decides the final concepts to include in the query, the user query is automatically expanded with all the existing terms in different languages for those user selected concepts.

- *Complete query expansion.* The final experiment is devoted to analyze the effect of applying both two heuristics for query expansion described in section 5.4.1. This experiment assumes that the user is using the GEMET terminological ontology for the definition of the query. As regards to the query expansion, apart from extending the topic concepts to the all possible terms, the expansion also considers related concepts in the terminological ontologies of AGROVOC and UNESCO. Using the strategy called *expansion through disambiguation*, based on the disambiguation mechanism (see section 4.4.2) and the reliability formula (explained in section 5.4.1), the concepts of GEMET were connected to related concepts in UNESCO and AGROVOC.

With respect to the performance measures obtained upon these experiments, it is worth mentioning that they have focused on the analysis of recall. Given the characteristics of the metadata corpus, the comparison of precisions for each experiment is not relevant. As stated before, the results obtained in experiments not using query expansion always get a high precision because the metadata collection contains very specific concepts, which are rarely affected by polysemy conflicts. Additionally, the topics used for the queries correspond to concepts extracted from the own keywords contained in metadata records. This can be also extrapolated to the other two series of experiments using query expansion. Again, thanks to the lack of polysemy and the specificity of the topics used in the experiments, the automatic expansion is supposed to be precise. On the one hand, the translations of terms derived from the use of a terminological ontology are inherently accurate (the terminological ontology has been constructed by experts with knowledge in different languages). On the other hand, the expansions due to the mappings of concepts between different ontologies are also accurate because of the specificity of the topics.

Figure 5.11 shows the recall curves obtained in each of the aforementioned experiments.

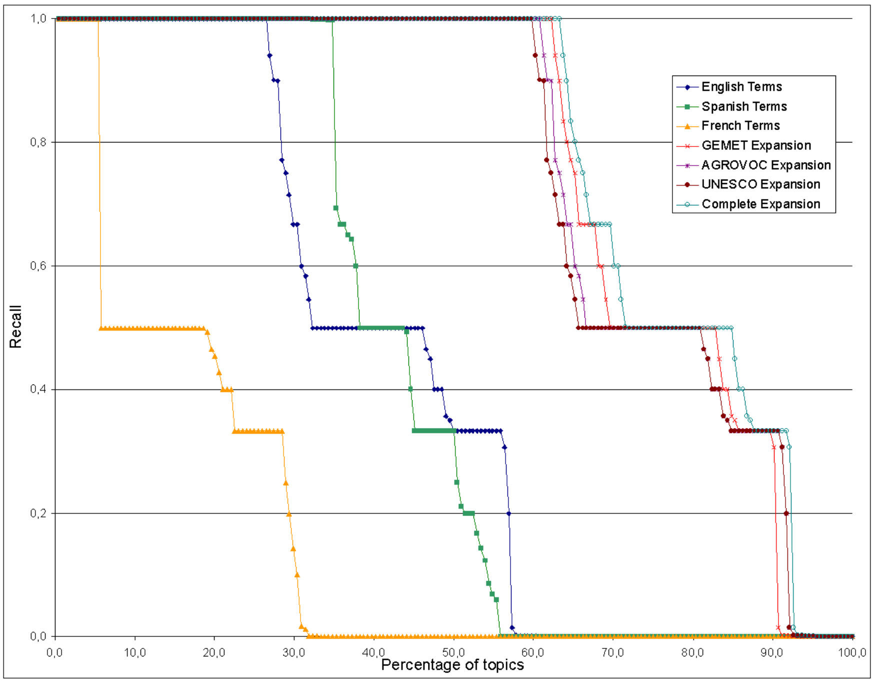


Figure 5.11: Comparison of recall using different query expansion alternatives

The topics in the *x-axis* of each recall curve are ordered by the recall obtained in the experiment strategy. This fact does not allow the comparison of recall for a particular topic in two experiments. However, the main purpose of the figure is to provide a general idea of the average recall in each experiment. The area covered in the polygons bounded by the recall curves and the positive sides of both *x-axis* and *y-axis* denotes the recall improvements in each experiment.

As a result of the experiments, it can be observed that query expansion strategies based on terminological ontologies (i.e., use of translations and synonyms) imply an important recall improvement. Without query expansion, only 6% of topics using French terms have a full recall. This is slightly improved in the case of experiments using English and Spanish terminology: 26% and 32% of topics with full recall respectively. But anyway, it can be verified that this strategy produces low recall measures in a multilingual corpus. Quite the opposite, the experiments guided by the use of a terminological ontology such as GEMET, AGROVOC and UNESCO obtain a high recall for most of the topics: 60% of topics have a full recall, and 80% of topics have a recall higher than 50%. Finally, the experiment using complete query expansion provides a small increase in recall with respect to the use of a single ontology.

Theoretically, the experiment with complete query expansion should have been obtained a perfect recall. However, there are still a small number of concepts that are not contained in the terminological ontologies used for the experiments. It must be taken into account that topics are derived from the keywords found in metadata records, but these keywords may not have been selected from a terminological ontology. Additionally, the labels (terms in multiple languages) used for a concept in a terminological ontology may not necessarily match with the terms manually mapped for the extraction of topic concepts.

Finally, it must be noted that independently of the query expansion method and the terminological ontology used, the results obtained are similar. This is caused by the fact that the ontologies selected for the experiment are thematically related to the metadata collection and contain a subset of concepts which are similar to the keywords contained in metadata records.

This information retrieval model based on the WOS component has been applied to different SDI projects (Spanish SDI[16], SDIGER Geo-portal and, SDI-EBRO prototype) to improve queries through the use of several multilingual thesauri such as GEMET, UNESCO, AGROVOC, and EUROVOC. The search clients constructed for these systems have been based on the client prototype described in section 4.6 but adjusting them to the specific requirements of each system (in general, through the change of the configuration parameters).

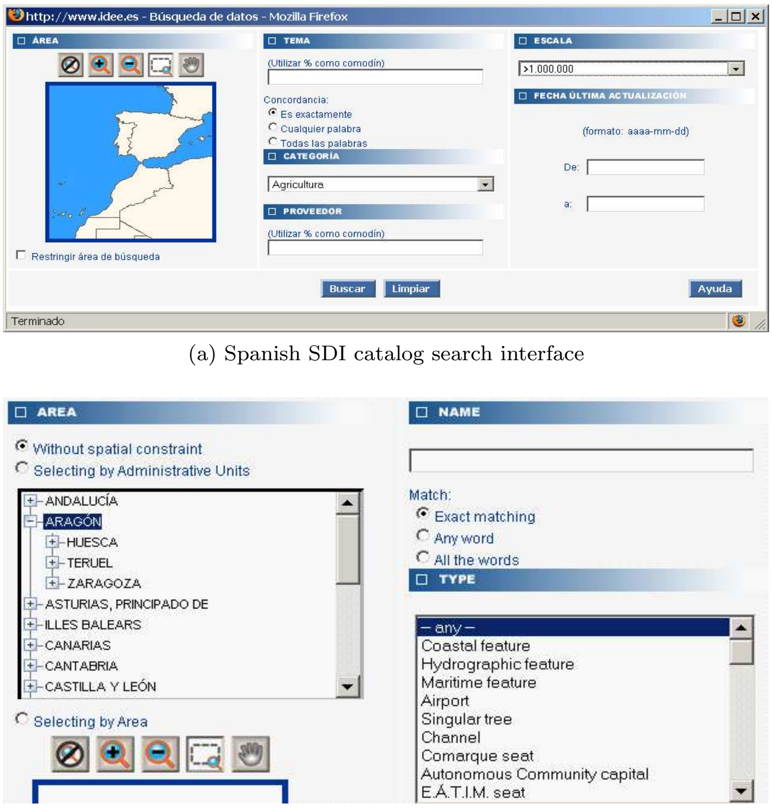
From the different projects where the WOS has been integrated, the Spanish SDI project is the most relevant. The search services of the infrastructure use a collection of search components (e.g., graphical display to restrict results by coordinates, or a free text area to introduce query elements) to construct the final user queries that include some components constructed according to the architecture described in section 4.6 to access to the WOS. Figure 5.12 shows as example two of these search services. On the one hand, figure 5.12a depicts the catalog search service of the Spanish SDI. In this search interface, the controlled lists that are shown (category, scale) are specialized WOS clients that provide access to the required terminology. On the other hand, figure 5.12b shows a subset of the Spanish SDI Gazetteer service interface that contains two different WOS clients. The first one (on the right of the figure), shows all the types of features contained in the gazetteer (it is a controlled vocabulary without hierarchy). The second one (on the left of the figure), shows the content of the Spanish administrative divisions thesaurus.

5.5 Information browsing

The use of terminological models in search interfaces is an effective way to simplify the construction of queries to the user. However, they cannot be directly applied in all situations, especially if the search interfaces provides browsing for the resources through the terminological ontology structure.

To describe properly a resource, metadata contain thematic information showing what the resource is about. These thematic elements are usually filled with concepts extracted from some kind of terminological model in the area of interest. When no suitable terminological

[16] http://www.idee.es/


http://www.idee.es - Búsqueda de datos - Mozilla Firefox
□ ÁREA
□ TEMA
(Utilizar % como comodín)
□ ESCALA
□ FECHA ÚLTIMA ACTUALIZACIÓN
(formato: aaaa-mm-dd)
De:
a:
Concordancia:
○ Cualquier palabra
○ Todas las palabras
□ CATEGORÍA
□ PROVEEDOR
(Utilizar % como comodín)
□ Restringir área de búsqueda
Buscar
Limpiar
Ayuda
Terminado

Concordancia: ● Es exactamente

CATEGORÍA: Agricultura

ESCALA: >1.000.000

(a) Spanish SDI catalog search interface

AREA: ● Without spatial constraint

Match: ● Exact matching

TYPE: – any –

(b) Section of Spanish SDI Gazetteer search interface

Figure 5.12: Spanish SDI search services

ontologies are available, the metadata creators use general purpose terminological models that are not focused on the subject the data is about, to create a new model with the elements they require (using processes such as the described in section 3).

However, a terminological ontology should contain all the possible terminology that may be required in the analyzed area of interest, not only the required for a specific collection, to be able to be used in several contexts. Therefore, some of the concepts of the ontology model may have not been used in the collection, and a search using them would return zero results. This is especially important in browsing interface, where the user is guided to the resources using a terminological model. It is not suitable to provide a term for selection if it is going to return any results.

Additionally, even if all the concepts of the terminological model have been used for classification, when the used model is too big, it becomes difficult to know what the collection is about. It is needed to provide a small classification of the collection that acts as an overview of the main themes in the collection.

The provided terminological models have to be adjusted to the user needs. In this context, the analysis of the metadata describing data collections is required to generate a suitable terminological model to search them. Depending on the user needs, the desired model can range from a whole terminological model such as the used previously but adapted to the collection, to a much reduced list of the general themes the collection is about.

This section describes a set of processes to generate terminological models of different specificity from a data collection that can be later be used to be provided to facilitate the browsing and the identification of the collection content. It describes two different approaches, one based on the construction of a topic map, and the other one based on the use of different clustering techniques to generate a collection classification.

5.5.1 State of the art in information browsing

If the terminological model used for the classification (or annotation) of data is available, the content of data collections may be browsed using this terminology. However, a common situation when integrating data from different collections is not to know the origin of the keywords used for the creation of metadata. In this context, the keywords can be used to try to reconstruct the original terminological ontology. Although a process of this type can only generate a model which is poor in relations, it has the advantage of reflecting the current contents of the collection.

From the different terminological ontology models that can be generated from the keywords, the most suitable for information browsing is the topic map. The structure of topic maps has been specifically designed to associate terminological concepts to resources that are about the theme (simpler models does not allow storing the relation of a keyword with the resources containing it). This structure makes the browsing process immediate (the associated resources are related to the topic), but requires a reconstruction of the topic map if there are changes in the associated collection. Depending on the detail required, the resulting model can vary from a detailed topic map of the collection to a reduced classification containing only the main topics of the collection.

Several works have advanced in how to construct automatically topic maps in a way that represents the source collection in a trustworthy way [2, 39, 183]. Boulos et al. [21] presents a system that creates a graphic topic map to enhance the access to a medical database using a graphical representation to locate the different topics over a graph of the human body. Schlieder et al. [184] and Schlieder and Vögele [183] insist on the idea of accessing to metadata collections through a network of intelligent thumbnails, being those thumbnails either concepts or locations. For instance, in the case of locations, the network of thumbnails corresponds to the hierarchical structure of administrative toponyms. Schlieder and Vögele [183] proposes the use of XTM

[192] as the topic map exchange format, which can be easily visualized by a wide range of tools compliant with this format. Demšar [39] and Podolak and Demšar [174] provide a visual data mining approach to explore in an easier way the complex structure/syntax of geographical metadata records. Podolak and Demšar [174] also describe a system to clusterize a collection of metadata into clusters of similar metadata elements using the cluster algorithm of Fisher [55]. Albertoni et al. [4] describes a system to visually show traditional representations of statistical information about a collection to the user to facilitate him the identification of patterns.

Many of these works base their generation algorithms on pattern analysis techniques, known as clustering, that are focused on dividing resource collections (i.e., metadata) into clusters of resources with similar characteristics. These techniques are useful to find groups of metadata records (clusters) that share similar values in one (or more) metadata elements according to a mathematical correlation measure (e.g. the Euclidean distance, Bernoulli, Gaussian or polynomial functions among others). It can be said that elements in a cluster share common features according to a similarity criteria [39].

Clustering techniques have proved to produce good results in the classification of big collections of resources for different purposes (data mining, signal analysis, image processing…) [191]. Kaufman and Rousseeuw [110] and Jain and Dubes [100] shown a great variety of clustering and classification techniques and their applications.

The MetaCombine project [116] is a good example of classification of heterogeneous metadata. It focuses on providing a browsing service, i.e. the exploration or retrieval of resources (OAI and Web resources), through a navigable ontology. It shows the effectiveness of different clustering techniques for heterogeneous collections of metadata records according to different factors, such as the time used to locate a resource, the number of clicks needed to reach it or the number of failures in locating the resource. Other related works are the proposed by Kang [109] that proposes a clustering method based on the document keywords; or Krowne and Halbert [116] and Cutting et al. [35] that use the clustering to improve browsing interfaces of digital libraries, classifying resources in such a way that these classifications can be used later for information browsing.

Clustering techniques can be grouped according to the way each element is associated to each cluster in two different families of techniques:

Hard clustering: It associates each element of the collection to a unique cluster. An example of this category is *K-means* and its variants [45, 110].

Fuzzy clustering: It associates each element to each cluster with a different probability. It includes fuzzy and probabilistic techniques between others. Some examples are fuzzy *C-means* [200] and finite mixture models (as Bernoulli or polynomial) [6].

In this section, techniques from each of the families have been used to obtain a small classification of a resource collection that identifies the main topics.

5.5.2 Generation of topic maps

As it has been described in section 2.2.1.5, topic maps are a representation of knowledge, with an emphasis on the find-ability of information. The international standard ISO 13250 [89] that defines the topic map concept (the standard proposes a model for representation and an interchange format) indicates that topic maps allow easy and selective browsing to the requested information, showing a thematic view of the collection of metadata, facilitating in that way the access to the information.

This subsection describe a method to extract a summary of the contents of a metadata collection generating a topic map from the thematic description of the collection on the basis of the thesaurus structure of the keywords used in the metadata collection. The main objective of the generated topic map will be to provide a thematic global vision of the collection giving information about what metadata records in the collection use each term of the topic map and the weight of each keyword in the collection. This generated model can be directly used for information browsing.

Another objective is to obtain a classification of the collection more reduced than the list of all topics used in the metadata records, to circumscribe the collection to a concrete theme.

The objective of the process described here is to extract a topic map from a metadata collection using the keywords section of the metadata records. The problem has been circumscribed to a collection of metadata records that, in their keywords section, use terms from a single terminological ontology following the thesaurus model. The process to automatically generate the topic map from the metadata collection is shown in detail in Figure 5.13.

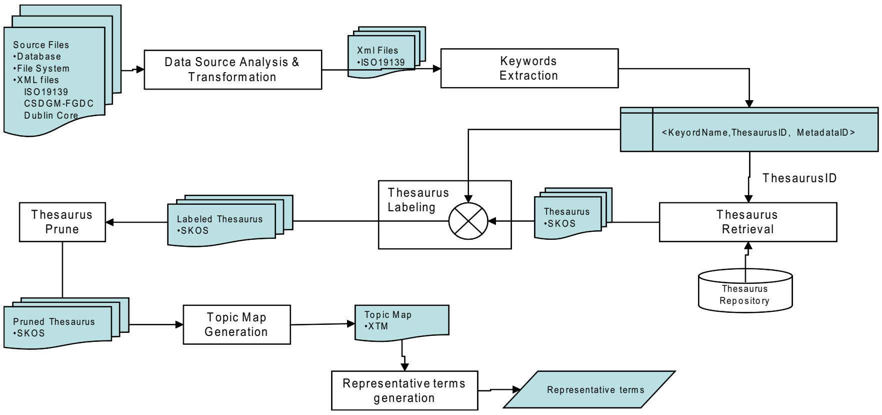


Figure 5.13: Topic Map generation Process

The first step is to obtain the metadata collection in a format readable for our system. The metadata records can be stored in different repository sources such as a metadata catalog, a relational database, or even in a directory of XML files; and their structure can follow different standards such as ISO 19115 [87], Dublin Core [88] or CSDGM-FGDC [51]. Therefore, metadata in different formats have to be transformed into a common one. In general, in a SDI context, the XML format described by ISO-19139 [95] standard is the representation format selected for metadata, and it has been used as the input for the process of topic map generation. Metadata in other formats have to be transformed to it. For the most typical situations where the source format is an XML file following other metadata standard, transformation process such as the crosswalks described by [163, chap. 3] can be used. For more special situation, dedicated transformation software may have to be constructed to perform the transformation.

The second step is the analysis of the metadata collection and the generation of the list of triplets <KeywordName, ThesaurusID, MetadataID> with the values of the keywords of the metadata, the identifier of their source thesaurus and the identifier of the metadata record where the keyword has been found. Such triplets are the base elements used to construct the topic map classification.

The third step consists in the retrieval of the thesaurus used to create the keywords of the metadata collection. A thesaurus can be provided by different kind of applications in different formats, but to be able to use it in our system to create a topic map it has to be manually obtained and transformed into SKOS core format [148] using the process described in section 3.2. Once the thesaurus used in the metadata records has been identified and retrieved, it is labeled with the information of the triplets obtained in the second step. This labeling process considers not only the direct uses but also the inherited through the thesaurus hierarchy, storing separately the direct relations from the inherited ones. This separation allows identifying in future steps if a term is used directly by some metadata records or if the used is one of its descendants.

The next step consists in pruning the branches and leaves of the labeled thesaurus whose terms are not referenced directly or by inheritance in any metadata record of the collection. Only terms with no direct or inherited references, added in the previous step, are deleted. Inner nodes with inherited relations are not deleted because it is better if the user has a more detailed hierarchy to select terms for browsing. This is useful when the user is not an expert in the themes of the metadata collection: if he does not understand the meaning of some of the more specific terms used in the lower levels of the hierarchy, he can select one of its more general ancestors to retrieve the associated information. Algorithm 9 shows a general view of the generation of the topic map process.

The labeled thesaurus is to all intents and purposes a topic map adjusted to the metadata collection that provides the resources associated to each term of the used terminological model. The topic map is generated in SKOS format, extending it to include the direct and indirect relationships. The SKOS file can then be used in a search service, to use it as topic map to navigate through the data, as part of a keyword selection system, or in a retrieval system to provide information about how many results are going to be returned when a keyword is

```
Procedure keywordExtractionandLabeling(List XMLMetadataList, ThesaurusCatalog thesCatalog);
begin
    List triplets = new ArrayList();
    for int i=0;i<XMLMetadataList.size();i++ do
        Metadata metadat = XMLMetadataList.get(i));
        tripletList.addCollection(metadat.getControlledTags();
    end
    Thesaurus thes=thesCatalog.get(tripletList.get(0).getThesaurusID());
    for int i=0;i<tripletList.size();i++ do
        Triplet triplet = tripletList.get(i);
        Concept concept = thes.getConcept(tripl.getKewWordName());
        concept.addRelation(”directReference”,triplet.getMetadataID());
        addIndirectReferences(concept, triplet.getMetadataID();
    end
    pruneNonUsedConcepts(thes.getTopConcepts(),thes);
    saveToSKOSFormat(thes,”topicMap.xml”);
end
Procedure addIndirectReferences(Concept concept, String metadataID);
begin
    concept.addRelation(”indirectReference”,triplet.metadataID());
    List conceptList = concept.getBroaders();
    addIndirectReferences(conceptList.get(i)),metadataID);
end
Procedure pruneNonUsedConcepts(List conceptList, Thesaurus thes);
begin
    for int i=0;i<conceptList.size();i++ do
        Concept concept = conceptList.get(i);
        List narrowerList = concept.getNarrowers();
        pruneNonUsedConcepts(narrowerList,thes);
        if (!concept.hasRelation(”directReference”)) && (!concept.hasRelation(”indirectReference”)) then
            thes.remove(concept);
        end
    end
end
```

Algorithm 9: Generation of the topic map

selected. An example of the SKOS representation of an element of the topic map can be seen in Figure 5.14. The extension done to the basic SKOS format has been to add two new properties "directReference" and "inheritedReference" used to store the identifiers of the metadata records that use directly of by inheritance the keyword.

The generated file can be directly used as topic map if the system to use it is able to read it, or it can be transformed into a standard format to allow that many existent applications could directly use this topic map. Among the available formats for topic maps representation, the XTM format [192] seems to be the most adequate for its use as interchange format for the topic map generated, by its simplicity and by the number of applications able to read and visualize it.

The last step consists in the extraction of the main themes of the collection by means of analyzing the topic map generated in the previous step. This can be manually done using visualization tools that allow the load of SKOS or RDF (as Protégé) and detecting manually the most used themes, but the updates that usually are done on the metadata collections can change the main themes of the collection. Therefore, to avoid the human work, an automatic process to extract this information has been designed. The process described here is based on

```
<skos:Concept
rdf:about="http://www.eionet.eu.int/gemet/concept/2405">
     <skos:prefLabel xml:lang="es">ciencias de la tierra</skos:prefLabel>
     <skos:prefLabel xml:lang="fr">sciences de la terre</skos:prefLabel>
     <skos:definition xml:lang="en">The science that deals with the earth or any part thereof;
     includes the disciplines of geology, geography, oceanography and meteorology, among others.
     (Source: MGH)</skos:definition>
     <topicMap:directReference>digital-data-30-boundary<topicMap:directReference>
     <topicMap:directReference> digital-data-30-P-13-cells<topicMap:directReference>

     <topicMap:inheritedReference>digital-data-30-P-4-conventional<topicMap: inheritedReference>

     <skos:narrower rdf:resource="http://www.eionet.eu.int/gemet/concept/5270"/>

</skos:Concept>
```

Figure 5.14: Labeled SKOS Concept

the idea of concept density used for the disambiguation of free text in Agirre and Rigau [1]. The objective is to obtain the representative nodes of the tree that aggregate a relevant percentage of records in the metadata collection. Clustering techniques analyze different properties of data to statistically group together the most similar. Elements in a cluster share common features according to a similarity criteria [39]. Inherently, the topic maps described above can be considered as hierarchical clusters. However, our intention in this last step is to obtain a 1-dimensional cluster that summarizes the collection at a first glance. Here, the hierarchical structure of the thesaurus provides a thematic context of similarity that enables terms of the same branch to be summarized by the root node of the branch. In order to identify this 1-dimensional cluster the Formula 5.5.1 has been proposed.

$$\frac{\sum\limits_{\forall node \in Branch} numberOfRecords}{\sum\limits_{\forall recods \in Collection} numKeywords} > threshold \tag{5.5.1}$$

The criterion used to identify a representative node divides the number of records containing this node (or any of its descendants) by the number of keywords in the collection. If this value is greater than a threshold, then this node is considered as a relevant node. At present this threshold has been selected experimentally, using a range of values between 0.05 and 0.2 (see section 5.5.4.2).

5.5.3 Clustering techniques

Section 5.5.2 has described a method to generate a topic map from a metadata collection based on the keywords section of metadata. It makes profit of the terminological ontologies (e.g., classification schemes, taxonomies, thesaurus or ontology) that has been selected to pick up those terms in the keywords section, enabling the creation of hierarchical topic maps thanks to the semantic relations existent in the selected terminological model. However, in distributed systems like an SDI, where the different catalogs store thousands of metadata records, the topic map summarizing the thematic contents of a collection may still be fairly complex. In this sense, 5.5.2 already identifies the necessity of obtaining a reduced set of representative terms from the generated topic map.

Focused on the context of geographic metadata, it has been analyzed the thematic classification and structuring of resources by means of clustering techniques. The clusters obtained as output of these techniques may help in two important issues of information retrieval systems operating on a network of distributed catalogs. On the one hand, the information of the clusters (e.g., names) may serve as metadata for describing the whole collection of metadata records. On the other hand, if the user is interested in browsing (instead of searching), the output clusters provide the means for structure guided browsing.

The technique proposed here improve classic clustering techniques with the hierarchical relations that may be derived from keywords found in metadata records, whenever these keywords have been selected from well-established terminological ontologies following thesaurus model. Clustering techniques group the resources by the similarity of some properties but do not have into account the relations that can exist between the analyzed properties. This section describes how to adapt some classic techniques to make profit of the hierarchical structure of the thesaurus concepts used to fill the keyword section of metadata records. The two following techniques make use of this information to improve the results obtained by traditional hard and fuzzy clustering approaches:

Clustering keywords selected from thesauri: The keywords of the metadata records are transformed into numerical codes that maintain the hierarchical relations between thesaurus terms. These numerical codes are then grouped by means of clustering techniques.

Clustering keywords as free text: Hard and fuzzy clustering techniques are directly applied over the text found in the keywords section. Previously to the clustering, some keyword expansion techniques are used to improve the results.

5.5.3.1 Selection of algorithms for hard and fuzzy clustering

As mentioned in Steinbach et al. [191], there are several techniques of hard clustering. From them, the *K-means* family of algorithms [45, 110] has been selected for the test with hard clustering algorithms. It has been selected by its simplicity and by its general use in other areas of knowledge to find patterns in collections of data and by the availability of numerous tools to perform it. The *K-means* algorithm is based on the partitions of N records into K disjoint subsets S_j (being $j \in 1..K$) that minimize the sum of the distances of each record to the center of its cluster. The mathematical function used to measure this distance varies depending on the *K-means* implementation (Correlation functions, Spearman Rank, Kendall's

Tau…). The function that has been used here for the experiments is the Euclidean distance, which is one of the most frequently applied.

The implementation of *K-means* used in the experiment section is the proposed in equation 5.5.2. The objective of this equation is to minimize the value of J, where x_n is a vector representing the *n-th* data record and v_j is the S_j centroid, being v_j calculated with formula 5.5.3, where N_j is the number of elements contained in S_j. The algorithm starts assigning randomly the records to the K clusters. Then two steps are alternated until a stop criterion is met (number of iterations or stability of J). Firstly, the centroid is computed for each record. Secondly, every record is assigned to the cluster with the closest centroid according to the Euclidean distance.

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} |x_n - v_j|^2 \tag{5.5.2}$$

$$v_j = \frac{\sum_{n \in S_j} x_n}{N_j} \ . \tag{5.5.3}$$

The main problem of this implementation is the need to select the number of clusters to create, because it is not able to automatically adjust the number of cluster returned. Other more advanced implementations use adaptive techniques. For instance, ISODATA algorithm [12] creates or joins clusters when needed, returning a number of clusters adjusted to the collection distribution. Another improvement of the proposed techniques would be to use clustering algorithms requiring no previous knowledge on the number of clusters (e.g., Extended Star [66]) to estimate the initial number of clusters before applying the approaches presented here.

With respect to the fuzzy clustering family, it includes techniques as fuzzy *C-means* (a fuzzy variant of *K-means*) or finite mixture models (Bernoulli, Gaussian or Polynomial). They assign a metadata record to each cluster with a probability value. Usually, the cluster that has the higher probability is considered as the cluster to which the record belongs, but in doubtful situations more than a cluster can be selected. Fuzzy clustering allows measuring to what extent a metadata record belongs to a cluster, distinguishing between the records that are clearly contained in a subgroup from those that may belong to several clusters. This distinction makes possible to sort the records of each subgroup by their degree of membership and to include a record in more than a cluster with different levels of relevance.

The fuzzy algorithm selected for the experiments has been the *C-means* algorithm proposed in Torra et al. [200]. It minimizes the distance of each record to every cluster centroid, adding the probability of the record to belong to each cluster. It minimizes the function in equation 5.5.4, where $A_j(x_n)$ stands for the probability of record n to be in the cluster *j*. Additionally, this algorithm takes into account the constraints in equation 5.5.5. The exponent m is a weighting parameter used to adjust the fuzziness of the clustering algorithm. As it can be seen, the function to minimize is similar to the one used in the *K-means* algorithm. The difference in this case is that the Euclidean distance of each record to a cluster center is multiplied by the probability of being such element in that cluster.

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} (A_j(x_n))^m \left| x_n - v_j \right|^2 \ . \tag{5.5.4}$$

$$(A_j(x_i)) \in [0,1] \ \ , \ \sum_{j=1}^{K} A_j(x_i) = 1 \ for \ all \ j \ . \tag{5.5.5}$$

5.5.3.2 Thematic clustering using the thesaurus structure

To detect the main themes of the collection, this first approach encodes the keywords of the metadata records into numerical values that preserve the hierarchical relations among the thesaurus concepts. Then, these encodings are clustered. This process let us group metadata records that share similar thematic characteristics without losing the benefits provided by the hierarchical structure of the thesaurus used to fill the metadata records.

The goal here is to generate a numerical identifier for each concept of the thesaurus that describes the hierarchical relations of the concepts, that is, its position in the thesaurus). This identifier is similar to the Dewey Decimal Classification System [170]. It consists of a set of numbers where each number indicates the position of the term in the thesaurus branch to which it belongs (a branch is a tree whose root is a top concept with no broader concepts and contains all the descendants of this concept in the "broader/narrower" hierarchy). For example, as shown in figure 5.15, the identifier *"02 03 00"* indicates that the term *Geotechnology* is the third child of the second top term (*Earth Science*) of the thesaurus.

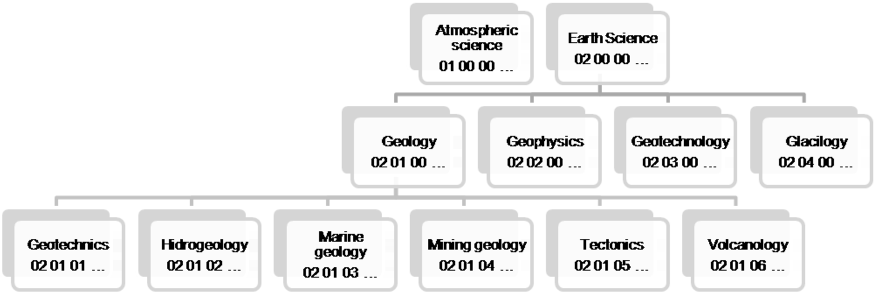


Figure 5.15: Numerical encoding of concepts

The result of clustering using these identifiers is the detection of groups of records containing concepts that are close in the thesaurus structure. These clusters are then processed to obtain a reduced set of branches that concentrate most of the keywords used in the metadata records.

Since these selected branches group most of the keywords in the metadata collection, they can be considered as its main themes. The approach consists of four sequential steps:

- As an initial step of this approach, the metadata collection has been processed to extract a list of pairs $< keywordInRecord\_URI, keyword\_id >$. In these pairs, *keywordInRecord _URI* represents a string that consists of a record identifier and a keyword term found in a record with this identifier. The second element in the pair (*keyword_id*) is the numerical identifier of the term found in the record. This list of pairs has been used as input for the hard and fuzzy clustering.

- The second step of the approach has been the application of the hard and fuzzy clustering algorithms. The result of hard clustering is a list of clusters indicating the set of $keywordInRecord_URI$s contained in each cluster. Fuzzy clustering returns a matrix where the rows represent the different *keywordInRecord_URI*s and the columns represent the identified clusters. Thus, each row in the matrix contains the membership degree of *keywordInRecord_URI* to each cluster. The cluster with the highest probability is selected as the right cluster for each $keywordInRecord\_URI$.

- Obviously, the results obtained in the second step are not the final output, because each record may have been assigned to several clusters, as many clusters as the number of times it appears in a $keywordInRecord\_URI$. Therefore, in order to detect the most proper cluster for each record, a third step has consisted in applying the following heuristic: the records have been associated to the clusters where they appear more times.

- Finally, the fourth step determines the names of the clusters. The numerical codes representing the keywords in each cluster have been processed back to obtain the terms behind them. And the common ancestors of the keywords have been analyzed to select the cluster names. The name given to each cluster is the most specific common ancestor of the keywords in the records grouped in the same cluster. To make the name more representative, there is an exception in this rule when all the records in a cluster can be grouped under a maximum number of three sub-branches. In that case the name of the cluster is the concatenation of the sub-branches names (e.g., “Product, Materials, Resource” cluster in section 5.5.4.3).

5.5.3.3 Thematic clustering using keywords as free text

The main problem in the previous approach is that it is very much dependent on taking as input metadata collections that contain terms from selected vocabularies, and that the choice for those vocabularies is relatively small. But, in practice, one can find metadata containing terms from a wide range of thesauri, or even keywords typed randomly by the users. Therefore, it seems relevant to explore other alternatives with not so restrictive prerequisites that facilitate the thematic characterization of collections described with heterogeneous metadata. This section studies the application of clustering techniques considering the keywords section as a free set of terms that do not belong necessarily to a selected vocabulary or thesaurus structure.

The keywords of the metadata records are clustered using the vector space model. In this model, documents are encoded as N-dimensional vectors where N is the number of terms in the dictionary, and each vector component reflects the relevance of the corresponding term with respect to the semantics of each document in the collection [18]. This relevance is directly proportional to the number of occurrences of a keyword in a metadata record. Additionally, not only the terms that exist in the keywords section of the metadata records are included but also all the ancestors (*broader* terms) of those terms extracted from a thesaurus. That is to say, making profit of thesaurus structure, we expand the text found in keywords sections when the terms belong to selected thesauri. Additionally, the use of ancestors helps to increase the relevance of some keywords.

Since metadata collections can be very heterogeneous, the output clusters can be seen as a set of different homogeneous sub-collections centered in very different themes. Thus, the naming of clusters may be quite problematic (the names of clusters should represent the theme of each cluster). The generation of the main theme of each sub-collection (its name) is created differently when hard or fuzzy clustering is applied. When hard clustering is used, the most frequent keywords in the output cluster are the ones selected to form part of the name. With fuzzy clustering the selected name contains the terms with the highest degree of cluster membership.

5.5.4 Testing the browsing methods

To test the viability of the process for the creation of a hierarchical topic map from the keywords of a collection of metadata and the generation of a classification with the main topics in the collection, a set of experiments have been done.

5.5.4.1 Description of the metadata corpus

As metadata corpus for experiments, the contents of the Geoscience Data Catalog at the U.S. Geological Survey[17] (USGS) were used. The USGS is the science agency for the U.S. Department of the Interior that provides information about Earth, its natural and living resources, natural hazards, and the environment. Despite being a national agency, it is also sought out by thousands of partners and customers around the world for its natural science expertise and its vast earth and biological data holdings. This metadata collection was processed as indicated in Nogueras-Iso et al. [163] until a collection of 753 metadata records compliant with CSDGM-FGDC [51] standard was obtained. The 626 keywords in the metadata collection, obtained from

[17] http://geo-nsdi.er.usgs.gov/

the GEMET thesaurus[18] have been the selected to be used in the experiment. The GEMET version used contains 5542 terms from which the collection of metadata uses 104 different ones, which is about 1.9% of the thesaurus size.

5.5.4.2 Topic map extraction

The topic map creation process has been applied to the described corpus using GEMET as base for the creation of the topic map. The topic map generated contains 216 nodes (the 104 used in the collection plus 112 inner nodes) reducing the percentage of GEMET to the 4% of its size. This huge reduction of size will provide a user a much more adjusted selection tool to locate a resource. To visualize the correctness of the topic map generated and to provide quickly to the users a tool able to navigate by the topic map and to locate the associated information, we have selected the TMNAV tool created in the TM4J project[19]. This tool allows the visualization of topic maps stored in XTM format [192] providing a graphical visualization of the properties of the records and the browsing by the relations. A branch of the generated topic map is shown in Figure 5.16 as example. In this example is shown the node Atmosphere (air, climate) of the topic map, it can be seen its relations with other terms of the topic map, such as climate and the two metadata records of the collection that indirectly contain this term of the topic map (One of their children has some occurrences in the metadata records of the collection). The example shown has not direct relations (metadata records that directly contain the term of the topic map) but if they were, they would have been also shown in the visual representation.

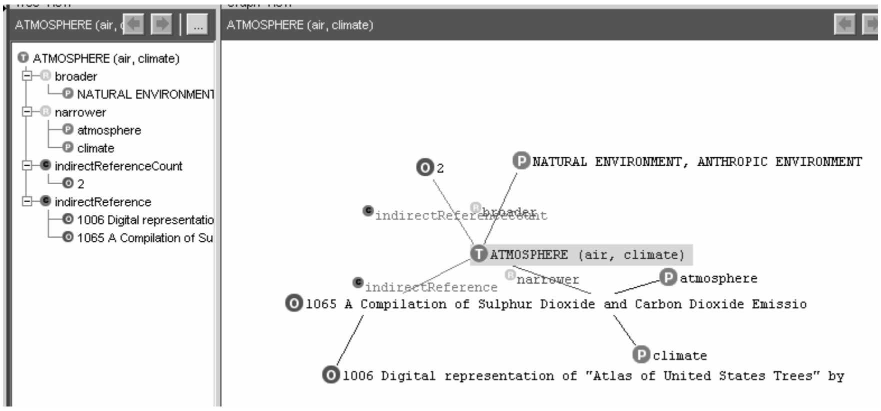

ATMOSPHERE (air, c
...
ATMOSPHERE (air, climate)
ATMOSPHERE (air, climate)
broader
NATURAL ENVIRONMENT
narrower
atmosphere
climate
indirectReferenceCount
2
indirectReference
1006 Digital representatio
1065 A Compilation of Su
2
NATURAL ENVIRONMENT, ANTHROPIC ENVIRONMENT
indirectReferenceCount
broader
ATMOSPHERE (air, climate)
indirectReference
narrower
atmosphere
1065 A Compilation of Sulphur Dioxide and Carbon Dioxide Emissio
climate
1006 Digital representation of "Atlas of United States Trees" by

Figure 5.16: Structure of a branch of the topic map

When the topic map is presented to the user in a graphical view, at first sight he can see how many results is going to obtain if he selects a term from the hierarchy (results that contain the selected keyword or one of its descendent). Then, the user can refine the selection navigating

[18] http://www.eionet.europa.eu/gemet

[19] http://tm4j.org/

for the tree, until it finds the most adequate term to query. This avoid him the execution of several queries that do not restrict the collection enough and produce too many results or select terms of a thesaurus not used in the metadata records and produce zero results. Once the topic map was generated, the next step of the experiment has been to extract automatically the main theme of the collection. The algorithm for the selection of relevant nodes (shown in previous section) was applied to the topic map using as threshold the values of 0.05, 0.1 and 0.2. With these parameters the thematic classifications of table 5.1 has been obtained.

Threshold = 0.05	**Threshold = 0.1**	**Threshold = 0.2**
human activities and products, effects on the environment	human activities and products, effects on the environment chemistry	human activities and products, effects on the environment
chemistry, substances, processes	chemistry, substances, processes	—
products, materials	—	—
product	—	—
fuel	—	—
fossil fuel	—	—
coal	—	—
material	material	—
raw material	raw material	—
natural gas	natural gas	—
social aspects, environmental policy measures	social aspects, environmental policy measures	social aspects, environmental policy measures
research, sciences	research, sciences	research, sciences
science	science	science
natural science	natural science	natural science
earth science	earth science	earth science
geology	geology	—
marine geology	—	—
natural environment, anthropic environment	—	—

Table 5.1: Threshold effect for the extraction of representative nodes

Table 5.1 shows the keywords selected as main themes of the metadata collection given a threshold. From the terms selected during the generation process, only those that has no descendants in the extracted terms are marked in bold face as main keywords, the rest of the selected terms are their hierarchical ancestors in the topic map. This example shows that the main themes of the metadata collection are "earth science" and "human activities and products, effects on the environment" terms, because they have been obtained with the highest threshold. It is also shown that the secondary themes are "geology" (child of "earth science") and "natural gas", both obtained if the clustering threshold is reduced to 0.1. The terms "coal", "marine geology" and "natural environment, anthropic environment" are less relevant, because they have been obtained with the lowest threshold.

5.5.4.3 Application of clustering techniques

The different clustering techniques described previously have been applied with the following results.

Thematic clustering using the thesaurus structure The first family of techniques described in 5.5.3 that have been tested are those that use the thesaurus structure for the clustering. The process has been validated with both hard *K-means* and fuzzy *C-means* clustering techniques to discover the difference between using them.

The collection selected for the experiment contains keywords with terms picked up from the GEMET thesaurus. This thesaurus has been processed to generate identifiers in the form already described in section 5.5.3.2. Because none of the branches of this thesaurus has more than 99 terms as *narrower* of a concept, two digits were enough to encode each level of the thesaurus.

In the case of hard *K-means* algorithm, five clusters (K=5) were asked, with the objective of obtaining the five most used branches of GEMET in the collection. However, the results obtained were disappointing because the algorithm did not converge, producing clusters with heterogeneous keywords.

The results obtained with fuzzy *C-means* algorithm were quite better. The clusters produced contained groups of records with similar keywords. The clusters obtained as output are the following:

Hydrosphere, Land: 19 records about the following topics (including hierarchical path):

- Natural Environment, Anthropic Environment. Hydrosphere
- Natural Environment, Anthropic Environment. Land

Biosphere: 10 records about the following topic (including hierarchical path):

- Natural Environment, Atrophic Environment. Biosphere

Product, Materials, Resource: 211 records about the following topics (including hierarchical path):

- Human activities and products, effects on the environment. Products, Materials. Materials
- Human activities and products, effects on the environment. Products, Materials. Product
- Human activities and products, effects on the environment. Resource

Chemistry, Substances and processes: 39 records about the following topics (including hierarchical path):

- Human activities and products, effects on the environment. Chemistry, Substances and processes

Research Science: 291 records about the following topic (including hierarchical path):

- Social Aspects, Environmental, politics measures. Research Science

These results exhibit how the metadata records of the collection are related, showing a broad thematic view of the collection. The use of upper level branches (the most generic) indicates disperse keywords in the metadata records, the use of lower level branches (more specific) expose a high relation with a specific theme and the lack of a branch of the thesaurus indicates that the metadata in the collection are not related to that theme.

Thematic clustering using keywords as free text In the second approach of section 5.5.3 the keywords of the metadata are considered as free text keywords, where some relations can be deduced from terminological ontologies. The use of hard and fuzzy clustering algorithms has produced quite different results with respect to the previous ones.

Keywords as free text and hard clustering The keywords of the USGS metadata records were transformed into a $C(NxM)$ matrix where N is the number of metadata records to cluster, and M the set of keywords contained in the collection plus the terms added from the GEMET thesaurus. An element $C(i,j)$ of the matrix takes value 1 when the term j is contained in the metadata record i or j is an ancestor in GEMET hierarchy of a term in record i. Otherwise, $C(i,j)$ takes value 0. The *K-means* algorithm were applied to the matrix asking for five clusters ($K = 5$). Then, the clusters obtained were processed to generate a representative name (combination of the most frequent keywords, or the name of the thesaurus branch that contains them), producing the following results:

Natural gas, Geology, Earth Science: This cluster consists of 151 metadata records all of them containing the keywords *Natural gas*, *Geology* and *Earth Science.*

Coal: 116 of its 117 metadata records contain the *Coal* keyword, and the other has *Lignite*, a keyword related hierarchically with *Coal* (its father).

Geology: This third cluster contains 128 metadata records from which 92 contain *Geology* and 34 *Marine geology* (narrower term of *Geology*). The remaining two records contain keywords with terms related to *Geology.* One of them contains *Earth Science* (broader term of *Geology*) and the other one contains *Mineralogy* (sibling term of *Geology*, i.e. having *Earth Science* as broader term).

Chemistry, substances and processes: This cluster contains 53 metadata records with keywords from the *Chemistry, substances, and processes* branch of GEMET.

Parameter & Others: This cluster of 32 metadata records is heterogeneous. It contains 14 metadata records from the *Parameter* branch of GEMET, however, the other 18 metadata records have no relation with them or among them.

Using the result obtained, the generated classification from the whole collection of metadata would be *Geology* (contained in 151+92 records), *Natural gas*, *Earth Science*, *Coal* and *Chemistry, substances and processes*. Each one being less relevant than the previous one (it appears fewer times).

The obtained results also show that inside the set of metadata records about *Geology* an important part of them is more specialized (they are also about *Natural gas* and *Earth Science*)

The expansion through the GEMET hierarchy in conjunction with clustering techniques allows detecting semantic aggregations not directly visible. For example, the cluster *Chemistry, substances, processes* is not detected when no hierarchy relations are considered. In addition, this technique also separates in different clusters records not sharing enough keywords. An example of this occurs in the *Geology* cluster. There are records that contain the terms *Geology* or *Earth Science*, also present in the *Natural gas* cluster, but do not contain the *Natural gas* term. In order to distinguish them from the set of records that always include *Natural gas* two different clusters have been created.

This separation causes that the two clusters share part of name, given that both contain the *Geology* term in most of their metadata. This can cause confusion when accessing the information, since two subsets are said to be about the same matter. When this happens, the solution adopted is to include all the elements of the more specific cluster inside the most general considering the *Natural gas, Geology, Earth Science* as a subset of the *Geology* cluster.

Another problem found in the obtained results is that the *K-means* algorithm assigns very heterogeneous records to the smallest cluster (in the example, the *Parameter* cluster), because they cannot be classified in the rest of clusters. The result is that the smallest cluster is quite useless. Therefore, the name obtained for the *Parameter* cluster is not adequate as it contains many heterogeneous elements. In order to remark this heterogeneity, the generic term *Others* has been added to the cluster name.

Keywords as free text and fuzzy clustering The same $C(NxM)$ matrix generated for the previous example was used with the fuzzy *C-means* algorithm asking for five clusters (K=*5*) with m=*2*. But due to the low degree of cluster membership of the records in two of the clusters, the experiment was redone with $K = 4$. In the results obtained, these two clusters were joined producing a new cluster whose main keyword was *parameter* with a probability of 0.5. There were no changes in the other clusters. The following results were obtained; they include the cluster name and the number of records that have been assigned to each cluster with the highest probability:

Natural gas, Earth Science (151 records) : The occurrences of *Natural gas* and *Earth science* keywords are contained in metadata records near the cluster center, with a mean probability of being in the cluster of 0.99, *Geology* is the following with a probability of 0.64.

Coal (117 records) : The occurrences of *coal* keyword are contained in metadata records near the cluster center, with a mean probability of being in the cluster of 0.8.

Marine geology (128 records) : The occurrences of *marine geology* are contained in metadata records near the cluster center, with a mean probability of being in the cluster of 0.75.

Others (85 records) : The other two clusters are not adequate because they contain more or less the same keywords but with low probability of belonging to the clusters.

In order to generate the name of each cluster (its main themes), the concept names present in the records with the highest cluster membership were selected. Depending on the requirements of the systems, concepts in records with a lower degree could be also included (e.g. *Geology* in the first cluster). In this situation, the membership degree could be displayed to indicate that not all the categories represent the collection in the same way.

5.5.4.4 Comparison of results

The results obtained in each experiment of section 5.5.4 are quite different. Figure 5.17 displays an skeleton of the GEMET hierarchy and how the different collection classifications obtained by each approach are located in this thesaurus hierarchy.

The main topics extracted from the topic map generated in the experiments using the process described in section 5.5.2 are the set of concepts that aggregate the keywords in the metadata collection. Regarding the first clustering approach, it has been shown that hard clustering does not work, and that fuzzy clustering produces very general results. In the second approach, fuzzy clustering produces more specific results than hard clustering. However, all the results are valid. Each one provides a vision from a different perspective of the collection: ranging from the more general vision of the first approach to the more specific vision of the last experiment.

The first clustering approach only works correctly if the keyword section has been created with terms of a single thesaurus. But, since, in practice, metadata contain terms from a wide range of thesauri, or even keywords randomly typed by the users, the second approach is more flexible. This second approach expands the keywords contained in the metadata (to improve the results) using the thesauri used as source. Nevertheless, it also works properly with several thesauri or even with keywords not contained in a controlled vocabulary.

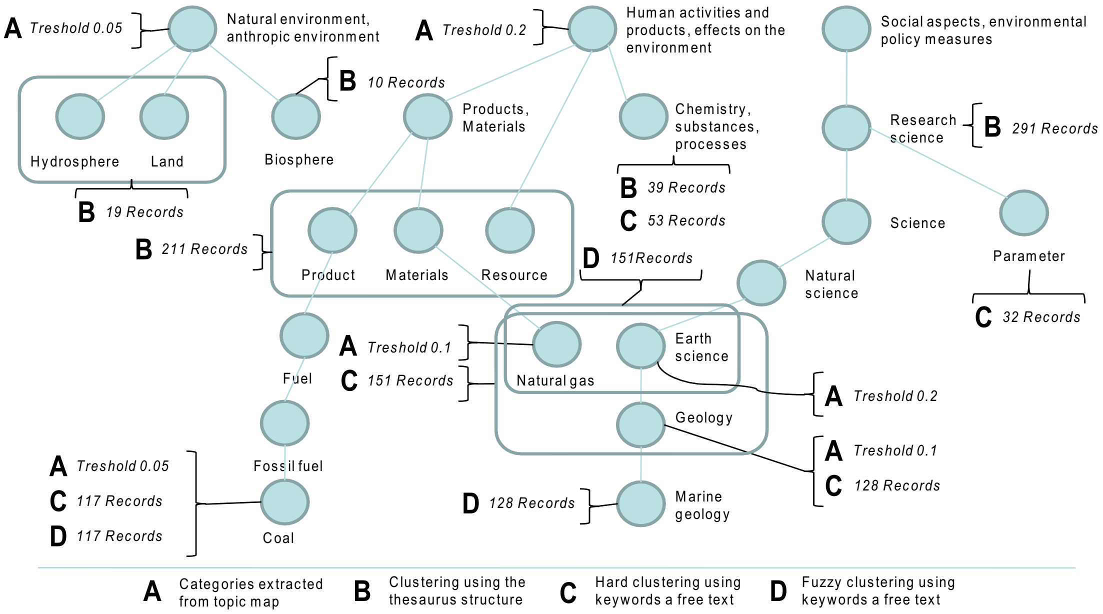


Figure 5.17: Clusters generated with the different approaches proposed

In general, hard clustering has the advantage of being more simple, but fuzzy clustering enables the identification of records with doubtful allocation (they have similar probability for two or more clusters). This is a clear advantage of fuzzy with respect to hard clustering. Fuzzy clustering detects a cluster not only on the basis of the number of times a keyword appears in a metadata record, but also taking into account that a keyword cannot be found in other clusters. An example of this is the treatment of the *Geology* concept in the second approach. When hard clustering is applied, *Geology* was as important for the cluster theme as the rest of the main keywords. However, with fuzzy clustering it is shown that the records containing *Geology* are less centered in the cluster than the other ones.

The main keywords extracted from topic map approach are quite similar to the obtained using clustering. The difference lies on the aggregation of sets of keywords. While the main keywords generated in the topic map approach only mean that they are frequently used in the collection, the clustering approaches generate set of records sharing a group of keywords (see figure 5.17); therefore, clustering techniques provides more detailed description of the thematic structure of the collection.

5.6 Conclusions

This chapter has described the main applications of terminological models for SDI discovery components. This chapter has described their integration in classification systems, classical information retrieval processes and browsing interfaces.

In the annotation area, this chapter has described the integration of the *ThManager* and Web Ontology Service technology (see chapter 4) into a metadata editor called *CatMDEdit*. In this context, the terminological ontology management components have been used to provide the metadata creator with the terminological models needed to complete those elements in the metadata records whose values belong to a controlled vocabulary. An additional functionality that has been proposed is the possibility to validate already created metadata and suggest for alternative terms contained in different terminological ontologies.

As regards the classical information retrieval context, an information retrieval model is proposed. In this model, terminological ontologies are used to expand queries by adding terms equivalents in meaning to the originally used in the query. The Web Ontology Service is used to provide the terminological ontologies used for term selection in the query construction interface, and for expansion of the query terms. Two different methods are proposed, first, a system where the expansion is based on the use of alternative concepts and translation to different languages; and a second one more elaborated in which the different terminological models on the repository are disambiguated according to an upper level ontology (WordNet) to be able to match the terms of the query from the original terminological ontology to the rest to obtain additional elements for the query. The results obtained show that the provided term expansion systems increase the recall of the user queries.

To relate the different terminological models, mappings are defined in the way indicated in section 2.4.3. Here, it is important to remark that the terms obtained by the use of mapping between ontologies should not have the same relevance that the original ones typed by the user in the query context. They are derived information extracted from mappings that usually are not perfect. Here, the concepts obtained are weighted according to the mapping liability to provide results sorted by relevance.

The last use of terminological ontologies described is related to browsing information. It is described how to extract a topic map from a collection of geographic metadata records. The method proposed assumes that the terms used in the keywords section have been selected from a well established thesaurus, and the aim of the method is to extract a hierarchical topic map that reduces significantly the size of the original thesaurus. Additionally, the method proposed also suggests a formula to obtain an even more reduced set of representative nodes (a 1-dimensional cluster), which summarizes the main themes of the collection at a first glance. Overall, it can be highlighted that the proposed topic maps facilitate enormously selection of the terms in a query system and that the generated themes (the reduced set of representative nodes) gives a synthesized and accurate summary of the contents of the metadata collection. A preliminary version of the process described here is shown in Lacasta et al. [123].

Following in the line of obtain a reduced set of representative categories for a collection for situations in which a reduced set of categories is required and the topic map size is excessive, some techniques to automatically generate a classification of a collection of metadata records using the elements of their keywords section have been proposed. These techniques are based on hard (*K-means*) and fuzzy (*C-means*) clustering and have into account the hierarchical structure of the concepts contained in the terminological ontologies used to select these keywords. Lacasta et al. [122] shows a resume of the process described here.

Two different approaches have been analyzed. In the first approach, the keywords in the metadata records picked up from a single thesaurus have been encoded as a numerical value that maintains the inner structure of the thesaurus. Then, this encoding has been used as the property for the definition of the thematic clusters. In the second approach, all the keywords in the metadata collection have been considered as free text. Additionally, in order to improve the results, when it has been recognized that a term belongs to a terminological ontology, the terms in the hierarchy of ancestors have been added to the set of keywords. Then, these sets of expanded keywords have been clustered using hard and fuzzy clustering techniques. This last approach aimed at avoiding the deficiencies of the first approach that assumed the use of a single selected vocabulary.

The generation of a model adjusted to a metadata collection provides several benefits to an SDI that can be grouped in the following categories:

- Facilitate the browsing. Ontologies provide a different way to navigate for the data in the collection. The concepts can directly have the list of metadata records that use a term, avoiding free text or even controlled queries that can easily produce empty result sets. An example of how a domain ontology in the form of a graphic topic map can be used to facilitate browsing is the Health Cyber Map project proposed in Boulos et al. [21].

- Contribute to the improvement of distributed catalog strategies. The Open Geospatial Consortium (OGC) specifies in Whiteside [217] the basic metadata that every service of a SDI should return. Between those metadata, the keywords section can indicate the theme of the data provided by the service. Those keywords can be used by a distributed catalog to redirect user queries only to the local catalogs with the appropriate themes, reducing in that way the response time of the query and the network overhead. These metadata can be created manually for the system administrator analyzing the metadata collection with visual data mining tools but with the inconvenient of having to do the same again each time there is an update. In this situation, an ontology can be used as base for the automatic creation of the keywords of the collection, facilitating their update if the collection changes.

- Facilitate the construction of queries. In metadata creation process it is usual to provide terminological ontologies in the form of controlled lists or thesauri to facilitate the creation of metadata records and to use the same structures in search systems to facilitate the location of resources. Providing directly to the user the thesauri used to create the keywords section of the metadata records sometimes is not useful because the thesauri can be too many and/or too big and, for many possible selections, the query system can produce zero results. Here, the automatic creation of an ontology where the number of associated metadata record is shown can replace the thesauri for selection of terms, given that it facilitates the user to guess if the collection contains useful data and reduces the possibility of constructing queries that produce zero results.

- Metadata quality evaluation. The identification of the main themes of a collection can be used to analyze if the thesaurus used in the metadata collection is the most adequate. For example, if it is detected that almost all the keywords of the collection are in the hydrology path of UNESCO thesaurus, possibly, to create the metadata records, instead of UNESCO, a thesaurus specialized in hydrology should have been used to allow the selection of more specialized keywords in the metadata records, to provide to the user a more complete description of the data.

It is important to note that the work described along this chapter does not stay in a simple declaration of intentions. To test their applicability, the systems, tools, and techniques have been integrated in the following real SDI projects:

- The Spanish SDI [20] project. It has as objective the integration of the data, metadata, services and information of geographical nature that are produced in Spain.

- SDIGER project[21] [221, chap. 6], [129]. SDIGER is a pilot project on the implementation of the Infrastructure for Spatial Information in Europe (INSPIRE) to support access to geographic information resources concerned with the European Water Framework Directive.

- SDI-EBRO prototype[22]. This project aims to publish the geographical data produced by the Ebro River Basin District, where clients with different visualizations of the terminological models have been used for catalog and gazetteer search interfaces, according to different criteria such as thematic or organizational.

As previously commented, not only the terminological ontologies used along this thesis work have applicability in SDIs for information retrieval, formal ontologies have also a big area of application. Future work will focus on the applicability of these models and in their integration in the context of the SDIs in the same way as it has been done for terminological models. In this area, already some work has been done. For example, Lacasta et al. [118] and López-Pellicer et al. [138] described the process of the creation of a formal ontology to model the

[20] http://www.idee.es/

[21] http://sdiger.unizar.es/

[22] http://80.255.113.15/portalIDE-Ebro/Default.vm

administrative units of Spain, with the objective of being extensible to the models of other countries. Other work focusing on formal models is [160] that use formal ontologies as part of the construction of a Gazetteer to facilitate interoperability between the different terminologies used in its content.

Chapter 6

Conclusions and future work

A Spatial Data Infrastructure aggregates a collection of technologies, policies and institutional arrangements that facilitate the availability and access to spatial data, providing a basis for spatial data discovery, evaluation, and application for users and providers within all levels of government, the commercial sector, the non-profit sector, academia and by citizens in general. The construction of these infrastructures requires a great degree of coordination between different institutions to collect all the required information and provide it in a suitable way. In an SDI, the information retrieval components have a special relevance because they are the components that provide the first point of access to a SDI and the resources contained inside. Improving these components involves an improvement in the entire infrastructure environment, making the resource holdings easier to find and access.

Ontologies are knowledge organization systems often used within the information retrieval context to improve the performance of systems. Terminological ontologies are the models most frequently used for classification and information retrieval. Thanks to their extensive use in digital and traditional libraries, these resources cover almost every area of interest, including geographical information science.

However, existent terminological models are scarcely reused across different communities. Each organization has its own systems to interpret, use and represent them, generating an heterogeneity that hinders their reuse. The geospatial community is not an exception, but the establishment of spatial data infrastructures has raised the need of managing the different models together in a simpler and uniform way. In the SDI area, ontologies are used to solve specific problems for different SDI components without providing an integrated framework of ontology management.

This thesis has focused on defining a common representation framework for terminological ontologies, together with a set of methods, architectural patterns, and guidelines for the development of accompanying artifacts that facilitate their creation, management and access. In particular this thesis has addressed the following issues:

- **Homogeneous representation of terminological ontologies:** The ontologies used by different SDI components have to be managed in a harmonized way, using the same model and format. Due to the multidisciplinary character of SDIs and its applicability to a wide range of application domains, there is a great variety of lexical ontologies with very different levels of specificity, language coverage, formalization or size. Additionally, the need to relate different terminological models to improve semantic interoperability has created the additional need to analyze how to represent the defined mappings properly.

- **Creation and reuse of terminological ontologies:** SDI retrieval systems need different terminological models depending on their purpose and required functionality. If the required terminological models exist but they are not represented in the desired format, they must be transformed and customized to the user requirements. However, if none suitable exists, a new terminological ontology must be built reusing, if possible, existent knowledge resources.

- **Management and access to terminological ontologies:** An SDI must rely on an efficient and robust ontology management service to filter and select the most appropriate ontology for each specific context. SDIs must consider many different types of terminologies for discovery, visualization, and access; each one with their own specific characteristics. In this context, a unified management system is required to simplify the access and control of the terminologies used along the infrastructure.

- **Applicability of terminological ontologies to SDI discovery systems:** Last, this thesis has addressed the integration of terminological models in SDI components related to search and presentation of information (geo-processing/catalog services, user applications, structure of content repositories and data/service catalogs) with the objective of simplifying classification of resources and improving information retrieval. Three main areas where the terminological ontologies have applicability have been reviewed: classification, discovery and browsing of resources.

In order to provide a simple and harmonized integration of terminological models in SDI components, a common representation framework has been proposed. The identification of the most suitable representation model and interchange format required a revision of different existent ontology models described in the literature, analyzing their characteristics and structure to find the common elements that the different models share. The need to establish the context in which the terminological models fit created the need to review all the main types of ontologies and not only the terminological ones. It has been found that although each otology type has usually associated a set of representation formats used to store and interchange the models, it is also very usual to have many different ad-hoc formats created for organizations to represent their models. As result of the analysis, SKOS format [148] has been identified as the most suitable format for the representation of terminological models. SKOS has a broad coverage and it is simple to define extensions to adapt it to specific user requirements. In addition, as it is shown by Lacasta et al. [121], SKOS format is very suitable for their integration in the discovery components of an SDI, and it starting to be used for organizations and companies that publish relevant terminological models in the SDI context.

Additionally, the proposed representation framework also provides support for representing mappings between terminological ontologies. Nowadays, there is a need to relate the ontology models of the information infrastructures to improve their functionality. It has been required to integrate the different terminological models to be able to jump from the terminology used in one system to the terminology used in the other, but also to perform transformations between models. This need has been found to be quite common. The use of many different vocabularies to classify information by different content creators has created the need of relating them to provide a single access point to different data collections, which use not only ontologies with different formalism degree but also different versions of the same ontology. The representation for mappings proposed in the framework has the following characteristics:

- It is based on the thesauri model, the mapping nomenclature and definitions described in the BS-8723 [27] standard. The structure of relations allows defining *exact*, *partial* and *inexact* equivalences. With respect to composite relationships the *intersection*, *union* and *difference* have been the selected. Additionally, the model is open to facilitate the addition of new relationships if required.

- The representation format is RDF based in the same way as SKOS mapping one, but the structure of mappings relations is adapted to the nomenclature of concepts and relationships used in the BS-8723 standard, and the number of tags required to define the relationships is reduced.

The integration of many terminological models in one system has required the use of the selected representation formats (SKOS) for all of them. This has simplified the management and made the input of the processes working with the ontologies homogeneous. However, given the heterogeneity of models and formats, it has not been a straightforward process. In the best case, when suitable terminological ontologies were available for the required purpose, the format was not SKOS and had to be transformed. In the worst case, the required ontologies did not exist and had to be created ad-hoc.

With respect to the transformation of terminological models to SKOS, given that information translation processes are usually ad-hoc processes, a translation generation process has been defined to simplify the needed for the creation of new translation software. The process includes the steps to follow and techniques to apply, which can be reused and applied to each transformation. Additionally, since the destination format is always SKOS an architectural pattern providing the common elements in all the translation processes has been created to facilitate the construction of new translation tools: Some conclusions have been extracted from this process:

- Transformation approaches described in the literature lack an homogeneous description of the source, destination models and formalism in the specification of the definition of relationships between the two models. This issue has been solved by defining a representation model to store the structure of the models and the relations between them.
- Part of the software created for previous translation processes can be reused. In this context, following the defined architectural pattern, to create the software for a new translation it is only needed to define a new reader for the source format and a set of translation functions to perform the transformation between the models.
- Errors in source formats are common and have to be properly managed. Therefore, it is needed a validation step to verify that the generated model is correct according to the structure that is supposed to follow.
- The generated process is reusable and applicable to a very different set of terminological models. Around 70 different terminological models have been translated to SKOS using it. Most of them have been simple controlled vocabularies used for classification purposes, such as the international codes for languages defined in ISO-639 [84] or the controlled lists contained in ISO-19115 [87], but also some more complex models such as AGROVOC, EUROVOC, GEMET and UNESCO thesauri or the Spanish and French Administrative Units Model.

For the situation in which the required terminological model does not exist and has to be created, many ontology creation methodologies exist. However, since creating an ontology from zero is a difficult task, a method to construct a new ontology has been proposed. It uses as input the knowledge found in other ontologies partially focused on the required theme. The generated model cannot be directly used but generates a base of knowledge that can be refined to obtain the desired model.

The process is based on the mapping between the concepts of different thesauri. It uses one focused on the desired thematic as core and prune those concepts not related to the desired thematic. The domain ontology obtained has several advantages in comparison with the thesauri used as source and the thesaurus used for filtering in the following areas:

- Consensus and focus: The concepts of the resulting network have been selected by consensus thanks to the mappings among the different sources, removing those concepts that are neither common nor focused on the desired thematic.

- Relations: With respect to the relation structure, the total number of available relations is bigger than the existent ones in each of the original sources. Besides each relation has a weight that indicates its relevance.
- Multilingual support: Thanks to the combination of different sources of knowledge with multilingual support, the output network is enriched with alternative terminology in different languages.

Partial versions of this approach have been applied for the generation of thesauri in the urban domain [125] and the hydrology domain [128] (as part of an information retrieval system).

In the same way that terminological ontologies can be created using other ones as base, this thesis has analyzed the feasibility of using terminological models to construct formal ones. The method focuses on the deriving of *is-a* relationships from the *broader/narrower* relationships of thesauri. The objective is to find how much the original model structure directly fit into a formal model. This prototype has been tested with the set of thematic thesauri used along the thesis (see section 2.2.1.4), and other thesauri generated in the previous merging. The results obtained have shown that the complexity of the formalization depends greatly on the structure of each processed model (it is easier to formalize the EUROVOC than the UNESCO thesaurus because it has a much higher percentage of *is-a* relations). With respect to the formalization of thesauri obtained as a result of the merging method, the results show clearly that the most common relations (those that are contained in a higher number of source models) have a much higher chance to be an *is-a* relationship. Additionally, the obtained results can be used as a measure of the structural quality of each thesaurus. They can be used as an additional factor in the decision of the thesaurus to select for each purpose.

The uniform use of terminological models in an SDI context has required the definition of a suitable management system for these models where their status and the relations with other models can be managed.

In order to facilitate the access to terminological models to other information components, the architecture of a ontology web service has been provided. The description of the ontologies using metadata facilitates their thematic location and access through search services. The models can be related with respect to a common upper lever ontology that acts as a bridge to be able to jump using a disambiguation technique. The used algorithm of disambiguation has evolved from its original versions [158, 162] to the final one referenced in this thesis. This architecture acts as an ground layer to construct on top of them the required management components:

- So as to fill the repository and have control of the cycle of life of the stored terminological ontologies a tool called *ThManager* has been designed. *ThManager* provides the functionality to create, describe, update and delete terminological ontologies stored in a repository and it is distributed as open source software. Partial versions of the design of ThManager have been published in [157] and [121].

- In order to provide the access to the ontologies contained in the repository, to the components requiring, it a centralized web service called *Web Ontology Service* (WOS) has been created. It has been designed as a centralized service where to facilitate access two different interfaces are provided: one for general access and the other one compliant with the OGC Web Services Architecture specification for its integration in an SDI. The final version of the WOS is based on the work developed in different previous research works [120, 119].

From the analysis and development of the described management components the following results have been obtained:

- The performance of the management system is suitable for the required purpose. The performance of the systems has been proved through a series of experiments on the management of a selected set of thesauri. The efficiency of the proposed storage model has been compared with respect to other model that loads the thesauri directly from a RDF file. In particular, the time spent for the graphical loading of thesauri decreases, easing the browsing of the thesaurus contents, as well as other typical operations like sorting or change of visualization language.
- The layered architecture of all the defined elements simplifies the integration of the developed components in other applications that need to use thesauri or other types of controlled vocabularies. For example, some ThManager components have been integrated within the Open Source CatMDEdit tool [223], a metadata editor tool for the documentation of geographic information resources (metadata compliant with ISO19115 geographic information metadata standard).

All the previous described elements were developed with the final objective of simplifying the integration of terminological models in SDIs, focusing especially on the discovery components. Three areas have been analyzed: annotation, discovery systems and browsing.

Within the annotation area, this thesis has studied the integration process of *ThManager* and Web Ontology Service technology into a metadata editor called *CatMDEdit*. The integration of *ThManager* management technology for terminological ontologies provides the metadata creator with the terminological models he needs to complete those elements in the metadata records whose values belong to controlled vocabularies. Additionally, it makes possible to validate already created metadata and suggest alternative terms contained in the stored terminological ontologies.

With respect to the classical information retrieval context, an information retrieval model is proposed to integrate the management of terminological model through the WOS. The WOS is used to provide the terminological ontologies used for term selection and the query construction interface. It has been used to provide terminological ontologies of catalog search systems guided by controlled vocabularies used in different SDIs such as the Spanish SDI, the SDIGER Geoportal [120], and the SDI-EBRO prototype. Additionally, the terminological ontologies are used to expand queries by adding terms equivalents in meaning to the originally used in the query. The WOS provides the required terminology and relations between models required for the defined expansion processes. The expansion is based on the use of alternative concepts and translation to different languages and it has been extended to match the query terms with respect to different terminological ontologies (functionality provided by the WOS) to obtain additional elements to expand the query.

The last use of terminological ontologies is related to browsing information. It is described how to extract a topic map from a collection of geographic metadata records. The method proposed assumes that the terms used in the keywords section have been selected from a well established thesaurus, and the aim of the method is to extract a hierarchical topic map that reduces significantly the size of the original thesaurus. Additionally, the method proposed also suggests a formula to obtain an even more reduced set of representative nodes (a 1-dimensional cluster), which summarizes the main themes of the collection at a first glance. A preliminary version of these processes has been published in [123].

Following in the line of obtaining a reduced set of representative categories for a collection, some techniques to automatically generate a classification of a collection of metadata records using the elements of their keywords section have been proposed. Lacasta et al. [122] shows a summary of the process described here. These techniques have been based on hard (*K-means*) and fuzzy (*C-means*) clustering and have into account the hierarchical structure of the concepts contained in the terminological ontologies used to select these keywords. A first approach has encoded each thesaurus concepts as a numerical value that maintains the inner structure of the thesaurus. Then, this encoding has been used as the property for the definition of the thematic clusters. This approach has shown as deficiencies the limitation of the concepts to be into a single terminological model and the impossibility to process keywords not contained in this model. In order to solve this issues, a second approach that considers the keywords in the metadata collection as free text has been considered. This approach uses the hierarchical *broader/narrower* relationships of the thesauri used for classification to add the terms in the hierarchy of ancestors to the set of keywords.

The generated topic map has proven to be applicable to facilitate the selection of the terms in a query system because it provides a synthesized and accurate summary of the contents of the metadata collection. With respect to the classifications obtained, they can be used to create metadata identifying the collection content, or as part of a general browsing interface that show the collection content according to its main topics.

Future work will continue in the research lines oriented to the improvement in representation, management and access of ontologies focusing on the following aspects:

- In addition to terminological ontologies, formal ontologies have also an important number of possible application in the SDI components, such as modeling metadata schemas or gazetteers, translation of symbology or data sharing and system development (e.g., interrelation of features from different services). To integrate them, it is needed to expand the proposed framework to take into account the use and management of formal models, analyzing where they use can provide a significant improvement of functionality. For example, SKOS could continue being the format used for terminological models, but for formal models OWL could be selected. This would not imply a complete redesign of the components architecture because both SKOS is RDF/OWL based.

- Some of the developed components and specifications are planned to be sent to standardization organizations to validate them and influence in the development of new international standards in the area. For example, it is planned to submit the specification of the second interface of the WOS as a new OGC Web Service specification that could be integrated in the future with the rest of Web Service specifications already issued by the Open Geospatial Consortium. Other element that can be used as part of a standardization process is the developed matching format that can influence in the current specification of the SKOS mapping model. With these actions it is expected, at least, to obtain the required feedback to improve, if necessary, the functionality offered by this service.

- The described techniques and process can be improved to obtain better results. The processes can be polished and the methods improved to try to obtain better results. There are many alternative techniques that can be tested in the different processes. For example, to generate the domain model used to generate a new terminological ontology based on other ones it can be taken into consideration the *grandparent* and *grandchildren* relations between thesaurus concepts to improve the calculation of the concept relationship relevance. Additionally, the semantics of the obtained relations can be enriched. The information provided by definitions, examples, and naming patterns in the properties of the original concepts should help to refine the current relations (e.g., broader relations could be refined as *part of*, *instance of* or *generalization* relations). In other different area, for example, different matching algorithms or bases of knowledge can be tested to improve the disambiguation of terminological models. Another possible improvement is to use other more sophisticated clustering approaches (instead of the classic K-Means and C-Means) to generate the classifications used for browsing.

- The integration of terminological ontologies has to be extended to other SDI components different from the associated to discovery, such as map servers, feature services or geo-coders between others; and in other SDI operational scenarios such as resource visualization or resource access and further processing. The services involved in these areas can also make use of terminological models to improve their working and the user experience. They also use controlled vocabularies to describe their capabilities and content and they can be managed in a similar way as it has been done for the discovery related components.

Bibliography

[1] Agirre E, Rigau G (1996). Word Sense Disambiguation using Conceptual Density. In *Proceedings of the 16th International Conference on Computational Linguistics (Coling'96)*, pp 16–22, Copenhagen, Denmark.

[2] Ahmed K (2000). Topic maps for repositories. In *XML Europe*, Paris, France.

[3] Alani H, Jones C, Tudhope D (2000). Associative and Spatial Relationships in Thesaurus-based Retrieval. *Lecture Notes in Computer Science. Research and Advanced Technology for Digital Libraries: 4th European Conference, ECDL.*, 1923/2000:45–55.

[4] Albertoni R, Bertone A, Demšar U, Martino MD, Hauska H (2003). Knowledge Extraction by Visual Data Mining of Metadata in Site Planning. In *Proceedings of the 9th Scandinavian Research Conference on Geographic Information Science, ScanGIS2003*, pp 119–130, Espoo, Finland.

[5] Aleksovski Z, Klein M, ten Kate1 W, van Harmelen F (2006). Matching unstructured vocabularies using a background ontology. *Lecture Notes in Computer Science*, 4248:182–197.

[6] Alfons J (2005). Reconeixement de Formes. Technical report, Universidad Politecnica de Valencia.

[7] Amann B, Fundulaki I, Scholl M (2000). Integrating ontologies and thesauri for RDF schema creation and metadata querying. *International Journal on Digital Libraries*, 3(3):221–236.

[8] ANSI/NISO (2003). Information Retrieval: Application Service Definition and Protocol Specification. Final draft for review Z39.50, Z39.50 Maintenance Agency. American National Standards Institute (ANSI). http://lcweb.loc.gov/z3950/agency/profiles/collections.html.

[9] ANSI/NISO (2005). Guidelines for the Construction, Format, and Management of Monolingual Thesauri. ANSI/NISO Z39.19-2005, American National Standards Institute (ANSI). Revision of Z39.19-1983.

[10] Antoniou G, van Harmelen F (2004). *A Semantic Web Primer*, chapter Ontology engineering, pp 205–222. Massachusetts Institute of Technology.

[11] Baeza-Yates R, Ribeiro-Neto B (1999). *Modern Information Retrieval.* New York. ACM Press, Addison Wesley.

[12] Ball G, Hall D (1965). ISODATA, A novel method of data analysis and pattern classification. NTIS AD699616, Standford Research Institute, Standford, California.

[13] Batschi WD, Felluga B, Legat R, Plini P, Stallbaumer H, Zirm KL (2002). SuperThes: A New Software for Construction, Maintenance and Visualisation of Multilingual Thesauri. In *Proceedings of the Environmental Communication in the Information Society*, Vienna.

[14] Bechhofer S, Goble C (2001). Thesaurus construction through knowledge representation. *Data & Knowledge Engineering*, 37(1):25–45.

[15] Bechhofer S, van Harmelen F, Hendler J, Horrocks I, McGuinness DL, Patel-Schneider PF, Stein LA (2004). *OWL Web Ontology Language Reference.* W3C, W3C Recommendation 10 February 2004. http://www.w3.org/TR/2004/REC-owl-ref-20040210/.

[16] Belew RK (2000). *Finding Out About.* Cambridge University Press.

[17] Bermudez L, Piasecki M (2006). Metadata Community Profiles for the Semantic Web. *Geoinformatica*, 10:159–176.

[18] Berry M, Drmac Z, Jessup E (1999). Matrices, Vector Spaces, and Information Retrieval. *SIAM Review*, 41:335362.

[19] Binding C, Tudhope D (2004). KOS at your Service: Programmatic Access to Knowledge Organisation Systems. *Journal of Digital Information*, 4 Issue 4. 26 pages.

[20] Borgida A, Brachman RJ, McGuinness DL, Resnick LA (1989). CLASSIC: A Structural Data Model for Objects. In *Proceeedings of the 1989 ACM SIGMOD International Conference on Management of Data*, pp 59–67.

[21] Boulos MNK, Roudsari AV, Carson ER (2001). Towards a Semantic Medical Web: Health-CyberMaps Dublin Core Ontology in Protégé-2000. In *Fifth International Protégé Workshop*, SCHIN, Newcastle, UK.

[22] Bouquet P, Serafini L, Zanobini S, Sceffer S (2006). Bootstrapping semantics on the web: meaning elicitation from schemas. In *Proceedings of the 15th international conference on World Wide Web table of contents*, pp 505 – 512, Edinburgh, Scotland.

[23] Bowers S, Ludäscher B (2004). An Ontology-Driven Framework for Data Transformation in Scientific Workflows. *Lecture Notes in computer Science*, 2994:1–16.

[24] Brachman RJ (1983). What IS-A Is and Isn't: An Analysis of Taxonomic Links in Semantic Networks. *Computer*, 16(10):30 – 36.

[25] British Standards Institute (1985). Guide to establishment and development of multilingual thesauri. BS 6723, British Standards Institute (BSI).

[26] British Standards Institute (1987). Guide to establishment and development of monolingual thesauri. BS 5723, British Standards Institute (BSI).

[27] British Standards Institute (2007). Structured vocabularies for information retrieval. Guide. BS 8723, British Standards Institute (BSI).

[28] Calvanese D, Giacomo GD, Lenzerini M (2001). A framework for ontology integration. In *Proceedings of the 1st Internationally Semantic Web Working Symposium (SWWS)*, Stanford, CA, USA.

[29] Chaudhri VK, Farquhar A, Fikes R, Karp PD, Rice JP (1998). Open Knowledge Base Connectivity 2.0. Technical Report KSL-98-06, Knowledge Systems Laboratory, Stanford, CA.

[30] Clark P, Thompson J, Holmback H, Duncan L (2000). Exploiting a thesaurus-based semantic net for knowledge-based search. In *Proc 12th Conf on Innovative Application of AI (AAAI/IAAI'00)*, pp 988–995.

[31] Cohen WW, Ravikumar P, Fienberg SE (2003). A comparison of string metrics for matching names and records. In *Proceedings of the KKD Workshop on Data cleaning and Object Consolidation*, pp 73 – 78, Washington (DC US).

[32] Coleman DJ, Nebert DD (1998). Building a North American Spatial Data Infrastructure. *Cartography and Geographic Information Systems*, 25(3):151–160.

[33] Compatangelo E, Meisel H (2002). Intelligent support to knowledge sharing through the articulation of class schemas. In *Proceedings of the 6th International Conference on Knowledge-Based Intelligent Information & Engineering Systems*, Crema, Itally.

[34] Cross P, Brickley D, Koch T (2001). RDF Thesaurus Specification. Technical Report 1011, Intitute for Learning and Research Technology.

[35] Cutting DR, Karger DR, Pedersen JO, W.Tukey J (1992). Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pp 318–329, Copenhagen, Denmark.

[36] d'Aquin M, Baldassarre C, Gridinoc L, Sabou M, Angeletou S, Motta E (2007). Watson: Supporting next generation semantic web applications. In *Proceedings of the WWW/Internet conference*, Vila real, Spain.

[37] Davis R, Shrobe H, Szolovits P (1993). What is a knowledge representation? *AI Magazine*, Spring:17 33.

[38] de la Beaujardiere J, (ed) (2006). *OpenGIS Web Map Server Implementation Specification. Version 1.3.0.* Open Geospatial Consortium (OGC).

[39] Demšar U (2004). A visualization of a Hierarchical Structure in Geographical metadata. In *Proceedings of the 7th AGILE Conference on Geographic Information Science*, pp 213–221. Heraklion, Greece.

[40] Denny M (2002). Ontology building: a Survey of Editing tools. *XML.com*, November:1–4. http://xml.com/pub/a/2002/11/06/ontologies.html.

[41] Dewey M (1876). *A Classification and Subject Index for Cataloguing and Arranging the Books and Pamphlets of a Library (Dewey Decimal Classification).* Project Gutenberg Literary Archive Foundation.

[42] Ding L, Finin T, Joshi A, Peng Y, Cost RS, Sachs J, Pan R, Reddivari P, Doshi V (2004). Swoogle: A semantic web search and metadata engine. In *Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management.*

[43] Doan A, Madhavan J, Domingos P, Halevy A (2002). Learning to Map between Ontologies on the Semantic Web. In *The Eleventh International WWW Conference*, Hawaii, US.

[44] Doerr M (2001). Semantic Problems of Thesaurus Mapping. *Journal of Digital Information*, 1, Issue 8(52):1–25.

[45] Dubes RC, Jain AK (1988). *Algorithms for Clustering Data.* Prentice Hall.

[46] Ehrig M (2007). *Ontology Aligment: Bridging the Semantic Gap.* Semantic Web and Beyond: Computing for Human Experience. Springer, 1 edition.

[47] European Union Commission (1994). Report on Europe and the Global Information Society: Recommendations of the High-level Group on the Information Society to the Corfu European Council. EU Commission - COM Document Supplement No. 2/94, European Union Commission. Bangemann Report.

[48] Euzenat J, Bach TL, Barrasa J, Bouquet P, Bo JD, Dieng R, Ehrig M, Hauswirth M, Jarrar M, Lara R, Maynard D, Napoli A, Stamou G, Stuckenschmidt H, Shvaiko P, Tessaris S, Acker SV, Zaihrayeu I (2004). State of the art on ontology alignment. Technical Report D2.2.3, Knowledge Web.

[49] Euzenat J, Shvaiko P (2007). *Ontology Matching.* Springer Berlin Heidelberg New York.

[50] Farquhar A, Fikes R, Rice J (1996). The Ontolingua Server: A Tool for Collaborative Ontology Construction. Technical Report KSL 96-26, Stanford University, Knowledge Systems Laboratory.

[51] Federal Geographic Data Committee (FGDC) (1998). Content Standard for Digital Geospatial Metadata, version 2.0. Document FGDC-STD-001-1998, Metadata Ad Hoc Working Group.

[52] Fellbaum C, (ed) (1998). *WordNet. An Electronic Lexical Database.* MIT Press.

[53] Fernández-Breis JT, Martínez-Béjar R (2002). A cooperative framework for integrating ontologies. *International Journal of Human-Computer Studies*, 56(6):665–720.

[54] Fikes R, Kehler T (1985). The role of frame based representation in reasoning. *Communications of ACM*, 28(9):904–920.

[55] Fisher DH (1987). Knowledge Acquisition Via Incremental Conceptual Clustering. *Machine Learning*, 2:139–172.

[56] Fisher DH (1998). *Structures and relations in knowledge organization: proc. 5th Int. ISKO Conference*, chapter From thesauri towards ontologies?, pp 18–30. Number 18-30. Würzburg: Ergon, Lille (France).

[57] Fitzke J, Atkinson R, (eds) (2006). *Gazetteer Service Profile of the Web Feature Service Implementation Specification.* Number 0.9.1 in OGC. Open Geospatial Consortium.

[58] Fonseca FT (2001). *Ontology-Driven Geographic Information Systems.* PhD thesis, The University of Maine, Orono, Maine.

[59] Fonseca FT, Egenhofer MJ, Davis CA, Borges KAV (2000). Ontologies and knowledge sharing in urban GIS. *Computers, Environment and Urban Systems*, 24:251–271.

[60] Foskett DJ (1997). *Readings in Information Retrieval*, chapter Thesaurus, pp 111–134. Morgan Kaufmann.

[61] Friedman-Hill E (2003). *Jess in Action: Rule-Based Systems in Java.* Manning Publication Co.

[62] Garshol LM (2004). Metadata? Thesauri? Taxonomies? Topic Maps!. Making sense of it all. Technical report, Ontopia.

[63] Gatius M, Bertran M, Rodríguez H (2004). Multilingual and Multimedia Information Retrieval from Web Documents. In *Proceedings of the 4th International Workshop on Natural Language and Information Systems (NLIS'04) (DEXA'04 workshop)*, Zagaroza, Spain. IEEE Computer Society.

[64] Genesereth MR, Fikes RE (1992). Knowledge Interchange Format, Version 3.0 Reference Manual. Technical Report Logic-92-1, Computer Science Department, Stanford University.

[65] Giarratano J, Riley G (1998). *Expert Systems: Principles and Programming.* PWS-Kent, Boston, MA., 3rd edition.

[66] Gil-García RJ, Badía-Contelles JM, Pons-Porrata A (2003). *Progress in Pattern Recognition, Speech and Image Analysis*, volume 2905 of *Lecture Notes in Computer Science*, chapter Extended Star Clustering Algorithm, pp 480–487. Springer.

[67] Giunchiglia F, Shvaiko P (2003). Semantic matching. *The Knowledge Engineering Review*, 18(3):265–280.

[68] Gómez-Pérez A, Fernández-López M, Corcho O (2003). *Ontological Engineering*, chapter Methodologies and Methods for Building Ontologies. Springer-Verlag, London (United Kingdom).

[69] Golbeck J, Fragoso G, Hartel F, Hendler J, Parsia B, Oberthaler J (2003). The national cancer institute's thesaurus and ontology. *Journal of Web Semantics*, 1(1):1–5.

[70] Gonzalo J, Verdejo F, Peters C, Calzolari N (1998). Applying EuroWordNet to Cross-Language Text Retrieval. *Computers and the Humanities*, Special Issue on EuroWord-Net(2-3):185 207.

[71] Gruber T (1993). A translation approach to portable ontology specifications. *ACM Knowledge Acquisition, Special issue: Current issues in knowledge modeling*, 5, Issue 2(KSL 92-71):199–220.

[72] Gruber TR (1992). Ontolingua: A mechanism to support portable ontologies. Technical Report KSL-91-66, Stanford University, Knowledge Systems Laboratory,. Revision.

[73] Guarino N (1998). Formal Ontologies and Information Systems. In Amsterdam IP, (ed), *Proceedings of FOIS'98*, pp 3–15, Trento, Italy.

[74] Guarino N, Boldrin L (1993). Ontological requirements for knowledge sharing. In *Paper presented at the IJCAI workshop for knowledge sharing and information interchange*, Chambery, France.

[75] Guarino N, Masolo C, Vetere G (1999). OntoSeek: Content-Based Access to the Web. *IEEE Intelligent Systems*, 14(3):70–80.

[76] Heath B, McArthur D, Vetter R (2005). Metadata lessons from the iLumina digital library. *Communications of the ACM*, 48(7):68–74.

[77] Heery R, Johnston P, Beckett D, Rogers N (2005). JISC metadata schema registry. In *5th ACM/IEEE-CS joint conference on Digital libraries*, page 381.

[78] Hepp M, de Bruijn J (2007). Gentax: A generic methodology for deriving owl and rdf-s ontologies from hierarchical classifications, thesauri, and inconsistent taxonomies. In *LNCS, Proceedings of the 4th European Semantic Web Conference (ESWC 2007)*, volume 4519, pp 129–144, Innsbruck, Austria. Springer.

[79] Hodge G (2000). *Systems of Knowledge Organization for Digital Libraries: Beyond Traditional Authority Files.* The Digital Library Federation, Washington DC.

[80] Horrocks I, Patel-Schneider P (2003). Foundations of the semantic web: Three theses of representation in the semantic web. *Proceedings of the Twelfth International World Wide Web Conference*, 1:39 – 47.

[81] International Council on Archives (2004). International Standard Archival Authority Record for Corporate Bodies, Persons and Families. Technical Report ISAAR (CPF), International Council on Archives (ICA).

[82] International Organization for Standardization (1985). Guidelines for the establishment and development of multilingual thesauri. ISO 5964, International Organization for Standardization (ISO).

[83] International Organization for Standardization (1986). Guidelines for the establishment and development of monolingual thesauri. ISO 2788, International Organization for Standardization (ISO).

[84] International Organization for Standardization (2002). Codes for the representation of names of languages. ISO 639, International Organization for Standardization (ISO). ISO/TC 37/SC 2.

[85] International Organization for Standardization (2003a). Computer applications in terminology - terminological markup framework. ISO/DIS 16642, International Organization for Standardization (ISO).

[86] International Organization for Standardization (2003b). Geographic information – Spatial referencing by geographic identifiers. Technical report, International Organization for Standardization (ISO), ISO/TC 211.

[87] International Organization for Standardization (2003c). Geographic information - Metadata. ISO 19115:2003, International Organization for Standardization (ISO).

[88] International Organization for Standardization (2003d). Information and documentation - The Dublin Core metadata element set. ISO 15836:2003, International Organization for Standardization (ISO).

[89] International Organization for Standardization (2003e). Information technology – SGML applications – Topic Maps. ISO/IEC 13250, International Organization for Standardization (ISO).

[90] International Organization for Standardization (2004). Geographic information – Feature concept dictionaries and registers. Technical Report N 1561:19126, International Organization for Standardization (ISO), ISO/TC 211.

[91] International Organization for Standardization (2005a). Geographic information – Methodology for feature cataloguing. ISO 19110:2005, International Organization for Standardization (ISO).

[92] International Organization for Standardization (2005b). Geographic information – Procedures for item registration. Technical report, International Organization for Standardization (ISO), ISO/TC 211.

[93] International Organization for Standardization (2005c). Geographic information – Rules for application schema. ISO 19109:2005, International Organization for Standardization (ISO).

[94] International Organization for Standardization (2005d). Geographic information - Services. ISO/DIS 19119, International Organization for Standardization (ISO), ISO/TC 211.

[95] International Organization for Standardization (2007a). Geographic information – Metadata – XML schema implementation. ISO/WD 19139, International Organization for Standardization (ISO), ISO/TC 211.

[96] International Organization for Standardization (2007b). Information technology – Common Logic (CL): a framework for a family of logic-based languages. Technical report, International Organization for Standardization (ISO).

[97] International Organization for Standardization (2008a). Language resource management lexical markup framework (lmf). ISO FDIS 24613, International Organization for Standardization (ISO).

[98] International Organization for Standardization (2008b). Terminology and other language and content resources computer applications in terminology termbase exchange format specification (tbx). ISO/DIS 30042.2, International Organization for Standardization (ISO).

[99] International Terminology Working Group (1996). Guidelines for Forming Language Equivalents: A Model Based on the Art&Architecture Thesaurus. Technical report, Getty Information Institute.

[100] Jain AK, Dubes RC (1988). *Algorithms for Clustering Data.* Prentice Hall.

[101] Janée G, Frew J (2002). The ADEPT digital library architecture. In *Proceedings of the second ACM/IEEE-CS joint conference on Digital libraries*, pp 342 – 350, Portland, Oregon, USA.

[102] Janée G, Ikeda S, Hill LL (2003). The ADL Thesaurus Protocol. Technical report, Alexandria Digital Library Project.

[103] Jones CB, Alani H, Tudhope D (2001). Geographical Information Retrieval with Ontologies of Place. *Lecture Notes in Computer Science*, 2205:322–335.

[104] Kalfoglou Y, Hu B (2005). CROSI Mapping System (CMS) Results of the 2005 Ontology Alignment Contest. In *Integrating Ontologies workshop at the 3rd International Conference on Knowledge Capture*, Banff, Canada.

[105] Kalfoglou Y, Schorlemmer M (2002). Information Flow based Ontology Mapping. In *1st International Conference on Ontologies, Databases and Application of Semantics (ODBASE'02)*, Irvine, CA, USA.

[106] Kalfoglou Y, Schorlemmer M (2003a). If-map: an ontology mapping method based on information flow theory. *Journal on Data Semantics*, 1:98127.

[107] Kalfoglou Y, Schorlemmer M (2003b). Ontology Mapping: The state of the art. *The Knowledge Engineering Review*, 18(1):1–31.

[108] Kalyanpur A, Parsia B, Sirin E, Cuenca-Grau B, Hendler J (2005). Swoop: A 'Web' Ontology Editing Browser. *Web Semantics: Science, Services and Agents on the World Wide Web*, 4(2):144–153.

[109] Kang SS (2003). Keyword-based document clustering. In *Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages*, pp 132–137.

[110] Kaufman L, Rousseeuw PJ (1990). *Finding Groups in Data: an Introduction to Cluster Analysis*. John Wiley & Sons.

[111] Kawtrakul A, Imsombut A, Thunkijjanukit A, Soergel D, Liang A, Sini M, Johannsen G, Keizer J (2005). Automatic Term Relationship Cleaning and Refinement for AGROVOC. In *Workshop on The Sixth Agricultural Ontology Service*, Vila Real, Portugal.

[112] Kim HL, Kim HG, Park KM (2004). Ontalk: ontology-based personal document management system. In *Proceedings of the 13th international World Wide Web conference*, pp 420 – 421.

[113] Klein M, Fensel D (2001). Ontology versioning for the Semantic Web. In *International Semantic Web Working Symposium (SWWS)*.

[114] Koch T, Neuroth H, Day M (2001). *Subject Retrieval in a Networked Environment: Papers Presented at an IFLA Satellite Meeting*, chapter Renardus: cross-browsing european subject gateways via a common classification system (DDC), pp 1–8. IFLA Section on Classification and Indexing & IFLA Section on Information Technology.

[115] Kotis K, Vouros G (2004). The HCONE Approach to Ontology Merging. *Lecture Notes in Computer Science*, 3053:137–151.

[116] Krowne A, Halbert M (2004). An Evaluation of Clustering and Automatic Classification For Digital Library Browse Ontologies. Metacombine project report, htttp://metacombine.org.

[117] Kuhn W (2005). Geospatial Semantics: Why, of What, and How? *Journal on Data Semantics III, Special Issue on Semantic-based Geographical Information Systems, Lecture Notes in Computer Science*, 3534:1–24.

[118] Lacasta J, López-Pellicer FJ, Floristán-Jusué J, Nogueras-Iso J, Zarazaga-Soria FJ (2006a). *Avances en las Infraestructuras de Datos Espaciales*, chapter Unidades administrativas, una perspectiva ontológica, pp 85–94. Universitat Jaume I, Castellón (España).

[119] Lacasta J, Muro-Medrano PR, Nogueras-Iso J, Zarazaga-Soria FJ (2005). Web ontology service, a key component of a spatial data infrastructure. In *Proceedings of the 11th EC GI & GIS Workshop, ESDI Setting the Framework.* 10 Pages.

[120] Lacasta J, Nogueras-Iso J, Béjar R, Muro-Medrano PR, Zarazaga-Soria FJ (2007a). A Web Ontology Service to facilitate interoperability within a Spatial Data Infrastructure: applicability to discovery. *Data & Knowledge Engineering*, 63(3):947–971.

[121] Lacasta J, Nogueras-Iso J, López-Pellicer FJ, Muro-Medrano PR, Zarazaga-Soria FJJ (2007b). ThManager: An Open Source Tool for creating and visualizing SKOS. *Information Technology and Libraries (ITAL)*, 26(3):39–51.

[122] Lacasta J, Nogueras-Iso J, Muro-Medrano PR, Zarazaga-Soria FJ (2007c). Thematic clustering of geographic resource metadata collections. *Lecture Notes in Computer Science (LNCS), 7th International Symposium on Web and Wireless GIS (W2GIS 2007)*, 4857:30–43.

[123] Lacasta J, Nogueras-Iso J, Tolosana-Calasanz R, López-Pellicer FJ, Zarazaga-Soria FJ (2006b). Automating the Thematic Characterization of Geographic Resource Collections by Means of Topic Maps. In *Proceedings of the 9th AGILE International Conference on Geographic Information Science*, pp 81–89. Visegrád, Hungary.

[124] Lacasta J, Nogueras-Iso J, Torres MP, Zarazaga-Soria FJ (2003). Towards the geographic metadata standard interoperability. In *Proceedings of AGILE 2003: 6th AGILE Conference on Geographic Information Science*, pp 555–565.

[125] Lacasta J, Nogueras-Iso J, Zarazaga-Soria FJ, Muro-Medrano PR (2008). *Conceptual Models for Urban Practitioners.*, chapter Generating an urban domain ontology through the merging of cross-domain lexical ontologies, pp 69–84. Società Editrice Esculapio, Bologna.

[126] Lacher MS, Groh G (2001). Facilitating the exchange of explicit knowledge through ontology mappings. In *Proceedings of the 14th International FLAIRS Conference*, Key West FL, USA.

[127] Lassila O, MacGuinness D (2001). The Role of Frame-Based Representations on the Semantic Web. Technical Report KSL-01-02, Knowledge Systems Laboratory, Standford University, Standford, California.

[128] Latre MA, Lacasta J, Mojica E, Nogueras-Iso J, Zarazaga-Soria FJ (2009). An approach to facilitate the integration of hydrological data by means of ontologies and multilingual thesauri. In *Lecture Notes in Geoinformation and Cartography (12th AGILE International Conference on Geographic Information Science - Advances in GIScience).* In press.

[129] Latre MA, Zarazaga-Soria FJ, Nogueras-Iso J, Béjar R, Muro-Medrano PR (2005). SDIGER: A cross-border inter-administration SDI to support WFD information access for Adour-Garonne and Ebro River Basins. In *Proceedings of the 11th EC GI & GIS Workshop, ESDI Setting the Framework*, Alguero, Italy.

[130] Lauser B, Sini M, Salokhe G, Keizer J, Katz S (2006). Agrovoc Web Services: Improved, real-time access to an agricultural thesaurus. *Quarterly Bulletin of the International Association of Agricultural Information Specialists (IAALD)*, 1019-9926(2):79–81.

[131] Lenat DB (1995). CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38.

[132] Lenat DB, Guha RV (1991). The evolution of CycL, the Cyc representation language. *ACM SIGART Bulletin, Special issue on implemented knowledge representation and reasoning systems*, 2(3):84 – 87.

[133] Lesk M (1997). *Practical Digital Libraries.* Morgan Kaufmann, San Francisco.

[134] Lieberman J, (ed) (2003). *OpenGIS Web Services Architecture, v0.3.* Number 0.3 in OGC. Open Geospatial Consortium.

[135] Lim EP, Srivastava J, Prabhakar S, Richardson J (1993). Entity identification in database integration. In *Procceedings of the 9th International Conference on DAta Engineering (ICDE).*

[136] Lindberg D, Humphreys B, McCray A (1998). The unified medical language system. *Journal of the American Medical Informatics Association*, 32(4):281 291.

[137] Longley PA, Goodchild MF, Maguire DJ, Rhind DW (2001). *Geographic information systems and science.* Willey press.

[138] López-Pellicer FJ, Florczyk AJ, Lacasta J, Zarazaga-Soria FJ, Muro-Medrano PR (2008). Administrative units, an ontological perspective. *Lecture Notes in Computer Science: Advances in Conceptual Modeling Challenges and Opportunities*, 5232:354–363. 2nd International Workshop on Semantic and Conceptual Issues in Geographic Information Systems (SeCoGIS 2008).

[139] Lutz M, Klien E (2006). Ontology-Based Retrieval of Geographic Information. *Journal of Geographical Information Science*, 20(3):233–260.

[140] Maedche A, Staab S (2002). Measuring similarity between ontologies. *Lecture Notes In Computer Science*, 2473:251 – 263.

[141] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A, Schneider L (2003). Wonderweb deliverable d17: The wonderweb library of foundational ontologies. Technical report, ISTC-CNR.

[142] Matthews BM, Wilson MD, Miller K, Ryssevik J (2001). Internationalising data access through LIMBER. In *Third international workshop on internationalisation of products and systems.*

[143] McGuinness DL, Fikes R, Rice J, Wilder S (2000). An environment for merging and testing large ontologies. In *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000)*, pp 12 – 15, Breckenridge, Colorado.

[144] McIlwaine IC (1998). The Universal Decimal Classification: Some factors concerning its origins, development, and influence. *Journal of the American Society for Information Science*, 48(4):331 – 339.

[145] McIlwaine IC (2000). *The Universal Decimal Classification: A guide to its use.* Number P035 in UDC Publication. UDC Publication, 3rd edition.

[146] Miles A, Brickley D, (eds) (2004). *SKOS Mapping Vocabulary Specification.* W3C. http://www.w3.org/2004/02/skos/mapping/spec/2004-11-11.html.

[147] Miles A, Brickley D, (eds) (2008). *SKOS Simple Knowledge Organization System Reference.* W3C, W3C Working Draft 10 May 2005, draft edition. http://www.w3.org/TR/2005/WD-swbp-skos-core-spec-20050510.

[148] Miles A, Matthews B, Wilson M (2005). SKOS Core: Simple Knowledge organization for the WEB. In *Proceedings of the International Conference on Dublin Core and Metadata Applications*, pp 5–13, Madrid, Spain.

[149] Miles A, Rogers N, Beckett D (2004). Migrating thesauri to the semantic web - guidelines and case studies for generating rdf encodings of existing thesauri. Technical Report Deliverable 8.8, SWAD-Europe.

[150] Minsky M (1981). *Mind design. Philosophy, Psychology, and Artificial Intelligence*, chapter A framework for representing knowledge, pp 95–128. MIT Press, Cambridge MA.

[151] Mizoguchi R, Vanwelkenhuysen J, Ikeda M (1995). *Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing*, chapter Task Ontology for Reuse of Problem Solving Knowledge, pp 46–59. IOS Press.

[152] Nebert D, (ed) (2004). *Developing Spatial Data Infrastructures: The SDI Cookbook v.2.0.* Global Spatial Data Infrastructure (GSDI), http://www.gsdi.org.

[153] Nebert D, Whiteside A, Vretanos P, (eds) (2007). *OpenGIS Catalogue Services Specification.* Number 2.0.2 in OGC. Open Geospatial Consortium.

[154] Network development and Marc Standard Office (2006a). Marc 21 Concise format for Authority Data. MARC 21, Library of Congress.

[155] Network development and Marc Standard Office (2006b). Marc 21 Concise format for Bibliographic Data. MARC 21, Library of Congress.

[156] Niles I, Pease A (2001). Towards a standard upper ontology. In *Proceedings of the international conference on Formal Ontology in Information Systems*, pp 2 – 9, Ogunquit, Maine, USA.

[157] Nogueras-Iso J, Bañares JA, Lacasta J, Zarazaga-Soria FJ (2003). A software tool for thesauri management, browsing and supporting advanced searches. In *Geodaten- und Geodienste-Infrastrukturen - von der Forschung zur praktischen Anwendung. Beiträge zu den Münsteraner GI-Tagen 26./27. Juni 2003*, volume 18, pp 105–118, Münster, Germany. IFGIprints.

[158] Nogueras-Iso J, Lacasta J, Bañares JA, Muro-Medrano PR, Zarazaga-Soria FJ (2004a). Exploiting disambiguated thesauri for information retrieval in metadata catalogs. *Lecture Notes on Artificial Intelligence (LNAI)*, 3040:322–333.

[159] Nogueras-Iso J, Latre MA, Muro-Medrano PR, Zarazaga-Soria FJ (2004b). Building eGovernment services over Spatial Data Infrastructures. *Lecture Notes in Computer Science (LNCS)*, 3183:387–391.

[160] Nogueras-Iso J, López-Pellicer FJ, Lacasta J, Zarazaga-Soria FJ, Muro-Medrano PR (2007). *Ontologies for Urban Development: Interfacing Urban Information Systems*, volume 61 of *Studies in Computational Intelligence*, chapter Building an Address Gazetteer on top of an Urban Network Ontology, pp 157–167. Springer.

[161] Nogueras-Iso J, Zarazaga-Soria FJ, Lacasta J, Béjar R, Muro-Medrano PR (2004c). Metadata Standard Interoperability: Application in the Geographic Information Domain. *Computers, Environment and Urban Systems*, 28(6):611–634.

[162] Nogueras-Iso J, Zarazaga-Soria FJ, Lacasta J, Tolosana-Calasanz R, Muro-Medrano PR (2004d). Improving multilingual catalog search services by means of multilingual thesaurus disambiguation. In *Proceedings of the 10th European Commission GI&GIS Workshop, ESDI: The State of the Art*, Warsaw, Poland. 14 pages.

[163] Nogueras-Iso J, Zarazaga-Soria FJ, Muro-Medrano PR (2005). *Geographic Information Metadata for Spatial Data Infrastructures - Resources, Interoperability and Information Retrieval.* Springer Verlag.

[164] Noy N, (ed) (2005). *Representing Classes As Property Values on the Semantic Web.* W3C.

[165] Noy NF, Fergerson RW, Musen MA (2000). *Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management*, volume 1937 of *Lecture Notes In Computer Science*, chapter The knowledge model of Protégé-2000: Combining interoperability and flexibility, pp 17–32. Springer-Verlag, Juan-les-Pins, France.

[166] Noy NF, Musen MA (1999). SMART: Automated Support for Ontology Merging and Alignment. In *Twelth Workshop on Knowledge Acquisition, Modeling, and Management*, Banff, Canada.

[167] Noy NF, Musen MA (2000). PROMPT: Algorithm and tool for automated ontology merging and alignment. In *Proceedings of the 17th NAtional Conference on Artificial Inteligence*, pp 450–455.

[168] Oard D (1998). A comparative study of query and document translation for cross-language information retrieval. In *Proceedings of the Third Conference of the Association for machine Translation in the Americas*, page 472483.

[169] O'Looney JA (2000). *Beyond Maps: GIS Decision Making in Local Government.* ESRI Press, Redlands, California.

[170] Online Computer Library Center (2003). *Dewey Decimal Classification System, 22nd edition.* Online Computer Library Center (OCLC).

[171] Ostländer N, Tegtmeyer S, Foerster T (2005). Developing an SDI for time-variant and multi-lingual information dissemination and data distribution. In *Proceedings of the 11th EC GI&GIS Workshop, ESDI: Setting the Framework*, Alghero, Italy.

[172] Palma R, Haase P, In AGP (2006). Oyster: sharing and re-using ontologies in a peer-to-peer community. In ACM Press, New York N, (ed), *Proceedings of the 15th International Conference on World Wide Web*, pp 1009–1010, Edinburgh, Scotland.

[173] PD CEN/TR 15449 (2006). Geographic information. Standards, specifications, technical reports and guidelines, required to implement spatial data infrastructure. CEN/TC 287, European Comiteee for Standarization.

[174] Podolak I, Demšar U (2004). Discovering structure in geographical metadata. In *Proceedings of the 12th conference in Geoinformatics*, pp 1–7, Galve, Sweden.

[175] Prasad S, Peng Y, Finin T (2002). Using explicit information to map between two ontologies. In *Proceedings of the AAMAS Workshop on Ontologies in Agent Systems*, Bologne, Italy.

[176] Pundt H, Bishr Y (2002). Domain ontologies for data sharing. An example from environmental monitoring using field GIS. *Computer & Geosciences*, 28(1):95 – 102.

[177] Rahm E, Bernstein PA (2001). A survey of approaches to automatic schema matching. *The VLDB Journal The International Journal on Very Large Data Bases archive*, 10(4):334 – 350.

[178] Rahm E, Do HH, Maßmann S (2004). Matching large xml schemas. *ACM SIGMOD Record archive*, 33(4):26 – 31.

[179] Ranganathan SR (1962). *Elements of library classification.* Asia Publishing House, Bombay.

[180] Resnik P (1995). Disambiguating noun groupings with respect to WordNet senses. In *Proc. of the 3rd Workshop on Very Large Corpora.* MIT.

[181] Roussey C (2005). Guidelines to build ontologies : A bibliographic study. Technical report nr. 1, COST Action C21. http://www.towntology.net/Documents/guidelines.pdf.

[182] Schaerf A (1994). *Query answering in Concept-Based Knowledge Representation Systems: Algorithms, Complexity and Semantic Issues.* PhD thesis, Dipartimento di Informatica e Sistemistica. Università di Roma 'La Sapienza'.

[183] Schlieder C, Vögele T (2002). Indexing and Browsing Digital Maps with Intelligent Thumbnails. In *Spatial Data Handling 2002 (SDH'02)*, Ottawa, Canada. 12 pages.

[184] Schlieder C, Vögele T, Visser U (2001). Qualitative Spatial Representation for Information Retrieval by Gazetteers. In *Proceedings of Conference of Spatial Information Theory COSIT*, volume 2205, pp 336–351, Morrow Bay, CA.

[185] Shafiq O, Toma I, Krummenacher R, Strang T, Fensel D (2006). Using Triple Space computing for communication and coordination in Semantic Grid. In *Proceedings of the 3rd Semantic Grid Workshop in conj. with the 16th Global Grid Forum*, pp 13–16, Athens, Greece.

[186] Sigel A (2006). From traditional Knowledge Organization Systems (authority files, classifications, thesauri) towards ontologies on the web. In *Workshop Introducing Terminology-based Ontologies at the 9th International Conference of the International Society for Knowledge Organization (ISKO)*, pp 3–53, Vienna, Austria. Published electronically on E-LIS (E-prints in Library and Information Science, http://eprints.rclis.org), 2006-07-14.

[187] Smits P, (ed) (2003). *Inspire Architecture and Standards.* JRC Institute for Environment and sustainability.

[188] Soergel D, Lauser B, Liang A, Fisseha F, Keizer J, Katz S (2004). Reengineering Thesauri for New Applications: the AGROVOC Example. *Journal of Digital Information*, 4(4):1–19.

[189] Soualmia L, Goldbreich C, Darmoni S (2004). Representing the mesh in owl: Towards a semi-automatic migration. In *Proceedings of the 1st Intl Workshop on Formal Biomedical Knowledge Representation (KR-MED 2004)*, page 8187, Whistler, Canada.

[190] Sowa JF (1996). Ontologies for Knowledge Sharing. In *Manuscript of the invited talk at Terminology and Knowledge Engineering Congress (TKE '96)*, Vienna.

[191] Steinbach M, Karypis G, Kumar V (2000). A comparison of document clustering techniques. In *Proceedings of the KDD Workshop on Text Mining*, pp 1–20, Boston, USA.

[192] Steve Pepper and Graham Moore (eds.) (2001). XML Topic Maps (XTM) 1.0. Technical report, http://www.topicmaps.org.

[193] Stumme G, Maedche A (2001). Ontology Merging for Federated Ontologies on the Semantic Web. In *Proceedings of the International Workshop for Foundations of Models for Information Integration (FMII-2001)*, Viterbo, Italy.

[194] Sure Y, Angele J, Staab S (2002). *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, volume 2519/2002 of *Lecture Notes in Computer Science*, chapter OntoEdit: Guiding Ontology Development by Methodology and Inferencing, pp 1205–1222. Springer Berlin / Heidelberg.

[195] Sussna M (1993). Word sense disambiguation for free-text indexing using a massive semantic network. In *Proc. of the Second International Conference on Information and Knowledge Management (CIKM-93)*, Arlington, Virginia.

[196] Tennis JT (2005). SKOS and the Ontogenesis of Vocabularies. In *Dublin Core Conferece: Vocabularies in Practice.*

[197] Tolosana-Calasanz R, Alvarez-Robles JA, Lacasta J, Nogueras-Iso J, Muro-Medrano PR, Zarazaga-Soria FJ (2006a). On the problem of identifying the quality of geographic metadata. *Lecture Notes in Computer Science (LNCS), Research and Advanced Technology for Digital Libraries, ECDL 2006*, 4172:232–243.

[198] Tolosana-Calasanz R, Nogueras-Iso J, Béjar R, Muro-Medrano PR, Zarazaga-Soria FJ (2006b). Semantic interoperability based on Dublin Core hierarchical one-to-one mappings. *International Journal of Metadata, Semantics and Ontologies*, 1(3):183–188.

[199] Tolosana-Calasanz R, Portolés-Rodrígez D, Nogueras-Iso J, Muro-Medrano PR, Zarazaga-Soria FJ (2005). CatServer: A Server of GATOS. In *Proceedings of AGILE 2005: 8th Conference on Geographic Information Science*, pp 359–366.

[200] Torra V, Miyamoto S, Lanau S (2005). Exploration of textual document archives using a fuzzy hierarchical clustering algorithm in the GAMBAL system. *Information Processing and Management*, 41:587–598.

[201] Tudhope D, Alani H, , Jones C (2005). Augmenting Thesaurus Relationships: Possibilities for Retrieval. *Journal of Digital Information*, 1, Issue 8(41, 2001-02-05). 22 pages.

[202] Tudhope D, Binding C (2005). Towards Terminology Services: experiences with a pilot web service thesaurus browser. In *Proceedings of the International Conference on Dublin Core and Metadata Aplications*.

[203] Tudhope D, Binding C, Blocks D, Cunliffe D (2006). Query expansion via conceptual distance in thesaurus indexed collections. *Journal of Documentation*, 62(4):509–533.

[204] United Nations Educational, Scientific and Cultural Organization (UNESCO) (1995). *UNESCO Thesaurus: A Structured List of Descriptors for Indexing and Retrieving Literature in the Fields of Education, Science, Social and Human Science, Culture, Communication and Information*. UNESCO Publishing, Paris. http://www.ulcc.ac.uk/unesco/.

[205] U.S. Congress (1991). *High Performance Computing and Communications Act of 1991*. Superintendent of Documents, Congressional Sales Office, U.S. Government Printing Office, Washington, DC 20402.

[206] U.S. Federal Register (1994). Executive Order 12906. Coordinating Geographic Data Acquisition and Access: the National Spatial Data Infrastructure (U.S.). *The April 13,1994, Edition of the Federal Register*, 59(71):17671–17674.

[207] Usländer T (2005). Trends of environmental information systems in the context of the European Water Framework directive. *Environmental Modelling & Software*, 20(12):1532–1542.

[208] van Assem M, Malaisé V, Miles A, Schreiber G (2006). A Method to Convert Thesauri to SKOS. In *Proceedings of the 3rd European Semantic Web Conference (ESWC-06)*, pp 95–109, Budva, Montenegro.

[209] van Assem M, Menken MR, Schreiber G, Wielemaker J, Wielinga B (2004). A method for converting thesauri to RDF/OWL. In McIlraith SA, Plexousakis D, van Harmelen F, (eds), *Proceedings of the Third International Semantic Web Conference (ISWC 2004)*, Hiroshima, Japan. Springer.

[210] van Heijst G, Schreiber AT, Wielinga BJ (1997). Using explicit ontologies in KBS development. *International Journal of Human-Computer Studies*, 46(2-3):183 – 292.

[211] Visser U, Stuckenschmidt H, Schuster G, Vögele T (2002). Ontologies for geographic information processing. *Computers & Geosciences*, 28(1):103 – 117.

[212] Volz R, Oberle D, Motik B, Staab S (2003). KAON SERVER - A Semantic Web Management System. In *12th World Wide Web, Alternate Tracks - Practice and Experience*, Hungary, Budapest.

[213] Vossen P (1998). Introduction to EuroWordNet. *Computers and the Humanities (Special Issue on EuroWordNet)*, 32(2-3):73–89.

[214] Vretanos PA, (ed) (2005). *Web Feature Service Implementation Specification. Version 1.1.0.* Open Geospatial Consortium (OGC).

[215] Vretanos(Eds) P (2005). Filter Encoding Implementation Specification, Version 1.1. OpenGIS project document OGC 04-095, OpenGIS Consortium Inc.

[216] Weißenberg N, Gartmann R (2003). Ontology Architecture for Semantic GeoServices for Olympia 2008. In *Proceedings of the Geodaten- und Geodienste-Infrastrukturen - von der Forschung zur praktischen Anwendung. Beiträge zu den Münsteraner GI-Tagen*, volume 18, pp 267–283, Münster, Germany. IFGIprints.

[217] Whiteside A, (ed) (2007). *OGC Web Services Common Specification. Version 1.1.0.* Open Geospatial Consortium (OGC).

[218] Whiteside A, Evans JD, (eds) (2006). *Web Coverage Service (WCS) Implementation Specification. Version 1.1.0.* Open Geospatial Consortium (OGC).

[219] Wielemaker J, Schreiber G, Wielinga1 B (2005). Using Triples for Implementation: The Triple20 Ontology-Manipulation Tool. *Lecture Notes in Computer Science (LNCS)*, 3729:773–785.

[220] Wielinga BJ, Schreiber AT, Wielemaker J, Sandberg JAC (2001). From Thesaurus to Ontology. In *Proceedings of the 1st international conference on Knowledge capture*, pp 194 – 201, Victoria, British Columbia, Canada.

[221] Zarazaga-Soria F, Nogueras-Iso J, Latre M, Rodríguez A, López E, Vivas P, Muro-Medrano P (2007). *Research and Theory in Advancing Spatial Data Infrastructure Concepts*, chapter Providing SDI Services in a Cross-Border Scenario: the SDIGER Project Use Case, pp 107–119. ESRI Press.

[222] Zarazaga-Soria F, Torres M, Nogueras-Iso J, Lacasta J, Cantán O (2003a). Integrating geographic and non-geographic data search services using metadata crosswalks. In *Proceedings of the 9th EC-GI&GIS Workshop: ESDI: Serving the User.* 12 pages.

[223] Zarazaga-Soria FJ, Lacasta J, Nogueras-Iso J, Torres MP, Muro-Medrano PR (2003b). A Java Tool for Creating ISO/FGDC Geographic Metadata. In *Geodaten - und Geodienste-Infrastrukturen - von der Forschung zur praktischen Anwendung (Beiträge zu den Münsteraner GI-Tagen)*, volume 18 of *IFGIprints*, pp 17–30, Münster, Germany.