UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

GUILLERMO NUDELMAN HESS

Towards Effective Geographic Ontology Semantic Similarity Assessment

Thesis presented in partial fulfillment
of the requirements for the degree of
Doctor of Computer Science

Prof. Dr. Cirano Iochpe
Advisor

Prof. Dr. Silvana Castano
Coadvisor

Porto Alegre, December 2008

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Hess, Guillermo Nudelman

Towards Effective Geographic Ontology Semantic Similarity Assessment / Guillermo Nudelman Hess. – Porto Alegre: PPGC da UFRGS, 2008.

118 f.: il.

Thesis (Ph.D.) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2008. Advisor: Cirano Iochpe; Coadvisor: Silvana Castano.

1. Geographic ontologies. 2. Semantic matching. 3. Similarity measurement. I. Iochpe, Cirano. II. Castano, Silvana. III. Título.

ACKNOWLEDGEMENTS

There are many people that somehow contributed directly or indirectly to the development of this work. Firstly I would like to thank the three most important people in my life: Renata for the love, companionship and patience , my mother Marion for everything she gave to me (love, support, values) and my sister Carolina, who is my best friend and always encouraged me although geographically distant. I would like to thank also my aunt Monica, my grandmother Margot, and Rosanne, Ricardo, Dora and Israel, who are also an important part of the family.

A special thank to my advisor prof. Cirano Iochpe for the years of partnership and my co-advisor prof. Silvana Castano for the opportunity she gave me to spend one year of my Ph.D. in her lab at the Universita degli Studi di Milano, in Italy. Furthermore, I would like to thank all the professors that contributed with ideas, critics and suggestions to this work, especially prof. José Palazzo M. de Oliveira, and all the people from UFRGS and UNIMI who create an excellent working environment.

I would also like to thank to my lab colleagues and friends, here in Brazil, such as Gleison, Carolina, Mariusa, Giselli, Daniel and Gabriel, and in Italy, such as Stefano Montanelli, Stefano Bruno and Alfio Ferrara. An special thank to Lucineia Thom and Gianpaolo Messa for the affection and friendship.

Finally, a special thank to the CAPES agency, for the 4-year Ph.D. scholarship here in Brazil and the 1-year sandwich program scholarship in Italy.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS AND ACRONYMS

CKB	Common Knowledge Base
DL	Description Logics
GDB	Geographic Database
GIIM	Geographic Information Integration or Mapping
GIS	Geographic Information System
NLP	Natural Language Processing
OGC	Open GIS Consortium
ORET	Ontology Reverse Engineering Technique
OWL	Ontology Web Language
RDFs	Resource Description Framework Schema
STOBJ	Spatio-temporal object
UML	Unified Modeling Language
UTM	Universe Transverse Mercator
XML	eXtensible Markup Language

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Integration of geographic information is becoming more important every day, due to the facility to exchange data through the Internet and the high cost to produce them. With the semantic web, the description of geographic information using ontologies is getting popular. To allow the integration, one of the steps in which many researches are focusing is the matching of geographic ontologies. A matching consists on measuring the similarity of the elements, namely either concepts or instances, of two (or more) given ontologies. The main problem with ontology matching is that the ontologies may be described by different communities, using different vocabularies and different perspectives. For geographic ontologies the difficulties may be even worse, for the particularities of the geographic information (geometry, location and spatial relationships) as well as due to the lack of a widely accepted geographic ontology model, and because the ontologies are usually described at different semantic granularities. The specificities of geographic ontologies make conventional matchers not suitable for matching geographic ontologies. On the other hand, the existing geographic ontology matchers are considerably limited in their functionality and deal with ontologies described in a particular perspective. To overcome the current limitations, in this work we present a number of similarity measurement expressions and algorithms to efficiently match two geographic ontologies, at both the concept and instance-level. These algorithms combine expressions used to assess the similarity of the so-called conventional features with expressions tailor made for covering the geographic particularities. Furthermore, this research also proposes a geographic ontology meta-model to serve as a basis for the development of geographic ontologies in order to standardize their description. This model is compliant with the OGC recommendations and is the basis upon which the algorithms are defined. For the evaluation of the algorithms, a software architecture called IG-MATCH was created with an additional feature of making possible to enrich the geographic ontologies with topological relationships and parent-child relationships by the analysis of the instances.

Keywords: Geographic ontologies, semantic matching, similarity measurement.

Towards Effective Geographic Ontology Semantic Similarity Assessment

RESUMO

A cada dia cresce a importância da integração de informações geográficas, em virtude da facilidade de intercambiar dados através da Internet e do alto custo de produção deste tipo de informação. Com o advento da web semântica, o uso de ontologias para descrever informações geográficas está se tornando popular. Para permitir a integração, um dos estágios no qual muitas pesquisas estão focando é o chamado *matching* das ontologias geográficas. *Matching* consiste na medida de similaridade entre os elementos de duas ou mais ontologias geográficas. Estes elementos são chamados de conceitos e instâncias. O principal problema enfrentado no *matching* de ontologias é que estas podem ser descritas por diferentes pessoas (ou grupos), utilizando vocabulários diferentes e perspectivas variadas. No caso de ontologias geográficas os problemas são ainda maiores, em razão das particularidades da informação geográfica (geometria, localização espacial e relacionamentos espaciais), em função da falta de um modelo para descrição de ontologias geográficas amplamente adotado e, também, porque as ontologias são, muitas vezes, descritas em diferentes níveis de granularidade semântica. Estas particularidades das ontologias geográficas torna os *matchers* convencionais inadequados para o *matching* de ontologias geográficas. Por outro lado, os *matchers* existentes para o domínio geográfico são bastante limitados e somente funcionam para ontologias descritas em um modelo específico. Com o objetivo de superar essas limitações, neste trabalho são apresentados algoritmos e expressões (métricas) para medir a similaridade entre duas ontologias geográficas efetivamente, tanto em nível de instâncias quanto em nível de conceitos. Os algoritmos propostos combinam métricas para medir a similaridade considerando os aspectos não geográficos dos conceitos e instâncias com expressões criadas especificamente para tratar as características geográficas. Além disto, este trabalho também propõe um modelo para ontologia geográfica genérico, que pode servir como base para a criação de ontologias geográficas de forma padronizada. Este modelo é compatível com as recomendações do OGC e é a base para os algoritmos. Para validar estes algoritmos foi criada uma arquitetura de software chamada IG-MATCH a qual apresenta também a possibilidade de enriquecer a semântica das ontologias geográficas com relacionamentos topológicos e do tipo generalização/especialização através da análise de suas instâncias.

Palavras-chave: ontologias geográficas, *matching* semântico, medida de similaridade.

1 INTRODUCTION AND MOTIVATION

Since the creation of Geographic Information Systems (GIS), new fields of research are emerging due to the peculiarities of geographic data, which is different from conventional, alphanumeric data. In fact, besides the descriptive components, namely relationships and attributes, and hierarchy, geographic data is featured by three other characteristics, namely geometry, spatial location and capability of holding spatial relationships (ARONOFF, 1991; FONSECA; DAVIS; CAMARA, 2003). Geographic data may also have the temporal component (SOTNYKOVA et al., 2005), even if this cannot be pointed as a specific feature for geographic data. Furthermore, geographic data is described using some particular metadata, which gives some important information about the data itself, such as the coordinate reference system, the projection system, the projection scale, and so on.

Spatial relations are relationships that can occur between two geographic objects, depending on both their geometries and spatial position. The spatial relations are classified into three different categories:

- **Topological**: This class of spatial relations defines the possible relationships between the geometries of the geographical objects. Table 1.1 presents Egenhofer's 9-intersection model (EGENHOFER; FRANZOSA, 1991). It defines the relationships regarding the geometries that an object may assume and is adopted in the majority of works we found dealing with topological relationships.

Table 1.1: 9-intersection model for topological relationships (EGENHOFER; FRANZOSA, 1991)

Relationships	**Applicable geometries**
Touches	A/A, L/L, L/A, P/A, P/L
Crosses	L/L, L/A, P/A, P/L
Inside (Within)	A/A, L/L, L/A, P/A, P/L
Overlaps	A/A, L/L, P/P
Contains	A/A, L/L, A/L, A/P, L/P
Disjoint	A/A, L/L, L/A, P/A, P/L, P/P
Intersects	A/A, L/L, A/L, A/P, L/P, P/P
Equal	A/A, L/L, P/P

where A, L and P represent the geometries. A means area (polygon), L means line and P means point.

- **Order(directional)**: The order relationships do not depend on the geometry of the associated geographic objects. They depend on the relative spatial position when comparing one to another. There are twelve possible directional relationships (FRANK, 1992):
  - At_north_of: A geographic object A is said to be at_north_of a geographic object B if the highest coordinate y of A is bigger than the highest coordinate y of B.
  - At_south_of: A geographic object A is said to be at_south_of a geographic object B if the lowest coordinate y of A is smaller than the lowest coordinate y of B.
  - At_east_of: A geographic object A is said to be at_east_of a geographic object B if the highest coordinate x of A is bigger than the highest coordinate x of B.
  - At_west_of: A geographic object A is said to be at_west_of a geographic object B if the lowest coordinate x of A is smaller than the lowest coordinate x of B.
  - At_northeast_of: A geographic object A is said to be at_northeast_of a geographic object B if the highest pair of coordinates (x,y) of A is bigger than the highest pair of coordinates (x,y) of B for both x and y.
  - At_northwest_of: A geographic object A is said to be at_northwest_of a geographic object B there is a pair of coordinates (x,y) of A which has an y higher than the highest y of B and an x lower than the lowest x on B.
  - At_southeast_of: A geographic object A is said to be at_southeast_of a geographic object B there is a pair of coordinates (x,y) of A which has an y lower than the lowest y of B and an x bigger than the highest x on B.
  - At_southwest_of: A geographic object A is said to be at_southwest_of a geographic object B if the lowest pair of coordinates (x,y) of A is smaller than the lowest pair of coordinates (x,y) of B for both x and y.
  - Above: A geographic object A is said to be above a geographic object B if the lowest coordinate y of A is higher than the highest coordinate y of B.
  - Below: A geographic object A is said to be below a geographic object B if the highest coordinate y of A is lower than the lowest coordinate y of B.
  - At_right_of: A geographic object A is said to be at_right_of a geographic object B if the lowest coordinate x of A is higher than the highest coordinate x of B.
  - At_left_of: A geographic object A is said to be at_left_of a geographic object B if the highest coordinate x of A is lower than the lowest coordinate x of B.
- **Metrics (distance)**: These relationships are usually measured by the GIS, and are not explicitly stored or modeled.

For each one of the possible spatial representations of a geographic concept, the following metadata may be associated:

- capture and update time and, if possible, the period in which that spatial representation is valid;

- coordinate system, projection and scale, if them exist;
- information about the data capturing system: source (satellite photo, image, aerial photo) and additional information about the capturing equipment (satellite, camera, flight, etc.);
- geometry storage format: raster or vectorial.

Actually, the metadata may vary more from instance to instance than from concept to concept. Therefore, it has influence only in the similarity measurement among instances. If two instances being compared are described using different metadata, probably the values of the properties which are influenced by the metadata would be different. For example, suppose we have two instances named *Milan*, each one belonging to one ontology and, furthermore, the concepts they instantiate were already identified as equivalent. It may happen that one of the instances is described using the <*latitude, longitude*> reference system, while the other is described using the *Universal Transverse Mercator (UTM)* reference system. In this case, the values for the *hasPosition* property would be $< 45^o20'N, 9^o10'E >$ and $< 5166930.21N, 1921142.04E >$, respectively. If the metadata is ignored, a matcher would return that the two locations are not the same, while they actually are.

Actually GIS are used every day. Some examples are the Global Positioning Systems (GPS) used in cars, the Google Earth and Google Maps tool, maps generators on the web, and so on. Producing geographic data is time consuming and expensive. Furthermore, in many cases the data needed is already available in some other systems or organizations. At the same time, the diffusion of the Internet allowed the interchange of information all around the world. If, on one hand, this interchange offers a lot of benefits, such as the reuse of information and knowledge sharing, on the other hand it generates the need to deal with the heterogeneities among the information obtained from distinct geographic sources. This problem is difficult to solve due to poor documentation as well as implicit semantics of the data and diversity of data sets. With the web 2.0 - known as the semantic web - the objective is to embed semantics in the data to be interchanged, in order to allow machines to identify useful resources.

One research field emerged with the semantic web is the ontology's. An ontology is an explicit specification of a conceptualization (GRUBER, 1993). More specifically, an ontology is a logical theory that corresponds to the intentional meaning of a formal vocabulary, i.e., an ontological commitment with a specific conceptualization of the world (GUARINO, 1997). An ontology consists in logical axioms that contain the meaning of the terms in a specific community. The logical axioms represent the concepts hierarchies and relationships among them. An ontology is specific for a community and must be accepted as an agreement of the community's members (BISHR et al., 1999).

Ontologies are being widely used for storing and interchanging information over the web, because they can act just as databases, but with associated semantics and requiring much less storage space. An ontology is composed by concepts organized in a taxonomy, properties, axioms and instances. The concepts describe the elements to be represented and the properties represent their characteristics, such as attributes and relationships. The axioms are the taxonomic relationships and restrictions for the properties and the instances are the individuals belonging to the concepts. When interchanging ontologies, one of the challenges is to identify similar or equivalent structures (concepts or instances). This process is called matching and, for conventional ontologies, there are already good proposals,

such as (CASTANO; FERRARA; MONTANELLI, 2006; GIUNCHIGLIA; SHVAIKO; YATSKEVICH, 2005; NOY, 2004). These matchers basically work at the concept-level and, unfortunately, are not capable of addressing the particular features of geographic information.

The scenario above encouraged the present research, in which we propose one solution for part of the problem of geographic information integration. Our focus is on matching. For this purpose we developed a matching technique which works at the concept-level and at the instance-level as well, by means of measuring the similarity of the concepts belonging to the ontologies and also the similarity between their instances (data). The technique we propose consists in a number of expressions (metrics) and algorithms, covering both the conventional features, i.e., non-geographic, as well as the specific characteristics geographic ontologies have. Some of the metrics are adapted from the non-geographic field and some of them are specially tailored for the geographic features. Furthermore, we introduce an ontology model to describe ontologies in a general, Open GIS Consortium (OGC) compliant way, which is the basis for the matching algorithms.

1.1 Objective and contributions

The objective of this thesis is the definition of algorithms and mathematical expressions (metrics) for assessing the similarity of geographic ontologies, at the concept-level and the instance-level as well, using as basis a geographic ontology model which is semantically rich enough to describe any kind of static, i.e., non-temporal, ontology. This geographic ontology model is also part of this work.

The development of the algorithms, metrics and of the geographic ontology model resulted in a number of contributions, as follows:

- **Geographic ontology model**
  In the geographic information systems field there is still missing a standard and widely accepted model for geographic ontologies. This leads to ontologies defined in different levels of semantic granularity and with conceptual incompatibilities. One example is if the ontology must or must not define geometries for the geographic concepts and how to describe the spatial component of a geographic concept. This creates a need for a model to translate the ontologies into prior to matching them.

  The first contribution of this research is, therefore, the proposal of a geographic ontology model, specific for matching purposes. It consists of an ontology with features (concepts and properties) specific for static geographic information, i.e., non-temporal. It can be seen as a *framework* to guide the construction of static geographic ontologies, by defining the concepts, properties and axioms needed to represent geographic information. Based on the elements of the geographic ontology model we can formally define the conflicts, also known as heterogeneities, that may occur when comparing two geographic ontologies, at both the concept and instance-levels.

- **Algorithm and metrics for matching geographic concepts**
  Based on the heterogeneities that may happen when comparing two geographic ontologies we studied the existing proposals for matching, integrating and mapping them, and, as discussed in Chapter 3, none of them is complete. Therefore, the

second contribution of this research is the definition of an algorithm to match geographic ontologies at the concept-level. This algorithm combines features and mathematical expressions (metrics) for assessing the similarity of conventional ontologies with some other tailor made metrics for the geographic concepts specificities.

- **Algorithm and metrics for matching geographic instances**
  As important as, or even more important than the matching of geographic concepts is the matching of geographic instances (data). The third contribution of this research is the proposal of an algorithm to assess the similarity of two geographic ontologies at the instance-level. The proposed algorithm is not limited to match the instances according to the concepts they belong to and to their spatial positions (coordinates). The metrics developed for this algorithm take into account the spatial characteristics of data and also the alphanumeric ones. Furthermore, also some metadata are considered in the similarity assessment process. This is of special importance when dealing with geographic data because if the metadata of the compared instances are different and this fact is ignored, certainly wrong results would be produced. Finally, the concept of geographic context region is introduced, in order to accelerate the matchmaking process.

- **Algorithm for inferring spatial and hierarchical relationships from ontology's instances**
  Sometimes the instances of a geographic ontology carry implicit information that cannot be gathered by analyzing the concepts they belong to. Less common, but also possible, is the occurrence of an ontology with only instances (data), i.e., without the explicit definition of the concepts. Although the existing ontology management tools do not support this, conceptually it is possible to have the concepts defined in a separate file from the one the instances are described.

  The fourth contribution of this research is an algorithm that perform the ontology's reverse engineering. Given the instances, it can rebuild concepts with the properties associated to their context and, furthermore, can infer spatial relationships (directional and topological) from the analysis of the instance's spatial location. Implicit hierarchical relationships, such as sibling concepts can be discovered as well.

- IG-MATCH **software architecture**
  As a side effect of the contributions above a software architecture was developed. Its main objective was to put all together the concept matcher algorithm, instance matcher algorithm and ontology enrichment algorithm and evaluate them with tests.

1.2 Outline

The reminder of this dissertation is organized in seven chapters. In Chapter 2 we present the theoretical background and formal definitions used as the basis of this research. We briefly discuss the existing proposals for a geographic ontology model and present our proposal for a geographic ontology model, with the specific purpose of geographic ontology matching. Based on the features covered by the geographic ontology model, we formally define the possible heterogeneities that may be found when comparing two geographic ontologies. We address the features at concept-level and instance-level as well.

Related work in semantic matching, mapping, alignment and integration of geographic information (ontologies, databases, conceptual schemas) are discussed in Chapter 3. Besides simply presenting the state-of-the-art in this field, we elaborate a set of criteria to perform a deep comparison of the existing proposals, with the goal of analyzing which features are already addressed and which features are still neglected, as well as studying how to combine these features to obtain better results when matching geographic ontologies, at both the concept and the instance-level.

In Chapters 4 and 5 we respectively detail the concept-level and the instance-level algorithms and mathematical expressions for matching geographic ontologies. Both chapters are organized in the same way, by presenting the matching algorithms and after the mathematical expressions (metrics). Finally, we report some test results. In Chapter 5 we also introduce the notion of a geographic context region as an artifact to accelerate the matching process.

In Chapter 6 we present the developed technique to (re)build an ontology (concepts, properties and taxonomy) from the instances. The algorithm for the ontology (re)construction is depicted, presenting the method for inferring topological relations that may hold between the geographic concepts.

In Chapter 7 we present the software architecture we created for evaluating the algorithms and metrics developed as the main contributions of the work, called IG-MATCH. It details the concept and instance-levels as well as the concept wrapper architectures.

Finally, in Chapter 8 we draw some conclusions from the elaboration of this dissertation. The overall results obtained with the research are discussed, highlighting the contributions of the work. The open issues and possible future works are discussed as well.

2 FORMAL DEFINITIONS

Due to the particularities of GIS data - geometry and location (FONSECA; DAVIS; CAMARA, 2003), and, eventually, temporal properties as well (SOTNYKOVA; CULLOT; VANGENOT, 2005), besides the usual descriptive attributes - a simple alphanumeric ontology (hereafter called conventional ontology) is not expressive enough to describe the geographic domain. The ability to build proper geographic ontologies will facilitate their integration and, subsequently, will advance semantic interoperability, which has been acknowledged as a primary concern in geographic information science nowadays (TOMAI; KAVOURAS, 2004).

Maedche and Staab (2000) state that an ontology should comprise the following: (a) Concepts, (b) the Lexicon, (c) Relations and (d) Axioms. Concepts are an integral part of an ontology as they stand for mental representation of all possible things (TOMAI; KAVOURAS, 2004). The Lexicon comprises the descriptions of the concepts, i.e., their definition in natural language. The semantic relations link pairs of concepts in hypernym/hyponym relations and in the meronym/holonym relations as well. The relation as semantic properties refer to the properties of the concepts in the ontology. The axioms refer to constraints imposed on concepts or relations.

Although ontologies are being widely used by the GIS community, there is still a lack for an actual spatio-temporal ontology. That is, the ontologies proposed and used at the moment are designed for conventional (descriptive), non-spatial purposes and the particularities of the geographic data, such as the geometry, temporality and topological relations are missed or poorly described. There are already some standard proposals (e.g., ISO 19109 and GML OWL encoding), but they focus more on the syntaxis than on the semantics of the concepts being described.

2.1 Existing geographic ontology models

Tomai and Kavouras (2004) extend Maedche and Staab's (MAEDCHE; STAAB, 2000) definition of ontology by defining the components of a geographic ontology. They basically proposed some semantic properties to be associated to a concept when it represents a geographic concept: Spatiality, Temporality, Nature, Material/cover, Purpose and Activity. The first two are the ones that actually characterize a geographic ontology. Spatiality covers the relative spatial properties of the concept, such as topology, location, and the internal spatial properties, such as size and shape. Temporality is divided into time (period or instant) and condition/status. In the comparison Table 2.1, we refer to this work as TK.

A spatio-temporal object (STOBJ) (XU; HUANG; LIU, 2006) has spatial and temporal properties as well. The former encompass geometries and the spatial relationships such

as distance, position, topological, and so on. Temporal properties are, basically, instant and period. Based on these properties, a spatio-temporal ontology is a normative system describing spatio-temporal objects and relationships between them. In the comparison Table 2.1, we refer to Xu et al. work as XHL.

Casati, Smith and Varzi (1998) separate a geographic ontology in two parts: objects and relations. The geographic objects are specialized into physical, such as mountains, rivers and forests, and human, such as countries, cities, and so on. A geographic object is composed by a number of descriptive attributes and by a border. The relations can be of type mereology, location or topology. In a mereology association, a geographic object A *is part of* a geographic object B. The location relation associates a geographic concept with a set of coordinates, and a topology relation spatially associates two geographic concepts. In the comparison Table 2.1, we refer to this work as CSV.

Souza et al. (2006) propose an ontology to represent contextual information in geospatial data integration. The ontology is composed by 5 contexts, as the authors present. Each one of them stores some kind of information. The main two are the *DataContext* and *AssociationContext*. The *GeospatialEntity* is the main concept of the *DataContext*, and contains the properties for geometric representation, location and some metadata. The *AssociationContext* has the information about the spatial association of the concepts and the semantic associations (degree of similarity) as well. As weak point of these works we can point the absence of temporal aspects and the impossibility of representing non-geographic concepts. In the comparison Table 2.1, we refer to this work as SST.

Fu et al.(2005) developed a geo-ontology restricted to geographic places, such as cities, countries, districts and so on. Each concept is described in terms of its names (can be multiple), geometry (called footprints by the authors) and some metadata. Furthermore, each place may be related to another by only one relation, the *containment relation* (FU; JONES; ABDELMOTY, 2005). In the comparison Table 2.1, we refer to this work as FJA.

Kolas et al. (2006) propose an architecture for what they call *Geospatial Semantic Web*. They define 6 ontologies, and one of them, called *Base Geospatial Ontology* is of interest in the context of this research. It forms the ontological foundation of geospatial information by mapping some GML's elements to OWL, in order to link the geographic data with knowledge outside the geospatial realm (KOLAS; DEAN; HEBELER, 2006). In the comparison Table 2.1, we refer to this work as KDH.

SWETO-GS (ARPINAR et al., 2006) is a spatio-temporal ontology with three dimensions, namely thematic, spatial and temporal. The thematic dimension contains the concepts of a general domain such as people, places and organizations, or for a specific domain such as travel and transport. In that dimension there are both geographic and non-geographic concepts. The geospatial dimension stores the spatial data and relationships. The concepts are described in terms of their coordinates, translated from the thematic dimension. The temporal dimension stores the temporal relations that may occur between concepts. Finally, some metadata can be associated to the SWETO-GS ontology. In the comparison Table 2.1, we refer to this work as ASR.

Bittner and Smith propose an ontological theory which contains resources to describe geographic processes and the concepts that participate therein (2003). For that purpose two (sub-)ontologies are presented, one describing the concepts with their properties, called SNAP, and one describing the processes and their parts and aggregates, called SPAN (BITTNER; SMITH, 2003). SNAP entities are described in terms of their properties, spatial relations and conventional relations, while SPAN entities are described also

considering time. In the comparison Table 2.1, we refer to this work as BS.

Table 2.1 compares the existing geographic ontology models according to a set of criteria considered as necessary and sufficient to correctly describe a geographic ontology. These criteria addresses the features are the ones that have influence in the matching process, according to the works presented in Chapter 3, and are also based on what can be represented by these models. The criteria are described below:

1 **Def. of geographic concepts**: The similarity measurement between geographic concepts is the central issue when matching geographic ontologies. Therefore, it is mandatory to a geographic ontology model the support for defining geographic concepts.

2 **Definition of geometries**: According to the OGC consortium, each geographic concept must be associated to a geometry. Therefore, it is important for the geographic ontology model to describe geometries.

3 **Definition of spatial position**: Every instance of a geographic concept must have a set of coordinates, which gives its spatial position. This is the main feature addressed when matching geographic instances and, therefore, it is important for a geographic ontology the coverage of the spatial position.

4 **Concept description/annotatio**: Some geographic ontology matchers use the concept annotation, i.e., a textual description, in the matching process. Thus, the support for adding annotation in the concept definition is important for a geographic ontology.

5 **Definition of spatial relations**: Spatial relations, specifically topological and directional, are features that distinguish geographic information from conventional information, and, therefore, their definition must be supported by a geographic ontology.

6 **Def. of non-spatial properties**: Non-spatial properties, such as attributes and relationships, are characteristics of both geographic and non-geographic information which are considered in the matching process. Therefore, the geographic ontology model must support their definition.

7 **Definition of temporality**: As temporality is a feature inherent of geographic information and can be used in a matching process, it must be supported by a geographic ontology.

8 **Definition of metadata**: As the metadata play an important role in the geographic ontology matching, specially at the instance-level, it definition should be supported by a geographic ontology.

As can be inferred, none of the proposed models fully satisfies the criteria list. For matching purposes Tomai and Kavouras (2004), Casati, Smith and Varzi (1998) and Souza, Salgado and Tedesco (2006) proposals are the ones closer to fit our necessities. However, the first two proposals do not provide means of describing the ontology's metadata, which is important when matching geographic ontologies, especially at the instance-level, as discussed in a following subsection. Furthermore, the description of non-spatial properties is limited in (TOMAI; KAVOURAS, 2004; SOUZA; SALGADO; TEDESCO, 2006). For these reasons we decided to build our own geographic ontology model, on the basis of the existing ones, with the specific purpose of matching.

Table 2.1: Comparison of the existing proposals for geographic ontology models

Criterion	TK	XHL	CSV	SST	FJA	KDH	ASR	BS
Def. of geographic concepts	√	√	√	√	√	√	√	√
Definition of geometries	√	√	√	√	√	√	X	X
Definition of spatial position	√	X	√	√	√	X	√	X
Concept description/annotation	√	X	X	X	X	X	X	X
Definition of spatial relations	√	√	√	√	±	√	√	√
Def. of non-spatial properties	±	X	√	±	X	X	X	X
Definition of temporality	√	√	X	X	X	√	√	X
Definition of metadata	X	X	X	√	√	X	√	X

the √ symbol represents that the criterion is covered by the proposed model. The ± symbol represents that the criterion is partially supported, while the X symbol represents that the criterion is not supported.

2.2 A geographic ontology model

Fonseca et al. (2003) state that in order to integrate geographic ontologies or schemas, they should be mapped from the original format to an ontology. The conceptual data model proposed here should enable the representation of all feature types that are usually used to characterize both geographic concepts and geographic instances. According to Spaccapietra et al. (2004), space and time can meet ontologies in three different ways: (1) as the spatial domain, specifying space, spatial elements and spatial relationships, or as the temporal domain, specifying time, temporal elements and temporal relationships; (2) as the implicit background to an application domain that relies on geographical data or; (3) to enrich the description of the concepts in the ontology, to represent their spatial and temporal localization, in the same way spatio-temporal data models support the description of spatial and temporal features stored in spatio-temporal databases.

A geographic ontology can be further classified as either a geographic domain ontology or as a geographic application ontology. According to Fonseca and Martin (2007), a domain ontology has the goal of giving the meaning to terms through the existing, explicit relationships between concepts from a specific domain. On the other hand, application ontologies are in the same abstraction level as conceptual schemas, which are built with a specific information system in mind (FONSECA; MARTIN, 2007). Since the majority of proposals for geographic information integration or mapping ***giim***, presented in Chapter 3, takes as inputs conceptual schemas or application ontologies, or data from them, we decided to define our model to fit the geographic application ontology needs. Therefore, from now on we refer to geographic application ontology as geographic ontology. The model is evolutionary, which means that we considered the existing models presented in the previous section as the basis for our model. Especially (TOMAI; KAVOURAS, 2004; CASATI; SMITH; VARZI, 1998; SOUZA; SALGADO; TEDESCO, 2006) were considered. Moreover, we took some principles from the existing proposals of geographic conceptual models (BORGES; DAVIS; LAENDER, 2001; SOTNYKOVA et al., 2005) and frameworks (FILHO; IOCHPE, 1999) to classify the types of concepts a geographic ontology may have as well as to define the properties that must be associated to a geographic concept.

Our model for a geographic ontology is an extension of the ontology definition found

in (SCHARFFE; BRUIJN, 2005) for general purpose ontologies, which can be represented as a 4-tuple $O =< C, P, A, I >$[1], where C is the set of concepts, P is the set of properties, A is the set of axioms and I is the set of instances.

The proposed geographic ontology model, also called "geo-ontology" for short, can be defined as 5-tuple of the type $O_g =< C_g, P_g, A_g, I, M >$, where C_g, P_g, I_g are extensions or specializations of, respectively, C, P and I, and A is as in (SCHARFFE; BRUIJN, 2005). M is the set of metadata associated to the instances of geographic concepts that are represented in the ontology.

A concept $c \in C_g$ is any real world phenomenon of interest. It is defined by a triple of information: a term t that is used to identify (label) it, its textual description, and by the definition of a so called *context* to which this concept is related.

The concept identifier is given by the unary function $t(c)$. The *context* of a given concept is determined by two sets: a subset of P_g and a subset of A. Each property of P_g that is in the context of c is given by a unary function $p(c)$. Similarly, each axiom of A, representing either a generalization/specialization relation or a restriction, that applies to c is given by a unary function $x(c)$. Therefore, formally the context of a concept c can be defined as a triple $ctx(c) =< t(c), \{p(c)\}, \{x(c)\} >$

In the model we propose we specialize the definition of a concept found in (SCHARFFE; BRUIJN, 2005). Depending on its context, a concept can be classified as a *domain concept*, such as, for instance, a *River* or a *Building*, or as a *geometry concept*, such as *Point* or *Polygon*, as depicted in Figure 2.1[2].

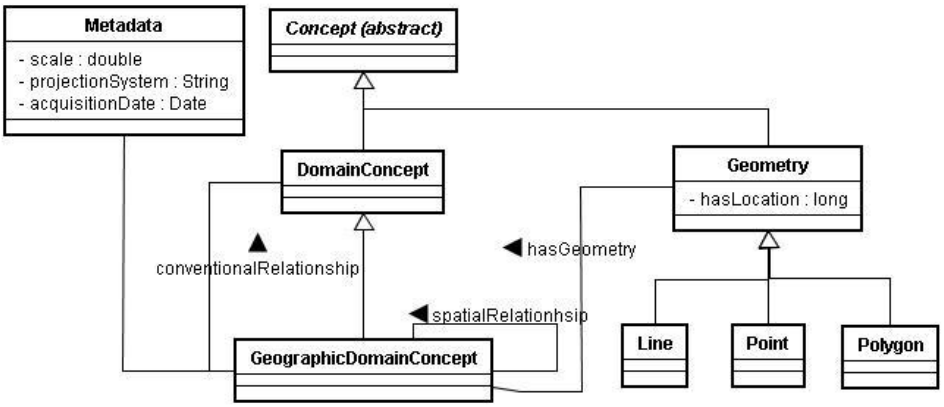

Figure 2.1: Types of concepts of the geographic ontology model

A *geographic domain concept* (gc) is a specialization of a domain concept that represents a geographic phenomenon. Besides the properties and relations of a domain concept, the definition of a geographic domain concept includes an association to, at least, one geometry concept. The geometry plays a fundamental role on defining the possible spatial relationships the concept may have. The association between a geographic domain concept and a geometry concept is OGC compliant, i.e., follows the Open GIS Consortium (OGC) (OGC, 2005) recommendation. However, this is not a consensus in the geographic ontology community (SPACCAPIETRA et al., 2004). The main argument against the

[1]This definition is based on the OKBC model (CHAUDHRI et al., 1998). In the original work, instead of P (properties) it was R (relations)

[2]Although in most of the models there is also a time class, we do not represent it in the model because it is not yet supported by our matching algorithms.

mandatory association between a geographic object and the geometry representing it is that an ontology is in an abstraction level higher than the so called representation level, where, for example, databases are designed (FONSECA; DAVIS; CAMARA, 2003).

A property $p \in P_g$ is associated to a concept c with the goal of characterizing it. The range of a property can be a data type, such as string or number, or an object type, i.e., another concept. Formally, a property can be defined as:

$$p =< t(p), pd, minCard(p), maxCard(p) >,$$

where $t(p)$ is the function which gives the property's name, pd is the property domain and $minCard(p)$ and $maxCard(p)$ are, respectively, the property's minimum and maximum cardinalities.

In the geographic ontology model, the property set P_g specializes the general ontology element P. Each property can be of one of four possible types, as depicted in the taxonomy of Figure 2.2: conventional, spatial, geometric or positional. A conventional property can be either an attribute of a domain concept (in this case, it is a data type property) or a relationship between two domain concepts, when at least one is not geographic (in this case, it is an object type property). A spatial property represents an association between two geographic domain concepts, i.e., is always an object type property. The spatial relationships have a pre-defined semantics and are standardized (EGENHOFER; FRANZOSA, 1991; FRANK, 1992), while conventional relationships may assume different semantics depending on the associated concepts. A geometric property (always an object type property) associates a geographic domain concept with a geometry concept. A positional property is a data type property that must be associated to a geometry concept, to give its location (set of coordinates).

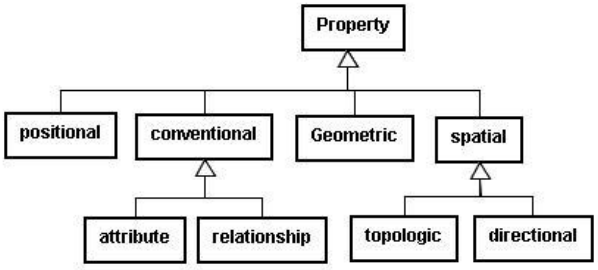

Figure 2.2: Types of properties of the geographic ontology model

In case of a conventional property, if it plays the role of an attribute, *minCard(p)* and *maxCard(p)* are not relevant. Furthermore, as the allowed values for domain are data types we can define an attribute as $a =< t(p), dtp >$, where *dtp* is the attribute's data type.

For any relationship, i.e., conventional, spatial or geometric, the property domain *pd* is another concept c_x, as follows

$$cr =< t(p), t(c_x), minCard(p), maxCard(p) >$$

.

In case of a conventional relationship *cr*, there is a restriction that c_x must be a domain concept, but not a geographic domain concept (gc), i.e., $cr = (p \in P|(c_x : \neg gc))$. In the case of a spatial relationship *sr*, both associated concepts must be geographic domain

concept, i.e., *sr* must be part of the context of a *gc* concept, and the allowed domains for c_x are other geographic domain concepts. Formally, $sr = (p \in P|(c_x : gc))$.

According to the OGC the geometry of a concept is given by the association of that geographic concept to a geometry concept, named *hasGeometry*. Therefore, a geometric property can only be associated to a geographic domain concept (gc) context. Furthermore, the associated concept c_x must be a geometric concept *geo*. Formally, $ge = (p \in P|t(p) = \text{“}hasGeometry'' \wedge (c_x : geo) \wedge minCard(p) = 1)$.

Finally, a positional property *pos* is a data type property with the restriction that its name must be *hasLocation*. Furthermore, it can only be associated to a geometry concept. Formally, $pos = (p \in P|(t(p) = \text{“}hasLocation'') \wedge (pd : dtp))$

An axiom describes either an hierarchical relationship between concepts, or provides an association between a property and a concept (through the property domain or through a concept restriction), or defines some restrictions for a property within the context of a concept. Examples are given in Figure 2.4.

To formally define a geographic domain concept *gc* and a geometry concept *geo* it is necessary to define two axioms. In the case of a geographic domain concept *gc*, the restriction says that *gc* must have in its context at least one geometric property *ge*, as follows:

$$gc = (c \in O|\exists p \in ctx(c) \wedge p : ge \wedge t(p) = \text{”}hasGeometry\text{”} \wedge minCard = 1)$$

.

A geometry concept *geo* must have, in its context, exactly one positional property *pos*. Formally, geometry concept can be defined as:

$$geo = (c \in O|\exists p \in ctx(c) \wedge p : pos \wedge minCard = 1 \wedge maxCard = 1)$$

Finally, the new element we introduce in the proposed geographic ontology model is the set of metadata M. A metadata $mdt \in M$ represent one of the possible metadata to be associated with the instances. It is defined by a unary function $t(mdt)$ which gives its label, as follows: $mdt =< t(mdt) >$

An instance $i \in I$ is a particular occurrence of a concept c, with values for each property $p(c)$ and axiom $x(c)$ in the context of c. It presents a unique identification $t(i)$. Thus, an instance in a geographic ontology may be defined as $i =< t(c), t(i), \{pv(i)\}, \{mdv(i)\} >$, where $t(c)$ is the concept being instantiated, $t(i)$ is the instance unique identifier, $pv(i)$ is the set of values for the properties and axioms belonging to the context of the instantiated concept. The value of a property within an instance is given by the binary function $vp(t(p), val)$, where *val* is the value of that property. Finally, $mdv(i)$ is the set of metadata values. The value for each metadata is given by a binary function $vmd(t(mdt), val)$. The instance of a domain concept that is not also a geographic domain concept present NULL values for the $mdv(i)$ component.

The spatial location of a geographic instance *gi* is obtained by analyzing the value for the *hasGeometry* property. It links the geographic instance to an instance of a geometry concept *geoi*. The *hasLocation* property of the *geoi* holds the coordinates values. This definition is compliant with the Open GIS Consortium.

Figures 2.2 and 2.4 present an example of an ontology defined according to the model, graphically and in a pseudo-language respectively.

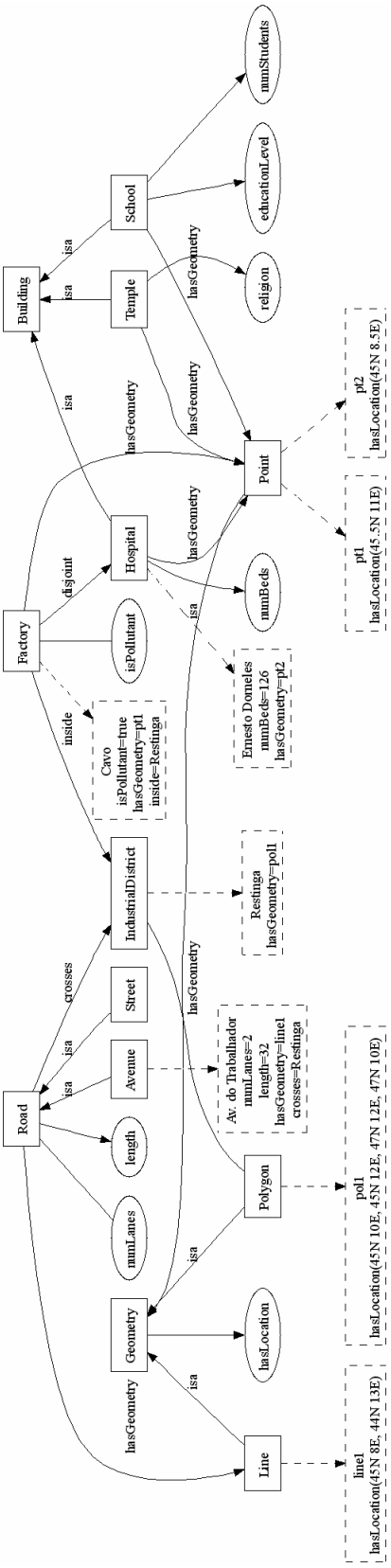

Figure 2.3: First example of geographic ontology O

The rectangles with continuous lines represent concepts, the ellipses the properties representing attributes associated with a concept and the dashed rectangles the instances belonging to a concept. The arcs linking two concepts correspond to the properties which represent relationships holding between them, while the isa labeled arrows are the taxonomic relationships (axioms) between two concepts, in which one is the specialization of the other.

C_g = $Road, Avenue, Street, Factory, IndustrialDistrict$ (geographicdomain)
$Building, Hospital, Temple, School$ (geographicdomain)
$Geometry, Line, Polygon, Point$ (geometry)

P_g = $isPollutant, numBeds, numLanes, length$ (conventional)
$religion, educationalLevel, numStudents$ (conventional)
$disjoint, crosses, inside$ (spatial)
$hasGeometry$ (geometric)
$hasLocation$ (positional)

A = $isa(Line, Geometry)$
$isa(Polygon, Geometry)$
$isa(Point, Geometry)$
$isa(Avenue, Road)$
$isa(Street, Road)$
$isa(Hospital, Building)$
$isa(Temple, Building)$
$isa(School, Building)$
$disjoint(Factory, Hospital)$
$crosses(Road, some(IndustrialDistrict))$
$inside(Factory, some(IndustrialDistrict))$
$hasGeometry(Road, Line)$
$hasGeometry(IndustrialDistrict, Polygon)$
$hasGeometry(Factory, Point)$
$hasGeometry(Hospital, Point)$
$hasGeometry(Temple, Point)$
$hasGeometry(School, Point)$

I_g = $instanceOf(Avenue, \text{“}Av.doTrabalhador'', \{(hasGeometry, line1),$
$(numLanes, 2), (length, 32), (crosses, Restinga)\})$
$instanceOf(IndustrialDistrict, \text{“}Restinga'', \{(hasGeometry, pol1)\})$
$instanceOf(Factory, \text{“}Cavo'', \{(hasGeometry, pt1),$
$(inside, Restinga), (isPollutant, \text{“}true'')\})$
$instanceOf(Hospital, \text{“}ErnestoDorneles'', \{(hasGeometry, pt2),$
$(numBeds, 126)\})$
$instanceOf(Line, line1, \{(hasLocation, ((45N, 8E); (44N, 13E)))\})$
$instanceOf(Point, pt1, \{(hasLocation, (45.5N; 11E))\})$
$instanceOf(Point, pt2, \{(hasLocation, (45N, 8.5E))\})$
$instanceOf(Polygon, pol1, \{(hasLocation, ((45N, 10E); (45N, 12E);$
$(47N, 12E); (47N, 10E); (45N, 10E)))\})$

M = $CoordinateReferenceSystem(UTM)$
$ProjectionScale(1 : 100.000)$

Figure 2.4: Ontology O' defined according to the proposed model

2.3 Classification of heterogeneities

In this section we formally present the heterogeneities to be faced when comparing two geographic ontologies, at the concept-level and the instance-level as well. For easiness of comprehension, Figure 2.5 shows an ontology that is compared against the ontology of Figure 2.2 to illustrate the heterogeneities. The respective encoding is presented in Figure 2.6.

2.3.1 Concept-level heterogeneities

In this section the possible heterogeneities are classified regarding the comparison of the context of a concept $c \in$ ontology O against the context of a concept $c^{'} \in$ ontology $O^{'}$. For now on, when we refer to a concept we are meaning its whole context. Considering the definitions presented in section 2.2, the possible heterogeneities are defined in the following.

2.3.1.1 Concept equivalence

Before of defining the heterogeneities, we must first define when two concepts are considered as equivalent.

Definition 1 *A concept $c \in O$ is said to be equivalent to a concept $c^{'} \in O^{'}$ when they have a similarity degree $Sim(c, c^{'})$ over a certain threshold ϵ. In this case it is said that c and $c^{'}$ are matching concepts.*

$$(c \equiv c^{'}) = (Sim(c, c^{'}) \geq \epsilon)$$

This similarity is measured considering the different features of a concept, such as the concept name and its context (properties, axioms, hierarchies). The similarity measurement procedure, also known as matching process, for the concept-level is detailed in Chapter 4.

2.3.1.2 Name heterogeneity

Definition 2 *The concept name heterogeneity NH occurs when given two concepts c and $c^{'}$, their labels $t(c)$ and $t(c^{'})$ are neither equal nor synonyms. The synonym relation $SYN(t(c), t(c^{'}))$ is tested by searching an external thesaurus.*

$$NH(c, c^{'}) = ((t(c) \neq t(c^{'})) \wedge (SYN(t(c), t(c^{'})) = false))$$

Considering the ontologies O and $O^{'}$, the concepts of *Building*, from O, and *Cathedral*, from $O^{'}$, are examples of name heterogeneity. On the other hand, *Road* from O and *Route* from $O^{'}$ do not have name heterogeneity, because although the terms are not equal, the function $SYN(t(c), t(c^{'})$ returns true when searching an external dictionary.

2.3.1.3 Attribute and relationship heterogeneity

Definition 3 *The concept conventional property heterogeneity PH occurs when there is an attribute heterogeneity AH or a relationship heterogeneity RH.*

The AH heterogeneity between $c \in O$ and $c^{'} \in O^{'}$ occurs when at least one of the attributes $a(t(p), dtp) \in \{p(c)\}$ in ontology O does not have a correspondent attribute, $a(t(p^{'}), dtp^{'}) \in \{p(c^{'})\}$ in ontology $O^{'}$. The heterogeneity can exist due to different attribute names ($t(p)$) or different attribute data types (dtp).

$$AH(c, c^{'}) = (\exists a(t(p), dtp) \in \{p(c)\} | \forall a(t(p)^{'}, dtp^{'}) \in \{p(c^{'})\}, (NH(p, p^{'})) \vee (NH(dtp, dtp^{'})))$$

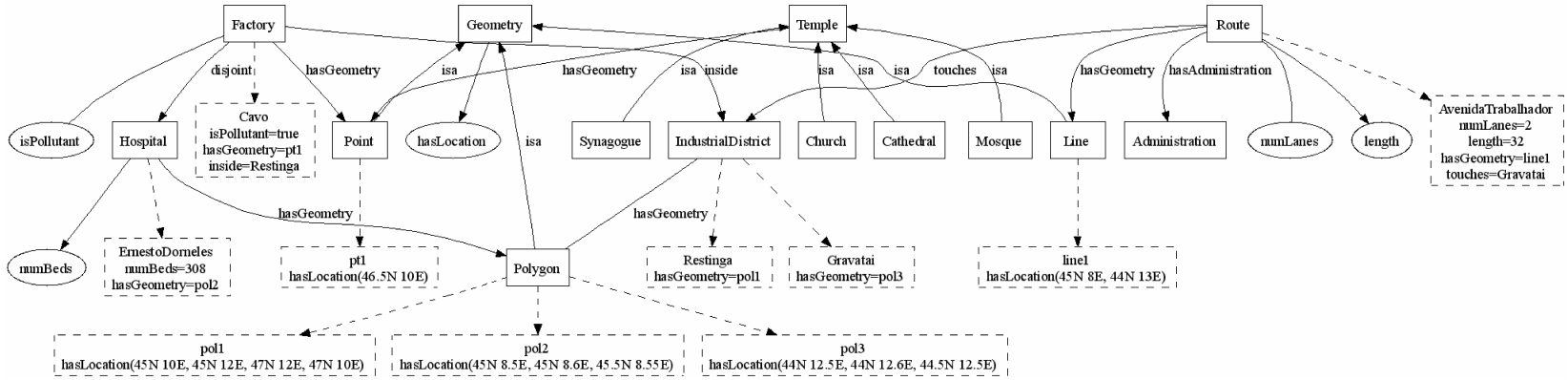

Figure 2.5: Second example of geographic ontology O'

The rectangles with continuous lines represent concepts, the ellipses the properties representing attributes associated with a concept and the dashed rectangles the instances belonging to a concept. The arcs linking two concepts correspond to the properties which represent relationships holding between them, while the isa labeled arrows are the taxonomic relationships (axioms) between two concepts, in which one is the specialization of the other.

C_g = $Route, Factory, IndustrialDistrict, Hospital$ (geographicdomain)
$Temple, Synagogue, Church, Cathedral, Mosque$ (geographicdomain)
$Administration$ (domain)
$Geometry, Line, Polygon, Point$ (geometry)

P_g = $isPollutant, numBeds, numLanes, length, hasAdministration$ (conventional)
$disjoint, touches, inside$ (spatial)
$hasGeometry$ (geometric)
$hasLocation$ (positional)

A = $isa(Line, Geometry)$
$isa(Polygon, Geometry)$
$isa(Point, Geometry)$
$isa(Church, Temple)$
$isa(Synagogue, Temple)$
$isa(Mosque, Temple)$
$isa(Cathedral, Temple)$
$disjoint(Factory, Hospital)$
$touches(Route, some(IndustrialDistrict))$
$inside(Factory, some(IndustrialDistrict))$
$hasGeometry(Route, Line)$
$hasGeometry(IndustrialDistrict, Polygon)$
$hasGeometry(Factory, Point)$
$hasGeometry(Hospital, Polygon)$
$hasGeometry(Temple, Point)$

I_g = $instanceOf(Route, \text{“}Av.doTrabalhador'', \{(hasGeometry, line1),$
$(numLanes, 2), (length, 32), (crosses, Gravatai)\})$
$instanceOf(IndustrialDistrict, \text{“}Restinga'', \{(hasGeometry, pol1)\})$
$instanceOf(IndustrialDistrict, \text{“}Gravatai'', \{(hasGeometry, pol3)\})$
$instanceOf(Factory, \text{“}Cavo'', \{(hasGeometry, pt1),$
$(inside, Restinga), (isPollutant, \text{“}true'')\})$
$instanceOf(Hospital, \text{“}ErnestoDorneles'', \{(hasGeometry, pol2),$
$(numBeds, 308)\})$
$instanceOf(Line, line1, \{(hasLocation, ((45N, 8E); (44N, 13E)))\})$
$instanceOf(Point, pt1, \{(hasLocation, (46.5N; 10E))\})$
$instanceOf(Polygon, pol1, \{(hasLocation, ((45N, 10E); (45N, 12E);$
$(47N, 12E); (47N, 10E); (45N, 10E)))\})$
$instanceOf(Polygon, pol2, \{(hasLocation, ((45N, 8.5E); (45N, 8.6E);$
$(44N, 8.6E); (44N, 8.5E); (45N, 8.5E)))\})$
$instanceOf(Polygon, pol3, \{(hasLocation, ((44N, 12.5E); (44N, 12.6E);$
$(44.5N, 12.5E); (44N, 12.5E)))\})$

M = $CoordinateReferenceSystem(UTM)$
$ProjectionScale(1 : 100.000)$

Figure 2.6: Ontology O' defined according to the proposed model

As an example of attribute heterogeneity, lets consider the concepts *Temple* from O and *Temple* from $O^{'}$. The attribute *religion* is a property of *Temple* in the first ontology, but it is not associated to *Temple* in the second one.

Definition 4 *The RH heterogeneity between $c \in O$ and $c^{'} \in O^{'}$ is defined over the conventional relationships (i.e., neither geometric nor spatial). It applies to both geographic as well as to non-geographic concepts. It occurs when at least one of the relationships $cr(t(p), t(c_x), minCard(p), maxCard(p)) \in \{p(c)\}$ in ontology O, where $t(p)$ is the name of the relationship, $t(c_x)$ is the name of the associated concept and $minCard(p)$ and $maxCard(p)$ are, respectively, the minimum and maximum cardinality of the relationship, does not have a correspondent* cr($t(p^{'})$, $t(c_x^{'})$, $minCard(p^{'})$, $maxCard(p^{'})$) $\in \{p(c^{'})\}$ *in ontology $O^{'}$. The heterogeneity may occur due to a different associated concept $t(c_x)$ as well as due to the relationship cardinalities $minCard(p)$ or $maxCard(p)$. Since in many cases the conventional relationships names are not significant as to identify the relationship, the component $t(p)$ can be ignored.*

$$RH(c, c^{'}) = (\exists cr(t(p), t(c_x), minCard(p), maxCard(p)) \in \{p(c)\} | \forall cr(t(p^{'}), t(c_x^{'}), minCard(p^{'}), maxCard(p^{'})) \in \{p(c^{'})\}, (NH(c_x, c_x^{'})) \vee (minCard(p) \neq minCard(p^{'})) \vee (maxCard(p) \neq maxCard(p^{'})))$$

As an example of relationship heterogeneity, lets consider the concepts *Road* from O and *Route* from $O^{'}$. In the context of *Route* there is a relationship *hasAdministration* with the concept *Administration*, that does no exist in the context of the concept *Road*.

2.3.1.4 Geographical heterogeneity

Regarding the geographic domain concepts, two additional types of heterogeneities can be identified, one for each type of relationship (geometry and spatial relation).

Definition 5 *The geometric concept heterogeneity GH between $gc \in O$ and $gc^{'} \in O^{'}$ happens when the two geographic concepts gc and $gc^{'}$ have different geometries, i.e., the* hasGeometry *property relates the geographic domain concepts to concepts representing different geometries.*

$$GH(gc, gc^{'}) = (\exists ge(\text{hasGeometry}, t(geo), minCard(p), maxCard(p)) \in \{p(gc)\} | \forall ge(\text{hasGeometry}^{'}, t(geo^{'}), minCard(p^{'}), maxCard(p^{'})) \in \{p(gc^{'})\}, NH(geo, geo^{'}))$$

In this case only the associated geometry concept geo counts, because it is the one which defines the geometry (point, line, polygon) of the geographic concept. Due to the possibility of the multi-representation of a geographic concept, i.e., multiple geometries, if at least one of the geometries of gc matches with a geometry of $gc^{'}$, there is no heterogeneity.

As an example of geometric heterogeneity, lets consider the concepts *Hospital* from O and *Hospital* from $O^{'}$. While in the former the *hasGeometry* property associates it with the concept *Point*, in the latter the *hasGeometry* property links it with the concept *Polygon*.

In the case of spatial relationships, specially in the case of the topological ones, the geometry plays an essential role. In (BELUSSI; CATANIA; PODESTÀ, 2005) the equivalences between topological relationships are defined according to the geometries of the

involved concepts. Following this idea, the spatial relationship heterogeneity can be divided into topological relationship heterogeneity and directional relationship heterogeneity. The metric relationships are not considered because in general they are calculated by a GIS and not defined as properties or restrictions of a concept. As the names of the spatial relationships are, in general, standardized in the literature, the component $t(p)$, which holds the relationship name, has to be considered.

Definition 6 *The directional relationship heterogeneity DH between two geographic concepts $gc \in O$ and $gc^{'} \in O^{'}$ occurs when there is at least one directional relationship* dr(t(p),$t(gc_x)$,minCard(p),maxCard(p)) $\in \{p(gc)\}$ *in ontology O without a matching (i.e., correspondent) $dr(t(p^{'}), t(gc_x^{'}), minCard(p^{'}), maxCard(p^{'})) \in \{p(gc^{'})\}$ in ontology $O^{'}$, where $t(gc_x)$ is the associated concept, $t(p)$ is the relationship name and $minCard(p)$ and $maxCard(p)$ are, respectively, the minimum and maximum cardinalities.*

$$DH(gc, gc^{'}) = (\exists dr(t(p), t(gc_x), minCard(p), maxCard(p)) \in \{p(gc)\} | \forall dr(t(p^{'}), t(gc_x^{'}), minCard(p^{'}), maxCard(p^{'})) \in \{p(gc^{'})\}, (NH(gc_x, gc_x^{'})) \vee (NH(p, p^{'})))$$

The definition of the topological relationship heterogeneity is a little more complex, because of the equivalences of relationships depending on the associated geometries.

Definition 7 *Given two geographic concepts $gc \in O$ and $gc^{'} \in O^{'}$, they are said to have topological relationship heterogeneity TH if the combination of the relationship name and the involved geometries, given by a function $top(t(geo), t(geo_x), t(p))$ and $top(t(geo^{'}), t(geo_x^{'}), t(p^{'}))$ are not equivalent, where $t(geo)$ and $t(geo_x)$ are, respectively, the names of geometries of the concepts gc and gc_x and $t(p)$ is the relationship name.*

$$TH(gc, gc^{'}) = (\exists tr(t(c), t(gc_x), minCard(p), maxCard(p)) \in \{p(gc)\} | \forall tr(t(p^{'}), t(gc_x^{'}), minCard(p^{'}), maxCard(p^{'})) \in \{p(gc^{'})\}, top(t(geo), t(geo_x), t(p)) \neq top(t(geo^{'}), t(geo_x^{'}), t(p^{'})))$$

An Example of a spatial relationship heterogeneity is the association *Road crosses IndustrialDistrict* in ontology O and *Route touches IndustrialDistrict* in ontology $O^{'}$. Even if we consider that *Route* and *Road* could be synonyms, in ontology O the relationship name is *crosses*, while in $O^{'}$ $t(p^{'})$=*touches*. As will be discussed later, these relationships may be equivalent, but in a first analysis it seems that we have a spatial relationship heterogeneity.

2.3.1.5 Hierarchy heterogeneity

Definition 8 *The hierarchy heterogeneity HH between two concepts $c \in O$ and $c^{'} \in O^{'}$ occurs when the set of superclasses $SUP(c)$ of the concept $c \in O$ is different from the set of superclasses $SUP(c^{'})$ of the concept $c^{'} \in O^{'}$. This means that at least one of the superclasses present in $SUP(c)$ is not found in $SUP(c^{'})$.*

$$HH^{'}(c, c^{'}) = (\exists c_x \in h(c, c_x) | \forall c_x^{'}) \in h(c^{'}, c_x^{'}), c_x \neg \equiv c_x^{'})$$

Even if the set of superclasses of the compared concepts are the same, they still can have hierarchy heterogeneity if the levels of the superclasses in the hierarchies are different, i.e., if two concepts $c_x \in O$ and $c_x^{'} \in O^{'}$ are equivalent and are superclasses

of, respectively, c and $c^{'}$, but the distances $dist(c, c_x)$ and $dist(c^{'}, {c_x}^{'})$ are different, then there is also hierarchy heterogeneity.

$$HH^{''}(c, c^{'}) = (\exists t(c_x) \in h(c, c_x), \exists t(c_x^{'}) \in h(c, c_x^{'}) | (c_x \equiv c_x^{'}) \wedge (dist(c, c_x) \neq (dist(c^{'}, c_x^{'}))))$$

where $dist(c, c_x)$ is the distance between the concepts c and c_x, i.e., the number of concepts between them.

Therefore, the hierarchy heterogeneity can be defined as:

$$HH(c, c^{'}) = HH^{'}(c, c^{'}) \vee HH^{''}(c, c^{'})$$

The concepts *Temple* and *Temple* from O and $O^{'}$, respectively, are examples of hierarchy heterogeneity. The former has as superclass the concept *Building*, while the latter does not have any superclass (actually, in an ontology, all concepts are subclasses of *thing*, but for easiness of comprehension we omitted it from the ontology).

2.3.1.6 Concept heterogeneity

By analyzing the definitions above, we can now define the heterogeneity when comparing two concepts contexts.

Definition 9 *Two concepts c and $c^{'}$ have heterogeneity when they present name heterogeneity NH or when they have heterogeneity between their contexts, as follows:*

$$CH(c, c^{'}) = NH(c, c^{'}) \vee AH(c, c^{'}) \vee RH(c, c^{'}) \vee HH(c, c^{'})$$

Two geographic concepts gc and $gc^{'}$ have heterogeneity when they present name heterogeneity NH or when they have heterogeneity between their contexts, as follows:

$$CH(gc, gc^{'}) = NH(gc, gc^{'}) \vee AH(gc, gc^{'}) \vee RH(gc, gc^{'}) \vee HH(gc, gc^{'}) \vee DH(gc, gc^{'}) \vee TH(gc, gc^{'})$$

2.3.2 Instance-level heterogeneities

As important as the matching of geographic ontology concepts is the matching of their instances. Especially in the geographic field there are many features that can influence the similarity measurement process which are not present when dealing with non-geographic ontologies. These features are, for example, the scale, spatial position, time when the instances were obtained, and so on. However, the non-spatial properties, such as the attributes (property) values, cannot be neglected either. In this section we define the heterogeneities that may occur at the instance-level when comparing two geographic ontologies.

2.3.2.1 Instance equivalence

Before of defining the heterogeneities at the instance-level, we must define when two instances are considered to be equivalent.

Definition 10 *An instance $i \in O$ is said to be equivalent to an instance $i^{'} \in O^{'}$ when they represent objects from equivalent concepts $c \in O$ and $c^{'} \in O^{'}$, respectively, and have a similarity degree $Sim(c, c^{'})$ over a certain threshold ϵ. In this case it is said that i and $i^{'}$ are matching instances.*

$$(i \equiv i^{'}) = (Sim(i, i^{'}) \geq \epsilon)$$

This similarity is measured considering the values of the different features of an instance, such as the instance identifier, the values of its properties and its location (spatial position). The similarity measurement procedure, also known as matching process, for the instance-level is detailed in Chapter 5.

2.3.2.2 Identifier heterogeneity

When a concept in an ontology is instantiated, in general the unique identifier has a really significant value. It is not like the *objectId* of an instance of a class which is automatically generated. In the case of an ontology it is the main way for both the user and the computer to identify the instance, i.e., the instance name. In the example of section 2.2 the identifier of the instance of the concept River is Po and for the instance of the Road concept is A1.

Definition 11 *When two instances $i \in O$ and $i^{'} \in O^{'}$ do not have the same identifier (in OWL, the ID parameter) there is an identifier heterogeneity IIH between them.*

$$IIH(i, i^{'}) = (NH(i, i^{'}))$$

The concepts c and $c^{'}$ the instances belong to are not considered because they should be already identified as equivalent. As an example of instance identifier heterogeneity we have *Av. do Trabalhador* in ontology O and *AvenidaTrabalhador* in ontology $O^{'}$ (assuming we have already inferred that *Route* and *Avenue* are equivalent concepts).

2.3.2.3 Coordinates heterogeneity

As already stated, one of the main characteristics of the geographic data is that it has a position over (or under) the earth surface. The set of coordinates of a given instance $i \in O$ is obtained indirectly through the instance of the geometry concept that is associated to it. If two instances $i \in O$ and $i^{'} \in O^{'}$ do not have the same spatial position, there is a positional heterogeneity ICH. In order to simplify the formalization of the positional heterogeneity, we assume that a function $pos(i)$ gives the location of the instance. This function gets the set of coordinates from the geometry instance which is associated to the geographic instance by a geometric property.

Definition 12 *If two instances $i \in O$ and $i^{'} \in O^{'}$ do not have the same spatial position, they have coordinate heterogeneity ICH.*

$$ICH(i, i^{'}) = (pos(i) \neq pos(i^{'}))$$

The simple comparison of the values of the spatial coordinates would be a naive and simplistic definition. If the coordinate reference system and projection system of the compared instances are not the same, the harmonization of this meta information must be executed first. Furthermore, if the geometries associated to the instances are different (e.g., i has a `point` geometry and $i^{'}$ has a `line` geometry), they must be firstly standardized to the same geometry and then the coordinates can be compared. In the definition above we assume that the coordinate reference system as well as the projection system, and the geometries are the same (originally or the translation was already performed).

An example of coordinate heterogeneity is found when comparing the instances of the concept *point* identified as *pt1* in O and in $O^{'}$ as well. The *hasLocation* property has the value "(45.5N,11.0E)" in O, but in ontology $O^{'}$ that property has the value of "(46.5N,10.0E)".

2.3.2.4 Metadata heterogeneity

The metadata does not have a direct influence on the heterogeneity between two geographic instances. Instead, the influence is indirect, which means that difference on the metadata values may lead to heterogeneities regarding the other properties of the instance, such as attributes, coordinates and relationships as well.

e.g.1. Depending on the value for the $date$ metadata, the value for some descriptive attributes may vary (for example, the population of a city). Even some spatial relationships may vary depending on the capture of the information.

e.g.2. Depending on the value for the $projection$ (UTM, planar) metadata, the geometry as well as coordinates of an instance may change.

e.g.3. Depending on the values for the *measurement units* metadata, the values for some descriptive attributes may vary (for example, the height of a monument, if expressed in meters or in feet).

Definition 13 *If two instances $i \in O$ and $i^{'} \in O^{'}$ are not described using the same values for the metadata, they have metadata heterogeneity IMH.*

$$IMH(i, i^{'}) = (\exists vmd(t(mdt), val) \in \{mdv(i)\} | \forall vmd(t(mdt^{'}), val^{'}) \in \{mdv(i^{'})\}, \\ (t(mdt) \equiv t(mdt^{'}) \wedge (val \neq val^{'}))$$

2.3.2.5 Attribute heterogeneity

When a property of a concept is a data type property which does not represent its coordinates, i.e., is not the *hasLocation* property, it represents an scalar attribute. They are properties which allowed values are string, float, integer, etc and do not represent spatial attributes. In this case the relation $vp(t(p), val)$ can be identified as $at(t(p), v)$.

Definition 14 *When two instances $i \in O$ and $i^{'} \in O^{'}$ have different values v for the same scalar data type property p there is an attribute heterogeneity IAH.*

$$IAH(i, i^{'}) = (\exists at(t(p), v) \in VP | \forall at(t(p^{'}), v^{'}) \in VP^{'}, (p \equiv p^{'}) \wedge (v \neq v^{'}))$$

where $t(p)$ is the name of the property p and v is the value of that property.

For example, let consider the instances identified as *Ernesto Dorneles*. They belong to the concept *Hospital* in both ontologies O and $O^{'}$. If we consider that the attribute *numBeds* was found in both concepts and considered as representing the same information in both ontologies, we can compare their values. In ontology O we find 126, while in ontology $O^{'}$ we get 308. This characterizes the instance attribute heterogeneity.

2.3.2.6 Relationship heterogeneity

When a property of a concept is an object type property it represents a relationship, i.e., a property which allowed values are instances of other concepts. In this case the relation $vp(t(p), val)$ can be identified as $rl(t(p), t(i_x))$.

Definition 15 *If the instances i_x and $i_x^{'}$ are in the range of object type properties of, respectively, $i \in O$ and $i^{'} \in O^{'}$, but are not equivalent instances, there is a relationship heterogeneity IRH.*

$$IRH(i, i^{'}) = (\exists rl(t(p), t(i_x)) \in VP | \forall rl(t(p^{'}), t(i_x^{'})) \in VP^{'}, NH(i_x, i_x^{'}))$$

where $t(p)$ is the name of the property p and i_x is the associated instance.

An example of instance relationship heterogeneity is found when comparing the instances *Av. do Trabalhador* and *AvenidaTrabalhador* from, respectively, *Road* in O and *Route* in $O^{'}$. Considering that the concepts are equivalent and supposing the properties *crosses* associated to *Road* and *touches* associated to *Route* are equivalent as well, the values these properties hold are, respectively, *Restinga* and *Gravatai*. As the instances *Restinga* and *Gravatai* are not equivalent, there is an instance relationship heterogeneity.

2.3.2.7 Instance heterogeneity

By analyzing the definitions above, we can now define the heterogeneity when comparing two instances.

Definition 16 *Two instances i and $i^{'}$ have heterogeneity when they present identifier heterogeneity IIH or when they have heterogeneity between the values of their properties, as follows:*

$$IH(i, i^{'}) = IIH(i, i^{'}) \vee IAH(i, i^{'}) \vee IRH(i, i^{'})$$

Definition 17 *Two geographic instances gi and $gi^{'}$ have heterogeneity when they present identifier heterogeneity IIH or when they have heterogeneity between the values of their properties, or if they have spatial position heterogeneity ICH, as follows:*

$$IH(gi, gi^{'}) = IIH(gi, gi^{'}) \vee IAH(gi, gi^{'}) \vee IRH(gi, gi^{'}) \vee ICH(gi, gi^{'})$$

2.4 Publications

The geographic ontology model presented here was published in the Brazilian Symposium on GeoInformatics 2007 (GeoInfo 2007) (HESS; IOCHPE; CASTANO, 2007a). The discussion and formalization of heterogeneities presented in this chapter was also already published, in the Second International Conference on Geospatial semantics (GeoS 2007) (HESS et al., 2007).

3 AN OVERVIEW OF GEOGRAPHIC ONTOLOGY SEMANTIC MATCHING

In this chapter we survey the state-of-the-art in the field of geographic information matching. As information matching as well as mapping, and integration are closely related concepts, instead of restricting our survey to matching proposals, we also include some references to integration and mapping as well.

3.1 Integration, mapping and matching definition

As in this chapter we include proposals for integrating and mapping geographic information, it is important to define what mapping, integration and matching mean. **Matching** is the process of identifying which elements (instances or classes) in a source $S1$ correspond to which elements in a target $S2$. During this process, a similarity measure is assigned for each pair of matching elements. Moreover, depending on the application, this measure should determine whether two elements are equivalent or not.

A **mapping** is a specification that describes how data structured according to a source schema (or ontology) $S1$ is to be transformed into data structured under a target schema (or ontology) $S2$ (FAGIN et al., 2005). The output of the mapping is a number of mapping functions, which, in general, are based on the result of a matching process.

Integration can be defined as a process that receives as input two sources (ontologies or schemas) $S1$ and $S2$ and produces an output S, which is composed by the elements from $S1$ and $S2$ (KALFOGLOU; SCHORLEMMER, 2003). The integration consists in the application of the functions produced by the mapping to actually translate the sentences that use the first ontology into the second.

To illustrate the definitions above, lets consider the two concepts (represented as classes) in Figure 3.1[1].

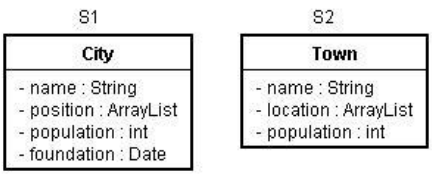

Figure 3.1: Concepts to be compared

[1]For the sake of clarity, in the example we consider only concept properties, but the considerations can be applied also to other elements such as axioms.

By informally applying the matching process to concepts, the two concepts are compared, considering their labels and properties. The output of the matching could be the following (the similarity values are illustrative):

Sim(City.name,Town.name) = 1.0
Sim(City.name,Town.location) = 0.3
Sim(City.name,Town.population) = 0.1
Sim(City.position,Town.name) = 0.2
Sim(City.position,Town.location) = 0.7
Sim(City.position,Town.population) = 0.6
Sim(City.population,Town.name) = 0.1
Sim(City.population,Town.location) = 0.6
Sim(City.population,Town.population) = 1.0
Sim(City.foundationDate,Town.name) = 0.0
Sim(City.foundationDate,Town.location) = 0.2
Sim(City.foundationDate,Town.population) = 0.2

Based on these similarity values, we can establish the following mapping functions between sources $S1$ and $S2$:

City.name → Town.name
City.position → Town.location
City.population → Town.population
City.foundationDate → ∅

Finally, an example of the integration of concepts of Figure 3.1 is shown by Figure 3.2:

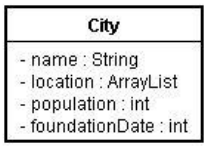

Figure 3.2: A possible result of the integration of concepts of Figure 3.1

3.2 The state-of-the-art

Each one of the works presented in this section may use a particular vocabulary for the features addressed by the proposed technique. Every one of these features is covered by the model presented in Chapter 2. Therefore, to homogenize the terminology we use in the rest of this chapter, tables 3.1, 3.4 and 3.6 present the correspondences between the reference model and the vocabulary of each geographic information integration or mapping ***giim*** proposal.

We classified the works into three main categories as follows. The concept-level approaches, which address the problem of geographic information integration, mapping and matching (***giim***) considering only the structure of the ontology, i.e., concepts, their properties and the taxonomy in which they are inserted. The instance-level works address the ***giim*** issue regarding the actual data, i.e., the individuals of the ontology and their assigned property values. Finally, the combined proposals deal with both concept and instance-levels of ***giim***.

3.2.1 Evaluation criteria

According to the model presented in Chapter 2 and the basic definitions given above, we define a general set of criteria to compare the ***giim*** approaches presented in this chapter. Additional criteria holding only for some specific system/approach are introduced

when necessary in the corresponding section. The criteria were chosen based on two different points of view: (1) regarding the coverage of the geographic ontology model, i.e., to see in what extent the features of the model we proposed in the previous chapter are used in the matching process (e.g., properties, spatial relations and hierarchies) of each specific proposal; and (2) regarding the results produced by the ***giim*** proposals (e.g., the type of output and the existence of a prototype implementing the approach). Therefore we elaborated a set of criteria to evaluate the works presented in this survey, which comprises the features we consider as important to match geographic ontologies and are therefore the ones we selected for our algorithms.

The set of general criteria used in the survey is the following:

1 **Output**: this criterion refers to the type of the result produced by the approach, i.e, mapping or integration of the input sources;

2 **Prototype**: the existence of tools implementing the approach;

3 **Linguistic**: the use of techniques that explore the textual features of the elements to be compared. They can be string-based (e.g., distance metrics), linguistic-based or semantic, i.e., based on a dictionary or thesaurus.

4 **Spatial features** (SR): differentiation of the properties which represent any kind of spatial relation between either concepts or instances.

5 **Measurement procedure** (measure. proc.): the technique used by the proposal to determine the similarity between concepts/instances.

Analogously to what Rahm and Bernstein (RAHM; BERNSTEIN, 2001) did in a survey on schema matching, we classify the ***giim*** approaches into three categories: concept-level, instance-level and combined. Concept-level techniques are the ones that address the mapping/integration regarding the schema of the ontology. Instance-level approaches work on the data stored in the ontology. Finally, combined techniques address the integration/mapping of concepts as well as of instances. Thus, some criteria can be applied to all categories, while some other belong specifically to one or another category. Therefore, in the following we define the set of criteria applicable for the three categories. The criteria specific for one of the categories are presented in the corresponding section.

3.2.2 Concept-level proposals

In this section we discuss the techniques which integrate, map or match the geographic ontologies/schemas only at the concept level. The common features with conventional ontologies/schemas are briefly described, while more attention is given to the specific features of the geographic information.

We selected ten works, which we consider as the most relevant and related to this research. Furthermore, only recent works, i.e., from 2002 at least, are discussed here. As these works do not share the same vocabulary, the first thing we did was translating them into the vocabulary used in our ontology model. Table 3.1 presents the correspondences among the works vocabulary and the ontology model presented in Chapter 2. Due to space limitations, instead of the full reference in the table headers, we used an acronym, which is composed by the first letter of each author's last name.

Table 3.1: Correspondences between the ontolgy model and the approaches vocabulary

#	Concept	Property (attribute)	Property (conv. relation)	Property (spt relation)	Axiom
RE	concept	attribute	parts	-	IS-A
FDC	concept	-	property	geog. relation	IS-A
SVCBA	concept	property	property	spatial	IS-A
CSC	concept	-	-	-	IS-A
SD	concept	-	association	relationship	IS-A
QRCG	concept	attribute	conv. relation	GeometryType	IS-A
KK	concept	semantic element	semantic elem.	semantic elem.	-
SR	concept	-	-	spt. relation	-
VSS	concept	-	-	-	-
BBM	geoConcept	attribute	relationship	spt. relation	IS-A

In addition to the general criteria listed in Section 3.2.1, we introduce specific criteria for this category, as follows.

6 **Common Knowledge Base (CKB)**: use of a global ontology, which acts as a reference knowledge base for the integration/mapping process.

7 **Language**: standard/formal language in which the input ontologies or schemas must be described.

8 **Annotation**: the textual description of concepts (e.g., the annotations in an OWL ontology).

9 **Non-geographic context** (non geo ctx): kind of semantic relations considered, such as properties (attributes (a) or relationships (b)) and axioms (c).

Tables 3.2 and 3.5 compare the concept-level ***giim*** approaches based on criteria 1 to 9.

As discussed in the introduction section, not all the proposals have the same goal. The mapping of two geographic ontologies or schemas at concept-level is produced as result in Quix et al. (QRCG) (2006), Kavouras et al. (KK) (2005), Brodeur et al. (BBM) (2005), and Cruz et al. (CSC) (2004; 2007). The integration of the two compared schemas is, on the other hand, the result of the approaches of Fonseca et al. (FDC) (2003), Stoimenov and Djordjevic-Kajan (SD) (2005), Rodriguez and Egenhofer (RE) (2003), the University of Munster Geomatics group (SR) (SCHWERING; RAUBAL, 2005a), Stonykova et al. (SVCBA) (2005) and Visser et al. (VSS) (2002).

Not all of the ***giim*** approaches consider linguistic features in the matching process. The ones that do consider linguistic features try to detect if two concepts are defined by synonym terms (labels). These synonyms may be defined at the internal of the compared ontologies (RODRIGUEZ; EGENHOFER, 2003) or may be searched in external dictionaries (KAVOURAS; KOKLA; TOMAI, 2005; STOIMENOV; DJORDJEVIC-KAJAN, 2005; VISSER; STUCKENSCHMIDT; SCHLIEDER, 2002; SUNNA; CRUZ, 2007; RODRÍGUEZ; EGENHOFER, 2004).

Table 3.2: Comparative analysis of geographic schema matchers

Criterion	RE	FDC	SVCBA	CSC	SD
Output	Integ	Integ	Integ	Mapp.	Integ
Prototype	X	X	√	√	√
Linguistic	√	√	√	√	√
SR	X	√	√	X	X
Measure. proc.	Set theory	Formal	Set theory	Algebra	N/A
CKB	√	√	√	√	√
Language	-	OWL	DL	XML	GML
Annotation	X	X	X	X	√
Non geo ctx	a,b,c	b,c	a,b,c	c	b,c

a means attributes, *b* means relationships and *c* means axioms.
the √ symbol represents that the criterion is covered by the work, while the X means it is not. - means it is not specified.
OWL is the Ontology Web Language, DL means Description Logics and GML is the Geography Markup Language.

Table 3.3: Comparative analysis of geographic schema matchers

Criterion	QRCG	KKT	SR	VSS	BBM
Output	Mapp.	Mapp.	Integ	Integ	Mapp.
Prototype	√	√	√	√	√
Linguistic	√[a]	X	X	√	√
SR	√	√[b]	√	X	√
Measure. proc.	see[a]	N/A	Algebra	NONE	Set theory
CKB	√	X	√	√	X
Language	OWL	N/A	Haskell	OIL	XML
Annotation	X	√	√	√	X
Non geo ctx	a,b,c[a]	a,c[b]	X	X	a,b,c

a means attributes, *b* means relationships and *c* means axioms.
[a]use of existing matchers [b]the relations are extracted from the description of the concepts
the √ symbol represents that the criterion is covered by the work, while the X means it is not. - means it is not specified.
OWL is the Ontology Web Language, DL means Description Logics and GML is the Geography Markup Language.

Regarding the concepts' context, the hierarchies are the features most commonly used in the matching process. Furthermore, the spatial features are also used. The spatial relations are often part of the similarity assessment, while the geometries are rarely considered.

Finally, the use of a common knowledge base is almost unanimous by the concept-level ***giim*** approaches. It can be used either as mediator in the matching process (FONSECA et al., 2002; FONSECA; DAVIS; CAMARA, 2003; STOIMENOV; DJORDJEVIC-KAJAN, 2005, 2003; QUIX et al., 2006) or as an auxiliary structure during the mapping/integration (RODRIGUEZ; EGENHOFER, 2003; KUHN, 2002; KLIEN et al., 2004; SCHWERING; RAUBAL, 2005a,b; SOTNYKOVA; CULLOT; VANGENOT, 2005; SOTNYKOVA et al., 2005; VISSER; STUCKENSCHMIDT; SCHLIEDER, 2002; CRUZ; SUNNA; CHAUDHRY, 2004).

In the following we critically analyze the techniques according to each general/specific criterion.

3.2.2.1 Output

Except from the work of Sotnykova et al. (SOTNYKOVA; CULLOT; VANGENOT, 2005) and of the GeoNis framework (STOIMENOV; DJORDJEVIC-KAJAN, 2005), all the others that produce as output an integrated ontology do not define levels of similarity. They simply have a unique integrated ontology at the end of the process. In (SOTNYKOVA; CULLOT; VANGENOT, 2005; SOTNYKOVA et al., 2005), depending on the degree of similarity of the two compared ontologies, four different outputs may be delivered:

- Fusion, which means that the information of both schemas are integrated. All the information is preserved, but, as a drawback, it generates a lot of NULL values;
- Union[2], which performs the intersection of the schemas, i.e., only what is found in both schemas is preserved;
- Multi-representation, which preserves both spatial representations for an entity when each schema uses one;
- Generalization-partition, which consists of preserving the two original concepts and creating a third one with the common information.

To enable the integration/mapping of two ontologies O and $O^{'}$, it is necessary to compare the concepts and relations they represent. For the matching of the concepts, it is quite usual to define different degrees of similarity between a pair of concepts $(c, c^{'})$ from, respectively, O and $O^{'}$ (SOTNYKOVA et al., 2005; KOKLA; KAVOURAS; TOMAI, 2005; BRODEUR; BÉDARD; MOULIN, 2005; SUNNA; CRUZ, 2007; STOIMENOV; DJORDJEVIC-KAJAN, 2005). The terminology may vary from proposal to proposal, but the meanings are basically the same:

- equivalence (equality, exact or *isequal*), when besides the equality of the terms labeling the concepts, the set of properties p in the context of c is the same as the set of properties $p^{'}$ in the context of $c^{'}$, as well as the values for that properties;

[2] Although the performed operation is actually an intersection, we kept the term union because is the one used in the original work.

- difference (dissimilarity or *isdisjoint*), when the concepts have different meanings, i.e., the set of properties p associated to c is completely different of the set of properties p' in the context of c';
- subsumption (contain, super/subset *includes*), when the set of properties (or values) of c is a subset of the set of properties (or values) of the c' or vice-versa;
- overlaps (intersection, approximate or *intersects*), when concepts share part of the properties, but there are properties p associated to c which are not in the context of c' and vice-versa.

For Quix et al. (2006) the matching process is done through a domain (global) ontology. Thus, each concept from a local ontology is stored as a new concept in the domain ontology and an *equivalent* relation is established between this new concept and its equivalent in the domain ontology. Then, the local ontologies are matched against each other and the results are new $equivalent$ relations in the domain ontology. Finally the domain ontology "enriched" with the local ontologies' concepts and $equivalent$ relations are analyzed by a reasoner. The $equivalent$ relations which are inconsistent are eliminated.

3.2.2.2 Prototype

Quix et al. (2006) argue that for conventional features there are good matchers already developed, and for this reason they do not implement a new one. Instead, the proposed architecture can accommodate almost any existing matcher. All the other prototypes implement their own conventional features matcher.

The proposals of Klien et al. (KLIEN et al., 2004) and the GeoNis framework (STOIMENOV; DJORDJEVIC-KAJAN, 2005) go beyond the implementation of only a matching engine. They also implement services for managing the geographic ontologies. In (KLIEN et al., 2004) the service is called Concept Definition Service. In GeoNis (STOIMENOV; DJORDJEVIC-KAJAN, 2005) it is called OntoManager.

3.2.2.3 Linguistic

Regarding the linguistic features, there are basically two types of approaches: the ones which use only the words, meanings and linguistic relations defined in the ontologies to be compared or in the global ontology, and the ones which use external dictionaries.

In the first group we can cite Rodriguez and Egenhofer (2003), the GeoNis framework (STOIMENOV; DJORDJEVIC-KAJAN, 2005), the *GsPrototype* (BRODEUR; BÉDARD; MOULIN, 2005) and by the BUSTER system (VISSER; STUCKENSCHMIDT; SCHLIEDER, 2002). The former considers two types of linguistic elements in the similarity assessment: words and meanings, and synonymy and homonymy. In the BUSTER system (VISSER; STUCKENSCHMIDT; SCHLIEDER, 2002), the domain ontology is built according to the metadata from the source ontologies, and is used to define a common vocabulary.

The AgreementMaker framework (SUNNA; CRUZ, 2007) compares each concept from one ontology (source) against each concept in the other (target) ontology according to their labels definition (concept's name), as provided by an external dictionary.

3.2.2.4 Spatial features (SR)

The spatial features addressed by the ***giim*** proposals analyzed in this section are of two types: the geometry associated to a concept and the spatial relations holding between

two geographic concepts. At least the spatial relations are considered by all the proposals which address spatial features (SCHWERING; RAUBAL, 2005a,b; SOTNYKOVA et al., 2005; QUIX et al., 2006; BRODEUR; BÉDARD; MOULIN, 2005; KOKLA; KAVOURAS; TOMAI, 2005). The geometry, on the other hand, is considered by only a few approaches (BRODEUR; BÉDARD; MOULIN, 2005; SCHWERING; RAUBAL, 2005a).

3.2.2.5 Measurement procedure

The similarity evaluation metric for assessing similarity of concepts proposed in (RODRIGUEZ; EGENHOFER, 2003) is based on the set theory. For each type of distinguishing feature a specific similarity function is proposed. The matching process computes the set intersection and the set difference to determine the cardinality of the measure. Weights are used when comparing concepts based on the distance between them in their hierarchies, i.e., the farer they are, the less similar they may be. Furthermore, for each distinguishing feature, a different weight may be assigned (RODRÍGUEZ; EGENHOFER, 2004)

In (SCHWERING; RAUBAL, 2005a), the similarity measure is done using conceptual regions, by applying previously defined distance measures. Several mathematic formulas and graphs, which can have $boolean$ or $other$ scales, can be used. The $other$ scale means that a relation may have several levels of similarity, while the $boolean$ do not have any degree of existence (SCHWERING; RAUBAL, 2005b). For the similarity computation, each relation of one concept is compared separately against the relations of the concepts in the other data source. The semantic distances are calculated based on differences of standardized values for each dimension. The final values are normalized by the number of dimensions used (SCHWERING; RAUBAL, 2005b) and the results may indicate that two concepts match, are similar or do not match.

In (SUNNA; CRUZ, 2007) the mapping process is semi-automatic, which means that the affinity values associated to the concepts may be assigned as function of the child concepts, as function of the sibling concepts or from the user input. When ambiguities or inconsistencies are encountered or if the algorithm is not capable of propagating any further, the non-mapped nodes are signed out. The user has then to manually assist the algorithm by mapping concepts by hand.

3.2.2.6 Common knowledge base

The use of a mediator ontology to match geographic schemas and eliminate incorrect matchings through reasoning is proposed by Quix et al. (2006). In the framework proposed by Fonseca et al. the ontology plays the role of a model-independent system integrator (FONSECA; DAVIS; CAMARA, 2003), although there is not a unique global ontology. In the GeoNIS framework (STOIMENOV; DJORDJEVIC-KAJAN, 2005) ontologies are used as a knowledge base to solve semantic conflicts as homonyms, synonyms and taxonomic heterogeneities.

3.2.2.7 Language

Most of the ***giim*** approaches require the input sources to be described in a standardized language (OWL, RDFs, XML, etc.), but there is not a consensus on which to use. Actually, there is a wide range of languages accepted by the approaches, from pure Description Logics (DL) (SOTNYKOVA et al., 2005) and

Haskell (SCHWERING; RAUBAL, 2005b) to semi-structured ontology languages such as OIL (VISSER; STUCKENSCHMIDT; SCHLIEDER, 2002) and Ontology Web Language (OWL) (FONSECA; DAVIS; CAMARA, 2003; QUIX et al., 2006). Also the use pure XML (CRUZ; SUNNA; CHAUDHRY, 2004) and Geography Markup Language (GML) (STOIMENOV; DJORDJEVIC-KAJAN, 2005) are adopted.

3.2.2.8 Annotation

Following the assumption the elements that could contribute to the semantic definition of geographic concepts (properties, functions, axioms, and so on) are either missing or superficially described (KAVOURAS; KOKLA; TOMAI, 2005), some proposals (SUNNA; CRUZ, 2007; KAVOURAS; KOKLA; TOMAI, 2005; KUHN, 2002) use the glosses (descriptions) of the concepts to extract additional information to be used in the matching process. These information may include properties and relations, and can be obtained by applying techniques such as natural language processing (NLP).

3.2.2.9 Non-geographic context

The non-geographic context comprises three types of elements: (1) axioms, which, in general, are the taxonomic relationships, but can also be restrictions for a property in the context of a given concept; (2) properties representing conventional relationships holding between concepts; and (3) properties representing attributes associated to a concept. All ***giim*** proposals that use non-geographic features address, at least, the concept's axioms (RODRIGUEZ; EGENHOFER, 2003; SOTNYKOVA et al., 2005; QUIX et al., 2006; FONSECA; DAVIS; CAMARA, 2003; STOIMENOV; DJORDJEVIC-KAJAN, 2005; CRUZ; SUNNA; CHAUDHRY, 2004; KOKLA; KAVOURAS; TOMAI, 2005). Properties representing conventional relationships are considered in (RODRIGUEZ; EGENHOFER, 2003; SOTNYKOVA et al., 2005; QUIX et al., 2006; FONSECA; DAVIS; CAMARA, 2003; STOIMENOV; DJORDJEVIC-KAJAN, 2005; KOKLA; KAVOURAS; TOMAI, 2005; BRODEUR; BÉDARD; MOULIN, 2005). Properties representing attribute are addressed in (RODRÍGUEZ; EGENHOFER, 2004; SOTNYKOVA et al., 2005; QUIX et al., 2006; KOKLA; KAVOURAS; TOMAI, 2005; BRODEUR; BÉDARD; MOULIN, 2005).

The approach of Egenhofer et al. (2003;2004) considers also part-Of relationships and a distinguishing feature called *functions*. Functions are intended to represent what is done to or with the concept, i.e., the role of the concept.

In the approach of Kavouras et al. (2005), the elements are extracted from the concept description and classified as subclass of $genus$, which represents the concept itself or as $differentiae$, which comprises all the other properties. The set of supported properties is pre-defined and limited to some semantic properties and semantic relations extracted from external knowledge sources.

The AgreementMaker proposed by Cruz, Sunna and Chaudhry (2004) views the ontologies as hierarchical structures and only the parent-child relationships are considered. As an evolution of that proposal, Sunna and Cruz (2007) developed another method, called Sibling Similarity Contribution, in which instead of considering the parent-child relationships, similarity is based on the number of common sibling the compared concepts have.

For Brodeur et al. (2005) the context of a concept is composed by its intrinsic and extrinsic properties. The extrinsic properties are the relationships holding between two concepts, and can be conventional, spatial or temporal. The intrinsic properties, on the

other hand, are the ones which provide the literal meaning of the concept. They consist of the identification (label), attributes, geometries and temporalities of a concept.

3.2.3 Instance-level proposals

In this section we present the approaches dealing only with instance mapping/integration. Some techniques do not concern about how or where the data to be integrated/mapped are stored; the schemas of the sources are not important, only the value of the instances. Such ***giim*** proposals work under the assumption that the sources have identical schemas. For example, if the data in source *A* has a property *temperature*, the data in the other source *B* must have the same property. The matching process compares only the values for identical properties in the two sources; otherwise the comparison is not possible.

We compare 5 recent works none older than 2005. Actually, we did not find other instance-level integration or matching proposals for geographic information. As these works do not share the same vocabulary, the first thing we did was translating them into the vocabulary used of our ontology model. Table 3.4 presents the correspondences among the works vocabulary and the ontology model presented in Chapter 2. Due to space limitations, instead of the full reference in the table headers, we used an acronym, which is composed by the first letter of each author's last name.

Table 3.4: Correspondences between the reference model and the approaches vocabulary

#	Instance	Property (attribute)	Property (conv. relation)	Property (spt. relation)	Spatiality (coord)
BDKSS	object	-	-	-	location
HSLM	object	attribute	-	-	coverage
BCP	instance	-	-	topology	-
SGV	location	-	-	-	coordinates
NB	values	-	-	-	spatial extent

In addition to the general criteria listed in Section 3.2.1, more specific criteria are introduced which are applicable only to this category.

10 **Spatiality**: consideration of the spatial position (location, coordinates) of the compared instances;

11 **Non-geographic property values** (non geo prp val): use of the values for the non geographic properties, in addition to the spatial position of the instances and spatial relationship values.

12 **Geometry**: type of supported geometry (line, point, polygon);

Table 3.5 compares the instance level ***giim*** approaches according to criteria 1 to 5 and 10 to 12.

Some general tendencies can be inferred from the comparison table. In general the integration/mapping procedures do not consider non-spatial characteristics. On the other hand, differently from what occurs with schema level integration/mapping, the ***giim*** approaches for instances consider somehow the geometry of the objects, in general for obtaining the geographic coordinates and measure the location similarity.

Table 3.5: Comparative analysis of geographic instance matchers

Criterion	BDKSS	HSLM	BCP	SGV	NB
Output	Integ	Query	Mapp.	Mapp.	Integ
Prototype	X	X	X	X	X
Linguistic	X	X	N/A	$\surd$	X
SR	X	X	$\surd$	X	X
Measure. proc.	Algebra	Geometry	N/A	Algebra	Geometry
Spatiality	$\surd$	$\surd$	X	$\surd$	$\surd$
Non geo prp val	X	$\surd$	N/A	X	X
Geometry	point	polygon	all	point	raster

the $\surd$ symbol represents that the criterion is covered by the work, while the X means it is not.

3.2.3.1 Output

At the instance-level both mapping and integration approaches can be found. Beeri et al. (BDKSS) (2005) propose the integration of three ontologies with methods that can be generalized to any number of ontologies. Two approaches are presented, one consists of simply splitting the three-ontology query into two two-ontology queries and apply them sequentially. The other is the query processing over the three ontologies simultaneously. The result of the algorithm proposed by Navarrete and Blat (NB) (2007) is a new, integrated, ontology that contains the concepts from both source and target and their instances as well. The concepts are organized in a taxonomy which is generated from the relations provided by the algorithm. The output produced by the approaches of Sehgal et al. (SGV) (2006) and of Belussi et al. (BCP) (2005) is the mapping between two ontologies. Finally, the approach of Hariharan et al. (HSLM) (2005) does not result in an explicit integration or mapping between the two sources. Actually, the approach was conceived for query answering.

3.2.3.2 Linguistic

Only Sehgal, Getoor and Viechnicki (2006) consider linguistic features in the matching process. They define a geographic instance as a set of features $l = [locationname, spatialcoordinates, locationtype]$, where the $locationname$ is the name of the instance and $locationtype$ is the concept the instance belongs to. For the location name matching the authors use some string based metrics, which means that two instances have their names compared not only in terms of equality, but in terms of similarity. As the authors state, because some location types may be defined using different vocabularies, it is important to establish equivalences between the location types. These equivalences may be found in auxiliary synonym sets or dictionaries.

3.2.3.3 Spatial features (SR)

Identifying topological similarities of multiresolution graphical maps is proposed by Belussi, Catania and Podesta (2005). They state that the geometry consistency of the instances to be integrated can be reduced to an equality test between two geometric map representations. Furthermore, topological representations are more abstract than geometric

representations and describe properties that are preserved after the dimension of an object changes. The topological relations addressed are the ones from the 9-intersection model (EGENHOFER; FRANZOSA, 1991), i.e., *disjoint, touches, contains, inside, equal, overlaps, covers, coveredBy* and *relation*.

3.2.3.4 Measurement procedure

At the instance-level, two different types of measures can be used. When addressing the instances position, some geometrical functions, such as overlay, are applied. The overlay may be exact, as in (NAVARRETE; BLAT, 2007), or approximate, as in (HARIHARAN et al., 2005). When considering the non-spatial features, some metrics also used for conventional data can be used. Moreover, some of the proposals use weights to combine the different measures (SEHGAL; GETOOR; VIECHNICKI, 2006; HARIHARAN et al., 2005).

Hariharan et al. (2005) propose approximate algorithms for answering spatial queries that require the integration of information provided by different sources based on the local analysis of the query region using space partitioning techniques. The spatial coverage of an ontology is measured by dividing the area of the query q covered by the ontology O_i by the area of the query q. The final rank for an ontology, which corresponds to the level of similarity between its instances and the query, is a weighted sum which combines both the spatial coverage and the information content based on the non-spatial attributes.

In (BELUSSI; CATANIA; PODESTÀ, 2005) for each one of the topological relationships the authors created a 3x3 matrix denoting the combination of geometries it supports. Based on the intersection of these matrices a table with the distances between the topological relationships according to the involved geometries was created. The lower the distance, the more similar are the relationships.

In (SEHGAL; GETOOR; VIECHNICKI, 2006) the location type is used only to remove incorrect matchings due to the comparison of non-equivalent concepts. For the integration of location name two approaches are proposed by the authors. The first one is to use one of the measures as a threshold and the other as a filter. The second solution is to consider both measures at the same time, but giving weights for each one. To define the weights they use a machine learning approach, by training the matcher with a set of pairs (a, b) where the correct matches as well as the incorrect ones are indicated to the matcher.

In the proposal of Navarrete and Blat (2007) the similarity assessment is based on the overlay of the two regions to be compared. Basically, what they do is to compare each region value from the source against each region from the target symmetrically, and based on the result it is established what they call semantic relation. These relations are, actually, taxonomic relations and can assume the values subclass/superclass of and equivalent to.

3.2.3.5 Spatiality

The ***giim*** approaches which deal with the spatial position of the instances (SEHGAL; GETOOR; VIECHNICKI, 2006; BEERI et al., 2005) usually consider only the point geometry, in order to reduce the coordinates to be compared to only one pair (x, y). Only in (HARIHARAN et al., 2005) the supported geometry is a region, which correspond to the minimum bounding rectangle (MBR) of the objects. The proposal of Navarrete and Blat (2007) does not consider vector objects (points, lines or polygons, for instance). Instead, the supported geometries must be described as a grid of cells, which is a raster format.

In Sehgal, Getoor and Viechnicki (2006) the similarity measure regarding the spatial

coordinates is defined as the inverse of the coordinates distance. Beeri et al. (2005) propose the integration of two or more ontologies through join operations by considering only the location, i.e., the spatial position of the objects.

The approach of Hariharan et al. (2005) is as follows. Given a query, all the possible ontologies are searched and ranked, based on how much of the query each one covers (spatial coverage). The answer is given by only one ontology which is the one that better answers the query (best ranked). If the answer is not fully satisfying, the query can be broken into sub-queries, and each one is submitted for all ontologies to rank the answers. The queries can be subdivided until the user is satisfied. The process of query division may chose a different ontology to answer each query depending on the area of the query.

3.2.3.6 Non-geographic property values

Regarding the context of an instance, the approach of (HARIHARAN et al., 2005) uses the attribute values to form a keyword, which is used to measure the information content of the region in respect to the query. This measure is estimated based on an extended version of the TF-IDF model used in information retrieval.

3.2.4 Hybrid proposals

In this section, we consider the approaches addressing the integration/mapping of geographic schemas and instances as well. Some of them start from the schema matching and then perform the instance matching. Others make the inverse, starting from the mapping established at the instance level, then build clusters of similar instances and mapping at the concept level are inferred. Because the features addressed by these approaches are basically the same as the ones presented in Sections 3.2.3 and 3.2.2, the criteria are the same. In addition, we consider the *Metadata* criterion, regarding the capability of the approach to consider the metadata associated with a geographic instance (e.g., the scale or the coordinate reference system);

We compare four works we found dealing with concept as well as instance-levels. As these works do not share the same vocabulary, the first thing we did was translating them into the vocabulary used of our ontology model. Table 3.6 presents the correspondences among the works vocabulary and the ontology model presented in Chapter 2. Due to space limitations, instead of the full reference in the table headers, we used an acronym, which is composed by the first letter of each author's last name.

Table 3.6: Correspondences between the ontology model and the ***giim*** approaches vocabulary

#	Concept	Instance	Property (attribute)	Property (conv. rel.)	Property (spt. rel.)	Axioms	Spatiality (coord)
DHD	concept	instance	property	-	-	IS-A	-
MBL	concept	instance	-	-	-	-	geom. prp.
VOLZ	concept	instance	thematic	thematic	topologic	thematic	geometric
DW	concept	object	thematic	-	-	IS-A	location

Table 3.7 compares the combined (schema and instance levels) ***giim*** approaches according to all tge evaluation criteria. Hybrid (combined) approaches:

Table 3.7: Comparative analysis of geographic combined matchers

Criterion	DHD	MBL	VOLZ	DW
Output	Integ	Integ	Mapp.	Integ
Prototype	√	√	√	√
Linguistic	√	√	X	√
SR	X	X	√	X
Measure. proc.	Formal	Algebra	Algebra	Formal
CKB	√	√	X	X
Language	DL	GML	XML	N/A
Annotation	√	X	X	X
Non geo context	a,c	N/A	X	c
Spatiality	X	√	√	√
Non geo prp val	√	X	X	X
Geometry	X	all	line/point	polygon
Metadata	√	X	X	X

a means attributes, *b* means relationships and *c* means axioms.
the $\surd$ symbol represents that the criterion is covered by the work, while the X means it is not.
DL means Description Logics, GML is the Geography Markup Language and XML is the eXtensible Markup Language.

As could be expected, the approaches kind of combine the features from both the concept and instance level integration/mapping. One fact to be highlighted is the fact that the hybrid approaches do not consider any kind of relationships (conventional or spatial). Furthermore, the use of geometries and spatial location is not a consensus.

3.2.4.1 Output

The integration of the two compared geographic ontologies is the result of Manoah et al. (MBL) (2004), Duckham and Worboys (DW) (2005) and Dobre et al. (DHD) (2003). In these ***giim*** proposals a global, integrated geographic ontology is built from the input local ontologies. Volz (VOLZ) (2005), on the other hand, delivers the mapping between two geographic ontologies.

When comparing two concepts, the output is one of four levels of similarity (DOBRE; HAKIMPOUR; DITTRICH, 2003; VOLZ, 2005):

- Equality, when two concepts $a \in A$ and $b \in B$ have the same intensional definitions;
- Specialization, when a concept $a \in A$ is a subconcept of $b \in B$, and the conjunction of both is exactly equal to a's definition;
- Overlapping, when the definition of $a \in A$ and $b \in B$ are in part equivalent, but the conjunction of both is not the definition of a nor b;
- Disjunction, otherwise.

3.2.4.2 *Prototype*

All ***giim*** proposals for combined techniques and approaches were implemented as, at least, prototype tools. However, none of them are freely available.

3.2.4.3 *Linguistic*

Three out of the four proposals for combined matchers consider linguistic features in the mapping/integration process (DOBRE; HAKIMPOUR; DITTRICH, 2003; WORBOYS; DUCKHAM, 2002; MANOAH; BOUCELMA; LASSOUED, 2004). The linguistic features are used at both the concept-level and the instance-level.

Manoah, Boucelma and Lassoued's (2004) approach is based on machine learning and one of the features considered for the similarity measurement is the instance name. In the proposal there is learner for matching the instances names. It exploits textual information, receiving as input the instance and the name of the property which identifies the instance. Then it identifies the different values for this property.

3.2.4.4 *Spatial features (SR)*

Volz (2005) addresses topological and geometrical features. The geometries are limited to lines and points and the instances have to be in the same (or very similar) scale. For the geometric aspects the similarity is measured in terms of length and angle, while for the topological in terms of adjacency relations.

3.2.4.5 *Measurement procedure*

Comparing the geographic ontologies at the concept-level and based on the results matching the instances is the approach adopted in (DUCKHAM; WORBOYS, 2005). Matching the concepts based on the result of the comparison of the instances is proposed in (VOLZ, 2005; DOBRE; HAKIMPOUR; DITTRICH, 2003; MANOAH; BOUCELMA; LASSOUED, 2004).

Regarding the final similarity value between two concepts or two instances, Volz (2005) produces measures within [0,1]. Furthermore, in both approaches the final similarity value is given by a weighted sum of the considered features. This weights are defined by the user.

During the integration procedure proposed by Worboys and Duckham (2002) the global ontology may be enriched by adding some intermediate concepts in its taxonomy. Furthermore, if a concept c is defined in both ontologies, but in ontology O as a superclass of a concept c_x and in ontology $O^{'}$ as a superclass of a concept $c_y^{'}$, and c_x and $c_y^{'}$ are not equivalent, not always c_x and $c_y^{'}$ are considered siblings. If one is more generic than the other, maybe a parent-child relationship is created between them.

3.2.4.6 *Common knowledge base*

In (HAKIMPOUR; GEPPERT, 2002; DOBRE; HAKIMPOUR; DITTRICH, 2003) the authors use a global ontology to map the concepts and instances from the local ontologies and then integrate the local information. This global ontology is created in a two-phase procedure. First only the concepts hierarchy is created, in a top-down way. One concept is created in the global ontology for each concept of the local ontologies, unless there is an equality relation. Furthermore, maybe new concepts have to be created to accommodate overlapping or disjoint concepts in the hierarchy (HAKIMPOUR; GEPPERT, 2002). The second phase consists of adding the attributes, as binary relations, to

the concepts.

The use of a global (or domain) ontology as a mediator is also proposed by Worboys and Duckham (2002;2005). The elements from the local ontologies are matched with the elements from the global ontology and then the similarity between the local ontologies can be inferred. The global ontology may also be enriched with concepts, properties and instances from the local ontologies.

3.2.4.7 Language

As it happens in the concept-level proposals, the use of a standardized language to describe the ontologies to be integrated/mapped is adopted by the majorities of the combined ***giim*** proposals. Once again, however, there is not a consensus about the language to be used. The use of semi-structured languages, such as XML (VOLZ, 2005) and GML (MANOAH; BOUCELMA; LASSOUED, 2004) in more recent works can be seen as a tendency. Description logics, on the other hand, is used in (DOBRE; HAKIMPOUR; DITTRICH, 2003).

3.2.4.8 Annotation

Hakimpour et al.'s (2003) is the only work which uses the description of the concepts and instances in the mapping process.

3.2.4.9 Non-geographic context

Regarding contextual features used in the matching process, at least the taxonomies are used in the majority of ***giim*** approaches. The context features are addressed by Hakimpour et al. (2003) in two different processes: the entity mapping, i.e., concepts hierarchies, and attribute mapping. In the framework proposed by Worboys and Duckham (2002;2005) the hierarchies of the concepts are the starting point for the integration process.

3.2.4.10 Spatiality

Worboys and Duckham's (2002;2005) approach takes into consideration the spatial location of the instances to be integrated and executes a product operation which actually gives as result an area containing the intersection of the instances.

Manoah, Boucelma and Lassoued (2004) state that different geographic concepts have often geometric properties that can be used to distinguish them from each other. The geo matcher receives a set of previously calculated geometric properties and matches the instances according to their geometries.

3.2.4.11 Non-geographic property values

Attribute values are used when matching instances in (DOBRE; HAKIMPOUR; DITTRICH, 2003; HESS; IOCHPE; CASTANO, 2006). In the former those are the only non-geographic values considered in the process.

3.2.4.12 Metadata

Hakimpour et al. (2003) also consider some metadata in the mapping process.

3.3 Summary

In this chapter we presented the state-of-the-art in the field of geographic ontology matching. We presented works addressing the concept-level only, the instance-level only and the combined proposals as well. As can be conclude by the analysis of Tables 3.2 and 3.5 at the concept-level the works in general address half of the features we considered in the comparison criteria set, specially regarding the specific geographical features, such as spatial relationships. On the other hand, the combined approaches address less features at the concept-level. They do not address, in general, the non-spatial properties and only one work (VOLZ, 2005) considers the spatial relationships.

At the instance-level, by analyzing Table 3.5, we can notice that the works consider the spatial position of the instances as basically the only feature in the matching process. Only one work (HARIHARAN et al., 2005) deals with the property values and also only one work (SEHGAL; GETOOR; VIECHNICKI, 2006) uses linguistic features in the matching process. At the instance-level, the combined matchers are quite similar to the instance-level matchers.

In the next two chapters we present our proposal for a geographic ontology matcher. At the concept-level (Chapter 4) we try to address all the features addressed by the concept-level matchers, i.e., all the criteria used in the comparison. The instance-level part of the matcher (Chapter 5) tries to go beyond the existing instance-level works by addressing property values, metadata values and linguistic features as well.

4 PROPOSING A NEW CONCEPT-LEVEL MATCHING APPROACH

This chapter presents our contribution of an algorithm and a set of metrics for the matching process of geographic ontologies at the concept-level. Furthermore, it reports some tests results we have already obtained and compares them with the results produced by other existing matchers.

Geographic ontologies obviously describe geographic concepts. However, also some non-geographic concepts may be described, associated to the geographic ones. Therefore, differently from the existing works dealing with geographic ontology matching, mapping and integration presented in Chapter 3, our algorithm is capable of matching geographic concepts and conventional, non-geographic, concepts as well. Since we use the geographic ontology model introduced in Chapter 2, it is easy to find out if a concept is either geographic or conventional. In the case of a geographic domain concept it holds a `hasGeometry` object type property, with a geometry concept as range. In the case of a conventional concept, this property is absent from the concept's context.

4.1 Algorithm

As Figure 4.1, shows the concept-level matching is a two-step algorithm. Firstly the similarity is measured in terms of the concept's name and attributes, and a partial similarity is retrieved. Based on that partial measure the similarity regarding the other features is assessed.

Each geographic domain concept gc from the ontology O is compared against each geographic domain concept $gc^{'}$ from the ontology $O^{'}$. Analogously, each (conventional) domain concept c from the ontology O is compared against each (conventional) domain concept $c^{'}$ from the ontology $O^{'}$. The features and metrics presented in section 4.2 are executed and compose a balanced sum for the overall similarity. If this similarity value is lower than a certain threshold, the pair (gc,$gc^{'}$) or (c,$c^{'}$) is excluded from the possible matching pairs. All the pairs (gc,$gc^{'}$) or (c,$c^{'}$) with similarity value higher than that threshold are presented to the user in a ranked list and he/she decides for the correct matchings.

;

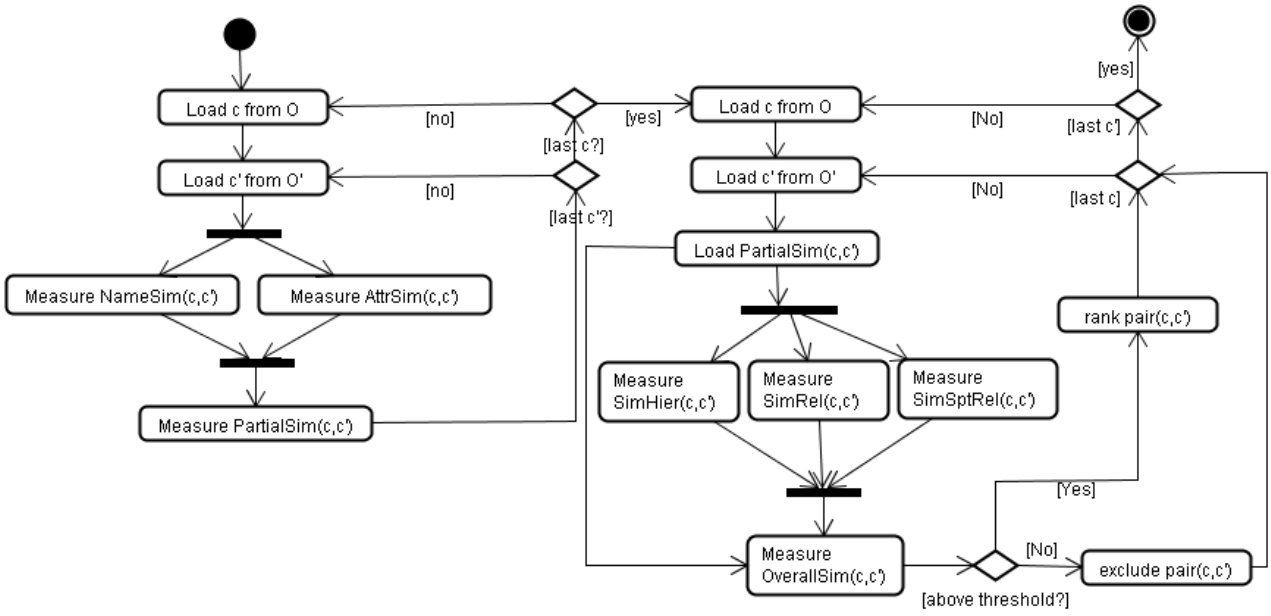

Figure 4.1: UML activity diagram for the concept matching

The encoding presented in Figure 4.2 summarizes the algorithm execution:

```
/*
  INPUTS:
    Concepts from two ontologies, Ont1 and Ont2
    The weights WN, WA, WH, WR and WS
    The threshold for accepting / eliminating pairs of concepts
*/
/*
  Vectors for geographic concepts (GC) and conventional
  concept (CC) for ontologies Ont1 and Ont2
*/
Vector vOnt1GC = new Vector();
Vector vOnt2GC = new Vector();
Vector vOnt1CC = new Vector();
Vector vOnt2CC = new Vector();
for each concept c from Ont1 {
  if hasProperty(c,hasGeometry))
    vOnt1GC.add(c);
  else
    vOnt1CC.add(c);
}
for each concept c from Ont2 {
  if hasProperty(c,hasGeometry))
    vOnt2GC.add(c);
  else
    vOnt2CC.add(c);
}
for each concept c from vOnt1GC {
  for each concept c' from vOnt2GC {
    simNameVal = SimName(c,c');
    simAttrVal = simAttr(c,c');
    if((WN*SimNameVal + WA*simAttrVal)/(WN+WA)>threshold){
      simHierVal = SimHier(c,c');
      simRelVal = SimRel(c,c');
      SimSptVal = SimSpt(c,c');
      if ((WN*SimNameVal + WA*simAttrVal + WH*SimHierVal +
           WR*simRelVal + WS*simSptVal) > threshold)
        store(c,c');
    }
  }
}
for each concept c from vOnt1CC {
  for each concept c' from vOnt2CC {
    simNameVal = SimName(c,c');
    simAttrVal = simAttr(c,c');
    if((WN*SimNameVal + WA*simAttrVal)/(WN+WA)>threshold){
      simHierVal = SimHier(c,c');
      simRelVal = SimRel(c,c');
      if (((WN*SimNameVal + WA*simAttrVal + WH*SimHierVal +
           WR*simRelVal)/WN+WH+WA+WR) > threshold)
        store (c,c');
    }
  }
}
```

Figure 4.2: pseudo-code for the concept matching algorithm

4.2 Metrics

As explained in section 2.3.1, the heterogeneities between two concepts $c \in O$ and $c^{'} \in O^{'}$ may occur regarding:

- Term used as concept name - heterogeneity NH.
- Hierarchy (set of concept's superclasses) - heterogeneity HH.
- Data type properties (attributes) - heterogeneity AH.
- Object type properties (relationships) playing the role of a conventional relationship - heterogeneity RH.
- Object type properties (relationships) playing the role of a spatial relationship - heterogeneities DH for directional relationships and TH for topological relationships. Together they form the spatial relationship SH.
- Object type properties (relationships) playing the role of the geometry of the geographic concept - heterogeneity GH.

4.2.1 Name similarity

To measure the similarity between the terms used as concept names there are three possibilities, in the following order.

I. Verify if the term $t(c)$ of the concept $c \in O$ coincides with the term $t(c^{'})$ of the concept $c^{'} \in O^{'}$

II. Search in an external dictionary or thesaurus, i.e., WordNet the level of linguistic affinity between the terms $t(c)$ for the concept $c \in O$ and the term $t(c^{'})$ for the concept $c^{'} \in O^{'}$. We call this the $SYN(c, c^{'})$ function. This function returns a value within [0,1], where 1 means the terms are synonyms and 0 means they are not related at all.

III. Use a string comparison metric to measure the similarity between the labels $t(c)$ of the concept $c \in O$ and $t(c^{'})$ of the concept $c^{'} \in O^{'}$. In this work we adapt the Stoilos et al. (STOILOS; STAMOU; KOLLIAS, 2005) metric, which considers all the common substrings that the two compared strings share and also the JaroWinkler metric.

$$SimName(c, c^{'}) = \frac{2 * \frac{length(max(ComSubstring(t(c),t(c^{'}))))}{length(t(c))+length(t(c^{'}))} + JaroWinkler(t(c), t(c^{'}))}{2} \tag{4.1}$$

Step II is executed only if the step I returns 0, i.e., if $t(c) \neq t(c^{'})$. Step III is executed only if the step II does not return a satisfactory value (the threshold has yet to be defined).

The similarity measure between the terms which nominate the concepts returns always a value within [0,1] where 0 means the terms are completely different and 1 that they are exactly equals (or synonyms).

4.2.2 Property similarity

As the property heterogeneity is classified in attribute (AH), relationship (RH), geometric (GH) and spatial relationship (SH, divided into DH and TH), we separately measure the similarity for each one of these aspects.

Attribute similarity: To measure the similarity between an attribute $a(t(p), dtp) \in \{p(c)\}$ in an ontology O and an attribute $a(t(p^{'}), dtp^{'}) \in \{p(c^{'})\}$ in an ontology $O^{'}$ the two components to be analyzed are the attributes' names and data types.

The similarity regarding the attributes' name $t(p)$ is measured in a similar way to the one used to concept names. The main difference is that only the steps I and II are performed. This means that first is checked if the attributes' names $t(p)$ and $t(p^{'})$ are equal and in case they are not the linguistic affinity is calculated.

$$SimNAt(a(t(p), dtp), a(t(p^{'}), dtp^{'})) = \quad \begin{cases} 1 & \text{if } t(p) = t(p)^{'} \\ SYN(t(p), t(p^{'})) & \text{otherwise} \end{cases}$$

The similarity of data types is measured by checking if the data types are the same ($dtp = dtp^{'}$, such as both integer or string) or if one is a subclass of the other ($dtp \subseteq dtp^{'}$ or $dtp \supseteq dtp^{'}$, such as float and integer).

$$SimDAt(a(t(p), dtp), a(t(p^{'}), dtp^{'})) = \quad \begin{cases} 1 & \text{if } (dtp \subseteq dtp^{'}) \vee (dtp \supseteq dtp^{'}) \\ 0 & \text{otherwise} \end{cases}$$

Each attribute has an associated weight ε, which corresponds to its relevance to the concept. ε is given by

$$\varepsilon = 1 - (min((\frac{Ca - 1}{C})(\frac{Ca^{'} - 1}{C'}))) \tag{4.2}$$

which means that the lower the number of concepts having an attribute, the more relevant the attribute is. In the equation, Ca is the number of concepts having the attribute a and C is the total number of concepts of the ontology.

The final measure of the attribute similarity is given by

$$SimAt(c, c^{'}) = \frac{\sum max((\delta * SimNAt(a_i, a_j) + (1 - \delta) * SimDAt(a_i, a_j)) * \varepsilon)}{|At(c) \cup At(c^{'})|} \tag{4.3}$$

where δ is the weight for the attribute name similarity. $At(c)$ is the subset of P which contains only attributes (data type properties).

Conventional relationships similarity: To measure the similarity of the conventional relationships, two components that determine the relationship heterogeneity $RH(c, c)$ have to be considered: (1) the concepts c_x and $c_x^{'}$ associated, respectively, with c and $c^{'}$ and, (2) the relationship cardinality. The name of the property that defines the association cannot be used because it may not be semantically relevant, since each ontology may use different labels to express the same relationship. The conventional relationship similarity between two concepts $c \in O$ and $c^{'} \in O^{'}$ is given by:

$$SimRel(c, c^{'}) = \tag{4.4}$$
$$\frac{\sum cr(t(p), t(c_x), minCard(p), maxCard(p)) \cap cr(t(p^{'}), t(c_x^{'}), minCard(p), maxCard(p^{'}))}{|CR(c) \cup CR(c^{'})|}$$

where $CR(c)$ and $CR(c^{'})$ are, respectively, the subset of properties from $\{p(c)\}$ and $\{p(c^{'})\}$ which correspond to conventional relationships involving, respectively, c and $c^{'}$.

The computation of the conventional relationship similarity is based on the results obtained by the similarity name ($SimName(c, c^{'})$) measurement. This is due the necessity of determining if the concepts c_x and $c_x^{'}$ are equivalent. If c is associated to a concept c_x and $c^{'}$ is associated to a concept $c_x^{'}$ and the name similarity $SimName(c_x, c_x^{'})$ is higher than a certain threshold the relationships are considered as equivalent, if the cardinalities are also equal.

Geometric similarity: Because of the possibility of having the same phenomenon described using different spatial representations, in our algorithms we do not compare directly the geometry of the compared concepts c and $c^{'}$. Instead, the geometry is used in the spatial relationship similarity measure.

Spatial relationships similarity: To measure the similarity between two concepts regarding the spatial relationships, the three components which cause the spatial heterogeneity must be considered: the concepts gc_x and $gc_x^{'}$ associated, respectively, to gc and $gc^{'}$, the cardinalities of the relationships, and the names $t(p)$ and $t(p^{'})$ of the relationships. The name of the association cannot be ignored because for the spatial relations the names are, in general, standardized and semantically relevant (EGENHOFER; FRANZOSA, 1991; FRANK, 1992). For example, although River $crosses$ City and River $inside$ City involve the same concepts, they do not mean the same. The spatial relations considered by our algorithms are the directional and the topological. We do not measure the similarity of the metric relationships because in general they are calculated by a GIS and not stored in the ontology.

In the case of directional relationships the geometry is not relevant, because the relationships do not depend on the geometric shapes but rather on the spatial coordinates. As at the concept-level the coordinates are not defined, the directional relationship is measured in terms of the restrictions of the concepts. For example, on the definition of a concept EuropeanCountry there may be a restriction that says that is must be *At_north_of* a concept AfricanCountry.

However, we cannot simply compare if $gc_x = gc_x^{'}$ and $t(p) = t(p^{'})$ when dealing with topological relationships. Because of the multi-representation possibility (i.e., the same data represented using different geometries in two datasets), there may be equivalent combinations of $(geometry, spatialRelation)$. A deep study on this issue is presented in (BELUSSI; CATANIA; PODESTÀ, 2005), in which the authors define the equivalences of the topological relationships depending on the geometries of the involved data. Based on that we developed a boolean function $eqTop(top_A, top_B)$, where top_A and top_B are two objects representing topological relationships. A top object is a triple of type $top = (tr_{name}, geom_a, geom_b)$, where tr_{name} is the name of the topological relationship, and $geom_a$ and $geom_b$ are the geometries involved in that relationship (taken from i and i_x). $eqTop(top_A, top_B)$ returns "true" if top_A and top_B are semantically equivalent.

Based on the statements above, we measure the spatial relationship similarity as

$$SimSpt(gc, gc^{'}) = \frac{\sum sr(t(p), t(gc_x), minCard(p), maxCard(p)) \cap sr(t(p)^{'}, t(gc_x^{'}), minCard(p^{'}), maxCard(p^{'}))}{|SR(c) \cup SR(c^{'})|} \quad (4.5)$$

where $SR(c)$ and $SR(c^{'})$ are, respectively, the subsets of $\{p(c)\}$ and $\{p(c^{'})\}$ that contain the spatial relationships. For the topological relationships two spatial relationships are

said to be equivalent when the $eqTop(top_A, top_B)$ returns true and the associated concepts gc_x and $gc_x^{'}$ are considered as equivalents by the partial similarity measure.

The computation of the spatial relationship similarity is based on the results obtained by the similarity name ($SimName(c, c^{'})$) measure. This is due to the necessity of determining if the concepts c_x and $c_x^{'}$ are equivalent. If c is associated to a concept c_x and $c^{'}$ is associated to a concept $c_x^{'}$ and the name similarity $SimName(c_x, c_x^{'})$ is higher than a certain threshold (to be defined, e.g., 0.8) the relationships are considered as equivalent, if the other components (cardinality and relationship names) are also equivalent.

4.2.3 Hierarchy similarity

As defined in Section 2.3.1.5, two concepts have hierarchy heterogeneity when there are differences in their respective concept hieararchies. The superclass similarity is then given by the number of equivalent superclasses of the concepts c and $c^{'}$ divided by the total number of superclasses of both concepts, as follows.

$$SimHier(c, c^{'}) = \frac{\sum(h(c, c_x) \cap h(c^{'}, c_x^{'})) * \psi}{|H(c) \cup H(c^{'})|} \tag{4.6}$$

where ψ is the difference of the superclasses level. If both classes c_x and $c_x^{'}$ are at the same distance from the concepts c and $c^{'}$, respectively, ψ is equal to 1. Otherwise, ψ is decreased. $H(c)$ is the number of superclasses of the concept c, direct or indirect. The similarity measure is a value within [0,1].

4.2.4 Overall similarity

The final value for the similarity between two concepts $c \in O$ and $c^{'} \in O$ is a weighted sum which considers all the similarities detailed in the previous subsections.

$$\begin{aligned} &Sim(c, c^{'}) = WN * SimName(c, c^{'}) + WA * SimAt(c, c^{'}) + \\ &WH * SimHier(c, c^{'}) + WR * SimRel(c, c^{'}) + WS * SimSpt(c, c^{'}) \end{aligned} \tag{4.7}$$

where WN, WA, WH, WR and WS are, respectively, the weights for the name, attributes, hierarchy, conventional relationships and spatial relationships similarities. The sum of these weights must be 1, and thus the value of $Sim(c, c^{'})$ lies within [0,1].

We tried to empirically optimize the values of WN, WA, WH, WR and WS. However, we could not find an ideal combination for these parameters, because it depends on how the input ontologies are structured. If the taxonomies are deep, then WH may have a higher influence than if the concepts are organized in a few-levels hierarchy. The same occurs for the properties. The more properties of a given type (i.e., attributes, conventional relationships and spatial relationships) exist, the higher should be the weight for that kind of property. Therefore, the (semi-)automatic combination of these parameters to achieve best results is yet an open issue.

4.3 Testing the proposal

In this section we report some test results obtained by executing the concept matcher algorithm. The goals are to evaluate the adequacy of the produced matchings in respect of

what humans consider as matching concepts as well as to compare them against the ones obtained by executing other existing matchers.

Unfortunately none of the existing proposals for concept-level geographic information matchers presented in Chapter 3 has neither an implementation or prototype available for downloading nor the similarity expressions described. Therefore, it was not possible to run the tests with any of them. H-MATCH (CASTANO; FERRARA; MONTANELLI, 2006) and Prompt (NOY, 2004) were the only tools we could use, even if they are not designed to cope with the particular features of geographic ontologies. Although in (HESS; IOCHPE; CASTANO, 2006, 2007b) H-MATCH was partially extended to deal with geographic ontologies, in the tests we used the original version of H-MATCH .

4.3.1 Test C1: Example ontologies

In the first test we used as inputs the ontologies presented in Figures 2.2 and 2.5. To evaluate the results produced by the matchers, we asked two humans to manually determine what they considered to be the equivalent concepts between the two ontologies. Then we compared the matchings with the ones produces by the matchers. Table 4.1 presents the results produced by the human matching, while Table 4.2 shows the matchings produced by the (semi-)automatic matchers. The concepts representing geometries are not reported, because the goal is to match domain concepts.

Table 4.1: Equivalences defined by human matching

Concept from O	Equiv. cpt in O' for human 1	Equiv. cpt in O' for human 2
Road	Route	Route
Avenue	-	-
Street	-	-
IndustrialDistrict	IndustrialDistrict	IndustrialDistrict
Factory	Factory	Factory
Building	-	-
Hospital	Hospital	Hospital
Temple	Temple	-
School	-	-

By observing tables 4.1 and 4.2, we can notice that both human users considered the concepts *Road* from O and *Route* from O' as equivalent, but only our algorithm was capable of identifying this equivalence. This can be explained because in both ontologies the concepts have a topological relationship with the concept *IndustrialDistrict*. The name of the relationship is different in both ontologies, but semantically equivalent. Furthermore, as we use an external dictionary to lookup for synonyms, the linguistic equivalence between *Route* and *Road* returned 1.

The most equivalent concept in O' for the concept *Temple* from O was considered *Synagogue* by H-MATCH and in our algorithm as well. This can also be explained because the words are synonyms in the external dictionary, and also by the taxonomies of the two ontologies.

In summary, if we consider only the common matches for both human users, 80% (4 out of 5) (precision) of the matches found by our algorithm were also matches for the

Table 4.2: Equivalences found by the matchers

Concept from O	Equiv. cpt in O' H-MATCH	Equiv. cpt in O' prompt	Equiv. cpt in O' our algorithm
Road	-	-	Route
Avenue	-	-	-
Street	-	-	-
IndustrialDistrict	IndustrialDistrict	IndustrialDistrict	IndustrialDistrict
Factory	Factory	Factory	Factory
Building	-	-	-
Hospital	-	Hospital	Hospital
Temple	Synagogue	Temple	Synagogue
School	-	-	-

human users, and 100% of the matches produced by them were found by the proposed algorithm (recall). For H-MATCH the precision was 66% (2 out of 3) while the recall was 50% (2 out of 4). Finally, for the Prompt matcher, the precision was 75% (3 out of 4) and the recall was also 75% (3 out of 4). This numbers are graphically expressed in Figure 4.3. In the graphic, for space reason, we already refer to the algorithm as IG-MATCH.

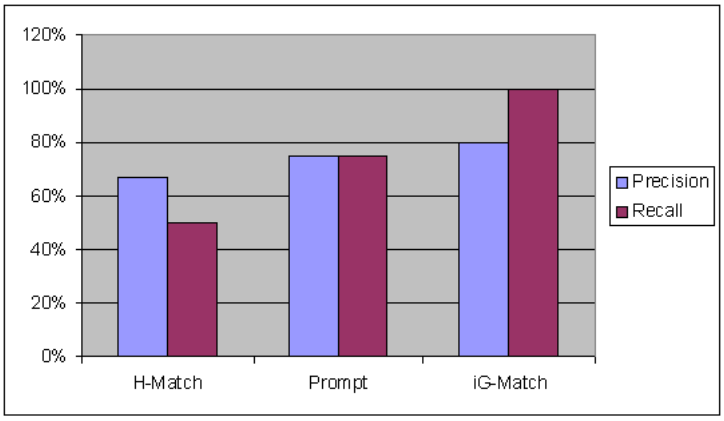

Figure 4.3: Recall and Precision of the (semi-)automatic matchers

4.3.2 Test C2: Ontologies designed by GIS Experts

In the second test we asked two people with large experience in designing geographic conceptual schemas to create an ontology describing sightseeing places. Besides maintaining the specific domain, they were asked to:

- describe the concepts in a taxonomy;
- be OGC compliant, i.e., to every geographic domain concept they had to relate a geometry concept of type `Point`, `Line` or `Polygon` using the `hasGeometry` object type property;
- every specialization of a geometry concept, a data type property `hasLocation` should be added to its context.

We did not give more information or instructions because we did not want to induce them in the design of the ontology. The original email sent to them is presented in the Annex 8.1.

The produced ontologies were very different from one another. The first one, presented in Figure 2 of Annex 8.1, was very generic. Besides the geometric concepts, only 4 domain concepts were defined: `Locality`, to represent a place, `LocalityType`, to represent type of places, `TouristicAttraction`, to model a sightseeing and `TouristicAttractionType`, to represent type of sightseeing. `TouristicAttraction` and `Locality` were geographic domain concepts.

The second ontology, presented in Figure 1 of Annex 8.1, was much more detailed, with 22 concepts, besides the ones representing geometries. All of them were geographic domain concepts.

We then submitted the two ontologies to three other people who conducted a manual matching. Only two pairs of concepts were found as equivalent: *TouristicAttraction,TouristAttraction* and *Locality,Place*. The results obtained by executing the matching with our algorithm, Prompt and H-MATCH are presented in Table 4.3. In the cells corresponding to our algorithm we put the retrieved similarity measure, while for the other matchers *YES* means the pair of concepts was found and *NO* means the pair of concepts was not found, since this is their output.

Table 4.3: Equivalences found by the matchers for test C2

Pair(c,c')	H-MATCH Similarity	Prompt Similarity	Our algorithm Similarity
(TouristAttraction,TouristAttraction)	YES	YES	83%
(Locality, Place)	NO	NO	60%
(Locality, Country)	NO	NO	55%

Once again, our algorithm showed to achieve better results than the results obtained by the conventional matchers. Although we found a non-matching pair (Locality, Country) as a possible matching, the correct pair (Locality, Place) was also retrieved, with higher similarity. The conventional matcher did not manage to obtain that pair.

4.3.3 Test C3: Ontologies downloaded from the Web

In the third test we downloaded from the Internet two tourism ontologies:

1. **e-tourism** (http://e-tourism.deri.at/ont/e-tourism.owl) from Innsbruck, Austria. It is composed by 19 concepts, 12 object type properties and 78 data type properties. This ontology is graphically presented in Figure 4 of Annex 8.1;

2. **andalucia-tourism** (http://mobi.yaco.es/andalucia.rdf/andalucia-tourism.owl), from Spain (it was originally in Spanish and we translated it into English). It is composed by 15 concepts, 7 object type properties and 33 data type properties. This ontology is graphically presented in Figure 3 of Annex 8.1.

The **e-tourism** ontology had properties and concepts describing geographic information. However, it was not OGC compliant. It had a concept labeled `Location`,

with two child-concepts, namely PostalAddress and GPSCoordinates. The latter had two data type properties: hasLatitude and hasLongitude. We therefore created the concept Geometry and a child-concept Point to accommodate the ontology into our ontology model. The hasLocation data type property was created to represent the same information held by a pair (hasLatitude, hasLongitude) of the GPSCoordinates concept. Furthermore, a hasGeometry object type property was created associating a concept with the Point concept. As all the instances of geographic concepts were related to instances of the GPSCoordinates having only one pair (hasLatitude, hasLongitude), it was not necessary to specialize the concept Geometry in further concepts.

The **andalucia-tourism** ontology did not have explicit concepts or properties describing geometries or spatial characteristics. We created the Geometry concept and the child-concepts Point, Line, and Polygon with the hasLocation data type property. Analyzing concept by concept we created the hasGeometry property and associated it to the context of the concepts representing geographical places. The range of this hasGeometry property is one of the children of Geometry. There were 8 topological relationships. Two of them represented by the overlaps property, one occurrence of contains and five occurrences of inside.

We asked two humans with experience in working with ontologies to match the ontologies. Then we compared the matches produced by them against the ones produced by our proposed algorithm. As the ontologies were not so similar, both the humans and the concept matcher algorithm identified only two pairs of equivalent concepts, which were the only that actually existed. We then submitted the same ontologies to Prompt and H-MATCH and got the same matching results.

4.4 Publications

The concept-level matching algorithm has some publications, chronologically distributed as follows: In the AGILE conference on geographic information science 2005 (HESS; IOCHPE, 2005) a first version of the algorithm and of the similarity measurement metrics was proposed. The case study was geographic, but the particularities of the geographic concepts were not particularly addressed.

In GeoInfo 2006 (HESS; IOCHPE; CASTANO, 2006) and in the extended version published as a chapter of the book Advances in Geoinformatics (HESS; IOCHPE; CASTANO, 2007b) the complete algorithm for the concept-level matching was presented, as an extension of the conventional matcher H-MATCH (CASTANO; FERRARA; MONTANELLI, 2006) .

Finally, in the paper published in the International Conference on Spatial and Temporal databases of 2007 (HESS; IOCHPE; CASTANO, 2007c) we tailor made metrics to deal with the special features of the geographic concepts were presented.

5 PROPOSING A NEW INSTANCE-LEVEL MATCHING APPROACH

In this chapter we present the algorithm we developed for matching geographic ontology instances, as well as the mathematical expressions we created for actually measuring the similarity. Instances are only compared if they belong to equivalent concepts. This means that the instance matching algorithm must be executed after the concept matching algorithm. However, it might be the case where instance-level matching should be carried on even if the concept-level matching is not executed.

As novelties, comparing our proposal against the ones presented in Chapter 3, we can highlight three: (1) consideration of some metadata; (2) introduction of the concept of geographic context region; and (3) consideration of features other than the spatial position (coordinates) and the instance label (identifier). In the following sections they are explained in more details.

5.1 Metadata

The metadata has a central role in determining the degree of similarity. However, the metadata itself cannot be considered in the matching process. Instead, the metadata is used to provide additional information for the matching of other features, especially the geographic ones. The topological relationship similarity is affected by the geometries of the data, which depends on the scale in which the data was captured. A *hospital*, for instance, in a *1:5000000* scale is represented as a *point*, while the same *hospital* may be a polygon in a *1:250000* scale.

The spatial location of the data is particularly influenced by the metadata. For example, suppose we have two geographic instances i and $i^{'}$, one (i) with spatial position [-30;-53], and $i^{'}$ with spatial position [-3,320,469.29;307,084.89]. If we simply compare these two geographic coordinates, the result of the matching would be that i and $i^{'}$ are not the same data. However, if the metadata *projectionSystem* of i has the value *Geodetic* and the same metadata for $i^{'}$ has the value *Cartesian-UTM*, before stating that i and $i^{'}$ represent different data, we must convert the coordinates from one reference system to the other. In the case of this example, after the coordinate translation, we would identify that both instances have exactly the same position.

The metadata *measurement units* influences mainly non spatial features, but may have influence also on some spatial characteristics, if they must be stored in the ontology. To exemplify, the concept Road has a property *length*. If, in one ontology, the measurement unit for that type of property is *kilometers* and in the other ontology the respective measurement unit is *miles*, instances of that concept would have different *length* values.

Furthermore, the metadata *capture_date* influences both the spatial relations and the descriptive attributes. For example, an instance i of the concept Hospital identified as ErnestoDorneles with the metadata *capture_date* equals to 1970 certainly will have the value of the property *numBeds* smaller than the instance $i^{'}$ ErnestoDorneles with the metadata *capture_date* equals to 2000, because of the hospital's expansion during this time interval.

Currently, the metadata considered by the algorithm in the instance set matching are: (1) the representation scale; (2) the coordinate system; and (3) the projection system.

5.2 Geographic context region

In many cases the instances of an ontology represent data from a specific geographic context region. Furthermore, the instances share the same set of metadata (e.g. projection system, scale and coordinate reference system). Therefore, with the goal of accelerating the matching process we introduce the notion of a *geographic context region* and, consequently, of geographic context region similarity. A geographic context region of an ontology is the minimum region that contains all the ontology's instances. It is formed by the minimum bounding rectangle necessary to cover all instances. Furthermore, the geographic context region generalizes the instances' metadata.

Yet before of actually performing the instance-level matching algorithm, by considering the geographic context regions of ontologies O and $O^{'}$, we can check which instances from ontology O may have a match among the instances in ontology $O^{'}$ and those that certainly will not have, due to their spatial position (location). This can be achieved by performing a spatial overlay operation between the geographic context regions R and R' of, respectively, O and O'. The overlay operation produces a R_o region. If the regions covered by the two instance sets are disjoint, i.e., $R_o = \emptyset$, meaning that they do not have common instances, there is no need to compare the instances, since they will not refer to the same objects. If $R_o \neq \emptyset$, instances outside R_o may be eliminated from the comparison set because they certainly will not have a match in the other ontology.

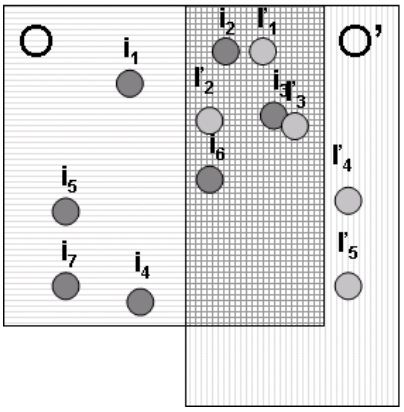

Figure 5.1: GeoRegion example

In the example of Figure 5.1, by applying the idea of the *GeoRegion*, when performing the *GeoRegionOverlay* some instances can be eliminated from the set of instances to be matched. In ontology O, instance i_1, i_4, i_5 and i_7 are outside the overlapping area. The instances $i_4^{'}$ and $i_5^{'}$ from ontology $O^{'}$ are outside the overlapping area as well. Therefore, in the next steps, instead of performing 35 spatial comparisons (7 instances from ontology

O against 5 instances from ontology O'), only 9 comparisons would have to be done, since only 3 instances from each ontology are inside the overlapping area. This can really save time.

5.3 Algorithm

After performing the concept-level matching and having identified the equivalent concepts, or after the user have identified the respective matching concepts, the instance-level matching can be performed. It only compares the instances i and $i^{'}$ if the concepts c and $c^{'}$ they instantiate, are considered to be equivalent. For all pairs of concepts $(c, c^{'})$ having similarity measure over the threshold, the instances are compared unless the user indicates otherwise.

As discussed previously, in order to gain some time on the instance-level matching, we introduce the notion of a geographic context region. As depicted in Figure 5.2, the region matching algorithm is responsible for homogenizing the two instances' set of metadata and for eliminating the instances from the ontology O that are geographically disjoint of all of the instances from the ontology $O^{'}$ and vice-versa.

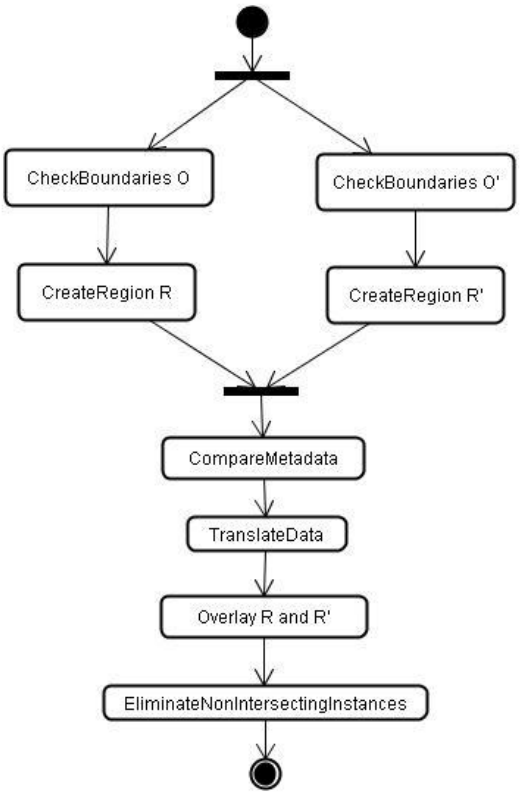

Figure 5.2: UML activity diagram for the geographic region matching

Firstly the boundaries of the two ontologies are defined based on the geographic coordinates of the instances stored in it (*CheckBoundaries* activity) and a geographic context region is created, using the minimum bounding rectangle (MBR) that covers all the boundaries (*CreateRegion* activity). Then, the metadata describing the two instances sets are compared and homogenized. The *TranslateData* activity performs the transformation of the instances property values which are affected by the metadata change, such as geographic coordinates, spatial representation, and so on. Finally, an overlay between the

two regions R and $R^{'}$ is executed. The instances outside the overlapping R_o area are eliminated from the instances set to be matched.

The encoding presented in Figure 5.3 summarizes the geographic context region algorithm execution. Only the code for the geographic context region generation and overlay is presented in the figure, not the metadata translation.

```
/*
  Vector vEqGeoCpt is the vector containing the pair of
  equivalent concepts (gc,gc').
  Vector vInst is the vector containing the
  instances of gc or gc'.
  considering latitude, longitude coords
*/
 minX = -90;
 minY = -180;
 maxX = +90;
 maxY = +180;
 for each concept gc from vEqGeoCpt {
   for each instance gi from gc.vInst{
   /*getposx and getposy return, respectively, the x and y
   coordinates*/
     minX = min(gi.getposx,minX);
     minY = min(gi.getposy,minY);
     maxX = max(gi.getposx,maxX);
     maxY = max(gi.getposy,maxY);
   }
 }
 /* creates the MBR of region R */
 R.setpos(minX,minY,maxX,maxY);
 minX = -90;
 minY = -180;
 maxX = +90;
 maxY = +180;
 for each concept gc' from vEqGeoCpt {
   for each instance gi from gc'.vInst{
   /*getposx and getposy return, respectively, the x and y
   coordinates*/
     minX = min(gi.getposx,minX);
     minY = min(gi.getposy,minY);
     maxX = max(gi.getposx,maxX);
     maxY = max(gi.getposy,maxY);
   }
 }
 /* creates the MBR of region R' */
 R'.setpos(minX,minY,maxX,maxY);
 RO = overlay(R,R');
 for each concept gc from vEqGeoCpt {
   for each instance gi from gc.vInst{
     if(contains(RO,gi)==false)
       gc.vInst.remove(gi);
   }
 }
 for each concept gc' from vEqGeoCpt {
   for each instance gi from gc'.vInst{
     if(contains(RO,gi)==false)
       gc'.vInst.remove(gi);
   }
 }
```

Figure 5.3: pseudo-code geographic context region algorithm

After having the instances described using the same set of metadata, the property values translated and eliminated the instances that for sure will not match with the instances from the other set, the instance matching can be performed. The instance-level matching algorithm is depicted in Figure 5.4.

The first step in the comparison of two instances is to measure their similarity (prox-

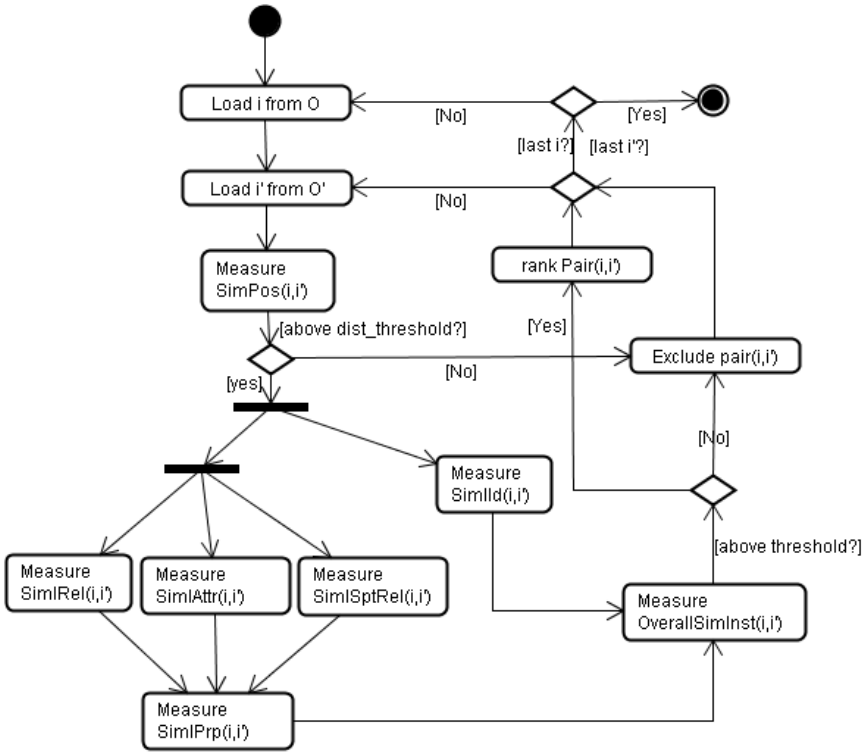

Figure 5.4: UML activity diagram for the instance matching

imity) regarding their spatial positions, i.e., locations. The *MeasureSimPos* activity calculates the distances between two instances using the metrics presented in Section 5.4.3. If the result is below a certain threshold, the pair (gi,$gi^{'}$) is eliminated before comparing the other features. Then the similarity is measured regarding the other four aspects that may cause heterogeneity and not yet addressed: instance identifier (*MeasureSimIId* activity), relationships (*MeasureSimRel* activity), attributes (*MeasureSimIAttr* activity) and spatial relationships (*MeasureSimISptRel* activity). An overall instance similarity is measured in a balanced sum and, just as at the concept-level, the pairs of instances (i,$i^{'}$) with similarity value lower than a threshold are excluded, while the others are presented to the user in a ranked list. If the instances do not belong to geographic concepts, they do not have a location. In this case the position similarity $SimPos(gi, gi^{'})$ is not measured, as well as $SimTop(gi, gi^{'})$ and $SimDir(gi, gi^{'})$.

The encoding presented in Figure 5.5 summarizes the instance-level matching algorithm execution.

5.4 Metrics

Two instances $i \in O$ and $i^{'} \in O^{'}$ are compared only if the concepts c and $c^{'}$ they instantiate were already identified as equivalent. The instance similarity measurement is based on six main components, which may cause the instance heterogeneity: (1) the instance identifier, (2) the value of the descriptive attributes (data type properties), (3) the value of the descriptive relationships (object type properties), (4) the value of the spatial relations (spatial object type properties), (5) the spatial position of the instances (coordinates) and, (6) metadata.

When a concept is instantiated, each associated property has a value $vp(t(p), val)$.

```
/*
  INPUTS:
    The set of instances of two equivalent concepts which
  were not eliminated in the geographic context region test
    The metadata of the instances
    The weight for the instance id similarity WID
*/
/*
  The inputs are two equivalent concepts, gc and gc'
  Vector vInst is the vector containing the
  instances of gc or gc'.
*/
  for each instance gi from gc.vInst{
    pos = centroid(gi);
    for each instance gi' from gc'.vInst{
      /*the centroid function returns the (x,y)
      coordinates of the center of the instance*/
      pos' = centroid(gi');
      SimPos = (1/dist(pos,pos'));
      if(SimPos > posThreshold){
        simId = SimIId(gi,gi');
        /* SimIAt(pv(gi,gi')) is the similarity for each
        attribute value av(gi,gi')*/
        simAttrVal = Sum(SimIAt(av(gi,gi')));
        /* SimIRel(pv(gi,gi')) is the similarity for each
        property value pv(gi,gi') representing a conventional
        relationship*/
        simRelVal = Sum(SimIRel(pv(gi,gi')));
        /* SimITop(pv(gi,gi')) is the similarity for each
        property value pv(gi,gi') representing a topological
        relationship*/
        simTopVal = Sum(SimITop(pv(gi,gi')));
        /* SimIDir(pv(gi,gi')) is the similarity for each
        property value pv(gi,gi') representing a directional
        relationship*/
        simDirVal = Sum(SimIDir(pv(gi,gi')));
        InstSim = WId * SimId + (1-WId)*(SimAttrVal +
                  SimTopVal + SimRelVal + SimDirVal)
        if(instSim > instanceThreshold
          store(gi,gi');
      }
    }
  }
```

Figure 5.5: pseudo-code instance-level matching algorithm

When two instances are compared, only the equivalent properties are verified, i.e., if $p \equiv p\prime$, which is determined previously, in the concept similarity assessment phase. Thus, the only component to be confronted is the value (val) from the triple. The similarity among the property values $vp(t(p), val)$ of two instances depends on the type of the property.

5.4.1 Identifier

When measuring the similarity between two instances i and $i^{'}$, the instance identifier has to be considered. As mentioned in Section 2.2 the id is the instance component property which represents the unique identifier of an instance, i.e., the id component of the 4-tuple cannot be the same for two instances in the same ontology O. The identifier similarity measure $SimIId(i, i^{'})$ of two instances i and $i^{'}$ is given by the same string comparison metric used to measure the concept name similarity in Section 4.2.1.

$$SimIId(i, i^{'}) = \frac{(2 * \sum_{t(i)} \frac{length(max(ComSubstring_{t(i)}))}{length(t(i))+length(t(i^{'}))}) + (JaroWinkler(t(i), t(i^{'})))}{2} \quad (5.1)$$

5.4.2 Property similarity

Attributes: In the case of a data type property, i.e., an attribute, to which the allowed values are numeric, a simple equality comparison is performed:

$$SimIAtN(vp(t(p), val), vp(t(p^{'}), val^{'})) = \begin{cases} 1 & \text{if } val = val^{'} \\ 0 & \text{otherwise} \end{cases} \quad (5.2)$$

If the numeric types are different, for example *integer* x *float*, only the common part is compared. This means that if one property value is 10 and the other is 10.5 only the integer part of the numbers is compared.

In the case of a data type property to which the allowed values are text (string) the similarity measurement is performed according to the string metric similarity defined for the concept-level.

$$SimIAtS(vp(t(p), val), vp(t(p^{'}), val^{'})) = \quad (5.3)$$
$$\frac{(2 * \sum_{val} \frac{length(max(ComSubstring_{val}))}{length(val)+length(val^{'})}) + (JaroWinkler(val, val^{'}))}{2}$$

Relationships: If the property is an object type property, it represents a relationship. At the instance-level, as we are concerned if the associated instances are equivalent, for both conventional and spatial relationships the similarity measure is the same. We simply compare if the instances i_x and $i_x^{'}$ associated, respectively, to i and $i^{'}$ are equivalent. The instances i_x and $i_x^{'}$ are the val component of the triple $vp(i, p, val)$.

$$SimIR(vp(t(p), val), vp(t(p^{'}), val^{'})) = \begin{cases} 1 & \text{if } val = val^{'} \\ 0 & \text{otherwise} \end{cases} \quad (5.4)$$

For the matching of spatial relationships, the only difference is that the relationship names must be considered, because the name of a spatial relation carries it semantics. In analogy to the concept-level, at the instance-level it is also necessary to use the $eqTop(top, top^{'})$ function when measuring the similarity regarding the topological relationships:

$$SimITop(vp(t(p), val), vp(t(p^{'}), val^{'})) = \begin{cases} 1 & \text{if } eqTop(top, top^{'}) \wedge val = val^{'} \\ 0 & \text{otherwise} \end{cases} \quad (5.5)$$

In the case of a spatial relationship other than topological, the geometry does not have influence, and thus the similarity can be measured simply by

$$SimIDir(vp(t(p), val), vp(t(p^{'}), val^{'})) = \begin{cases} 1 & \text{if } val = val^{'} \wedge t(p) = t(p^{'}) \\ 0 & \text{otherwise} \end{cases} \quad (5.6)$$

If a property is present in only one of the compared instances, i.e., it is associated to only one of the concepts, the similarity regarding that property is considered zero. Hence, the final equation for measuring the similarity between two instances i and $i^{'}$ is

$$SimPrp(i, i^{'}) = \quad (5.7)$$
$$\frac{\sum SimIAtN + \sum SimIAtS + \sum SimIR + \sum SimIDir + \sum SimITop}{|P| \cup |P'|}$$

5.4.3 Geographic coordinates

The geographic coordinates are aspects which play a crucial role in the integration of geographic instances. Although the spatial position of an instance may vary along time, the coordinates may be of great use in most of the cases.

To compare the similarity regarding the geographic coordinates of two geographic instances gi and $gi^{'}$, first it is necessary to reduce them to the same geometry. Thus, all the instances are transformed to points. The transformation from $line$ and $polygon$ to point is made in the same way: as the coordinates may not form a straight line or a regular polygon, we first calculate the minimum bounding rectangle (MBR) to cover all the coordinates area and then we extract the centroid of this rectangle. Then, given two pairs of coordinates $(x, y) \in i$ and $(x^{'}, y^{'}) \in i^{'}$ the similarity is measured by the inverse of the euclidian distance between these two pairs of coordinates, as follows:

$$SimPos(i, i^{'}) = \frac{1}{dist(i, i^{'})} \tag{5.8}$$

The coordinates similarity may not be used for the final similarity measure, but this can exclude some pairs of instances which are located to far from each other. Thus, if the coordinates similarity does not reach a certain threshold, the pair $(i, i^{'})$ is excluded from the list of possible matches.

Another possibility for measuring the degree of similarity regarding the spatial position is to perform a spatial join between the two MBRs. The overlapping area gives, then the similarity of the two spatial positions, as follows:

$$SimPos(i, i^{'}) = \frac{MBR(i) \otimes MBR(i^{'})}{MBR(i) \cup MBR(i^{'})} \tag{5.9}$$

The algorithm for the spatial join is still under research, but is not going to be developed for the thesis. We will use an existing one, such as the one of Fornari et al. (FORNARI; COMBA; IOCHPE, 2006).

5.4.4 Overall similarity

The final similarity value when comparing two instances i and $i^{'}$ is then given by:

$$SimInst(i, i^{'}) = \rho * SimIId(i, i^{'}) + (1 - \rho) * SimPrp(i, i^{'}) \tag{5.10}$$

where ρ is the weight for the identifier similarity and can be a value within [0,1].

In the case the instances are not geographic, i.e., they instantiate conventional concepts, the coordinate similarity is not considered. Furthermore, the spatial relationships similarity is also ignored.

5.5 Testing the proposal

In Chapter 4 we used the austrian **e-tourism** and spanish **andalucia-tourism** ontologies for the concept matching tests. However, these ontologies are geographically disjoint and, thus, do not store the same, or at least equivalent, instances. Therefore, we kept only the **e-tourism** ontology.

The goals of the tests we run were to evaluate the adequacy of the matchings found by our algorithm in respect to the ones manually identified by humans as well as to evaluate the benefits of using the concept of geographic context region in the matching process. The benefits can be measured regarding the time saved by early eliminating instance that are located in non overlapping areas of the two ontologies *versus* the miss of possible pairs of matching instances. As the existing proposals for instance matchers didn't have prototype tools available neither detailed the similarity measurement mathematical expressions used, we could not test our proposal against them.

To check if the algorithm could produce correct matching, we checked if it was capable of matching identical instances. Therefore, we created a copy of that ontology and matched their instances. As expected, all instances from the original ontology O had a match in the copy ontology $O^{'}$.

5.5.1 Test I1: Instances against

In the first test we manually changed some of the **e-tourism** ontology property values creating a second ontology. We introduced small changes in the name (label) of the places, their location and in some of the descriptive properties. As that ontology did not describe spatial relationships, they were not considered in the matching process.

We ended up with three concepts:

- `Accommodation` with 10 instances;
- `Infrastructure` with 12 instances and;
- `Guestroom` with 33 instances.

The first two were geographic concepts, while the latter was a conventional domain concept. As both ontologies were derived from the same original ontology, their instances were described using the same projection scale, coordinate reference system and projection system metadata. Furthermore, although we changed a little the spatial position of the instances, the regions covered by the sets of instances of the equivalent concepts from both ontologies were the same. Therefore, the use of the *GeoRegion* would not make any sense.

In this test we expected to have 55 matches, i.e., we expected the instance matching algorithm to find that all instances from one ontology had a match in the other ontology. This indeed happened.

5.5.2 Test I2: Few geographically distant instances

In a second test we picked up the 10 instances of the `Accommodation` concept and created a new ontology. Then we created some new properties and also deleted some of the existing properties. Table 5.1 shows the context for each concept.

We added four instances to the original ontology and created 15 new instances in the second ontology.

- Seven of them were exactly equal to the instances of the original ontology;
- Three were very similar, i.e., had spatial positions very close and more or less the same property values;
- Five instances did not have a match in the original ontology. Among these five, three were in locations quite distant from the others.

Table 5.1: Contexts and equivalence of the two `Accommodation` concepts

`Accommodation` in O	`Accommodation` in O'	type
hasBaggageRoom	-	boolean
hasBreakfast	hasBreakfast	boolean
hasElevator	hasElevator	boolean
hasGeometry	hasGeometry	relationship
hasName	hasName	string
hasPool	hasPool	boolean
hasPostalAddress	hasPostalAddress	string
hasRoom	hasRoom	relationship
hasSauna	-	boolean
hasStarRating	hasStarRating	integer
-	hasType	string
-	offersChildCare	boolean
spokenLanguages	spokenLanguages	relationship

the properties in the same lines represent equivalent properties, while when there is a "-" in one of the cells means that that property did not have an equivalent in the other concept.

The four additional instances of the first ontology were spatially located in the opposite direction from the ones in the second ontology.

We run two different tests, one using the *GeoRegion* and one not using it. The idea was twofold:

- to verify if the accuracy of results were affected by the early elimination of instances outside the overlapping area of the two instances sets;
- to analyze the amount of time saved.

Table 5.2 shows the results produced by the two executions of the algorithm.

Table 5.2: Results of the execution of test I2

Parameter	Without geographic region	With geographic region
Pairs found	9	9
Recall	90%	90%
# of comparisons	210	120
Overall execution time(s)	35	40

In the execution we did not use the *GeoRegion*. Therefore, 14 instances from one ontology had to be compared against 15 instances in the other ontology, with a total of 210 comparisons. In the second test we used the *GeoRegion* and only 120 comparisons were performed (10 x 12). The results in terms of recall were the same, but the time increased in the second execution in 5 seconds, although performing less comparisons.

Analyzing the reason for the increase in the matching time we realized that the time spent for processing the *GeoRegion*, i.e., creating the *GeoRegion*, performing the overlay

and deleting the instances outside the overlapping area, was longer than the time saved by not performing 90 comparisons.

5.5.3 Test I3: Many geographically distant instances

To test the actual benefits of the *GeoRegion* we had to run one more test, with larger sets of instances. We created eight more instances to one of the ontologies and six instances to the other ontology, specifying only their geographic location. They were, of course, located outside the area covered by both of the ontologies. We had, than, 22 instances in one ontology and 21 instances in the other ontology. We knew that only nine matches would be found, which represents something about 42% of the instances from each one of the ontologies. Again, we first executed the matching without using the *GeoRegion* and a second time with the *GeoRegion*, and the results are presented in Table 5.3.

Table 5.3: Results of the execution of test I3

Parameter	Without geographic region	With geographic region
Pairs found	9	9
Recall	90%	90%
# of comparisons	462	120
Overall execution time(s)	65	45

As can be concluded, when there a many instances from both ontologies outside the overlapping area the time spent for performing the geographic context region worths the time saved by reducing the number of instances to be compared (120 instead of 462).

5.6 Publications

A first version of the instance-level matching algorithm was published in the International Symposium on Spatial and Temporal Databases (SSTD) in 2007. The geographic region was not detailed in that paper.

6 GEOGRAPHIC ONTOLOGY REVERSE ENGINEERING

Ontologies can be defined by different groups, for different purposes. Therefore, the level of detail describing the ontologies may vary. Furthermore, although the Open GIS Consortium recommendation states that a geographic concept (object, in the OGC's vocabulary) must be related to a geometry through a `hasGeometry` property (OGC, 2005), this is not consensual in the GIS community (SPACCAPIETRA et al., 2004). This leads to a scenario where two ontologies to be compared may be very different regarding their taxonomies and the properties associated to their concepts.

The ontology reverse engineering technique (ORET) was conceived with the goal of homogenizing the input ontologies semantic granularity. It focus mainly on enriching the taxonomy of the ontologies, defining spatial relationships, especially the topological ones, and on assuring that a geographic concept is associated to a geometry. Although it is not the primary goal of the ORET, it can also (re)built an entire concept from the values of the properties associated to an instance. The result is an OGC compliant ontology, based on the reference model presented in Chapter 2.

6.1 Related work

To the best of our knowledge there are no works addressing exactly the same issue we do, or at least, not by dealing with all the features we deal with. Regarding the generation of concepts from instances, the most related work found in the literature address the process of generating ontologies from data intensive web sites (STOJANOVIC; STOJANOVIC; VOLZ, 2002; ASTROVA; STANTIC, 2005; BENSLIMANE et al., 2006), i.e., web sites that are dynamically generated at the time of user requests, and are based on relational databases. Therefore, the focus of these works is to use reverse engineering techniques to create ontologies from relational databases, either from the schema or from the data itself. The generation of an ontology from a relational database is also discussed in (TRINKUNAS; VASILECAS, 2007; LI; DU; WANG, 2005). The use of a reference (background) domain ontology to help structuring concepts extracted from tags existing in web sites is proposed in (ALEKSOVSKI et al., 2006; SPECIA; MOTTA, 2007). These proposals, however, are meant for conventional, non-geographic databases and ontologies. Baglioni et al. (2007) is the only proposal we found for automatically generating a geographic ontology from a spatial database.

Li, Du and Wang (2005), Benslimane et al. (2006) and Stojanovic, Stojanovic and Volz (2002) propose a number of rules for generating classes, properties and property characteristics, cardinality and instances from relational databases or entity relationship conceptual schemas. The rules consider the primary and foreign keys of the relations to determine whether a class must be created, or if an object or data type may be associated

to a class. Furthermore, axioms defining the property cardinality can be created. Although in these three approaches there is one rule for constructing hierarchies, they can basically create flat ontologies, i.e., structures that either present all concepts at the same level in the hierarchy, or, at most, with one *parent-child* relationship level. Astrova (2004) goes beyond the above cited works by creating rules not only for single, direct inheritance, but also for multiple inheritance. Furthermore, as not only the primary and foreign keys are analyzed, but all the attributes of a relation, it is possible to discover that two concepts are "siblings", even if there is not an explicit common super-concept.

Specia and Motta (2007) extract concepts from web site tags and use string-metric distance to group morphologically very similar tags into a single concept and than cluster concepts using statistical analysis of co-occurrence. Then, with the support of a number of reference ontologies they try to discover relationships between pair of concepts. A relationship may be hierarchical or as domain-range values for a property. In the first case, one concept can be a generalization/specialization of the other, they both can have the same parent, or a common ancestor. In the domain-range case, a tag is the range or the value of the property of the other concept (SPECIA; MOTTA, 2007).

In (BAGLIONI et al., 2007) a proposal for generating geographic ontologies from geographic databases is presented. The proposal applies translation rules from spatial tables to geographic concepts in the ontology. The database must be OGC compliant, which means that the tables must have a "the_geom" column, with the information of the geometry of the data and the set of geographic coordinates. In the ontology, each tuple from the spatial table is mapped into three classes: (i) a specialization of the `geographic object` concept, describing the concept's attributes and relationships with other concepts; (ii) a specialization of the `georef` concept, holding the geographic coordinates of the data; and (iii) an instance of a `geometry` concept or of one of its specializations, such as `point`, `line` and `polygon`. All these concepts in the ontology are related. This ontology can be enriched by searching a domain ontology. This allows, for example, the definition of the ontology's taxonomy.

6.2 The proposed technique for geographic ontology enrichment

In the proposal presented here we adapt some of the principles used in the referenced works to create ontologies not from databases instances, but from OWL instances. Especially regarding the (re)construction of the ontology hierarchy, we extend the existing proposals by eliminating some redundant parent-child relationships and also by creating some "intermediary" concepts to accommodate siblings. Furthermore, we use the location property values of the instances to infer some topological relationships between a pair of extracted concepts. Differently from (BAGLIONI et al., 2007), we do not need a domain ontology to guide the ontology reconstruction. The process can be executed either in the presence of the domain ontology or not. Furthermore, as there is not yet a consensus of how describing the spatiality of a concept in a geo-ontology - with a geometry class, following the OGC recommendation, or with only the set of geographic coordinates - our proposal accepts both as inputs.

The overall algorithm executed by the geographic ontology reverse engineering algorithm is as follows:

1. Parse instances and create concepts: if the explicit definition of concepts is not available, the owl tags of the instances are analyzed. Concepts and properties are created for the different tags. The properties are attached to concepts.

2. Define geometries: In case of a non OGC compliant ontology, parse the coordinates of each instance of a given extracted concept and discover its geometry. Create a geometry concept and attach it to the geographic concept being parsed through a *hasGeometry* property.

3. Infer topological relationships: Having the geometries defined for every concept, for each pair of concepts discover the possible topological relationships that may hold between them. Add properties representing these relationships to the context of the concepts.

4. Rebuild hierarchies: Combining reverse engineering techniques and the support of a reference ontology (if there is one) define parent-child relationships for the existing concepts and enrich the ontology taxonomy by creating new, intermediary, concepts to better accommodate the ones from the original input ontology (e.g., sibling concepts).

Figure 6.1 presents the UML activity diagram for the geographic ontology reverse engineering algorithm.

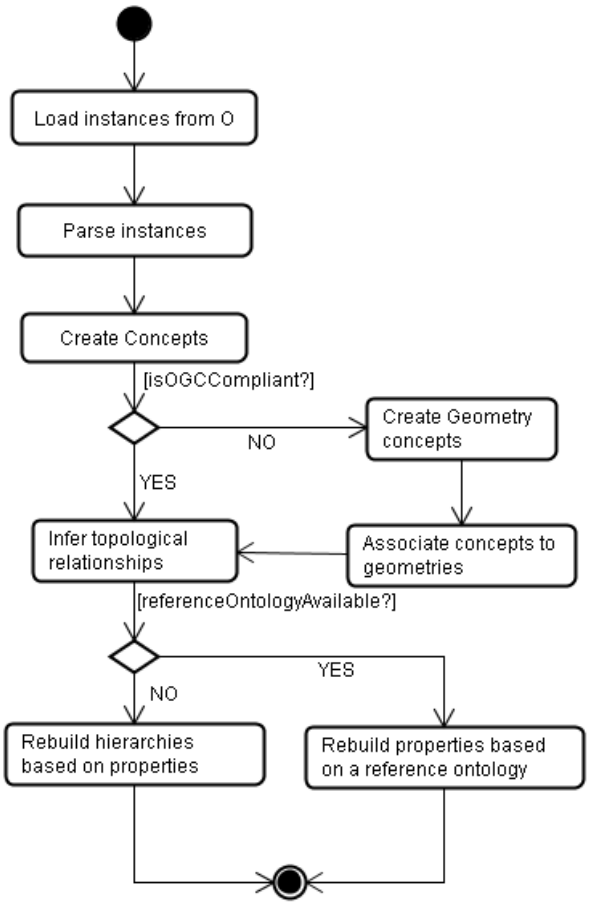

Figure 6.1: UML activity diagram for the reverse engineering algorithm

To illustrate how the ontology reverse engineering technique works, let's consider the OWL encoding below. It shows a piece of a geographic ontology composed only by instances. This ontology is compliant to the OGC recommendation. This can be verified by the existence of instances of `Polygon`, `Point` and `Line` and a property

hasGeometry that links an instance of a given concept with an instance of one of those three concepts. Furthermore, the geographic location is encoded in the geometric instance.

```
<Polygon rdf:ID="Pol2">
  <location rdf:datatype="string">
    (12,45);(20,45);(20,50);(12,50);(12,45)
  </location>
</Polygon>
<Point rdf:ID="Pt2">
  <location rdf:datatype="string">(16,47)</location>
</Point>
<Road rdf:ID="IpAv">
  <hasName rdf:datatype="string">Ipiranga</hasName>
  <hasGeometry>
    <Line rdf:ID="Line2">
      <location rdf:datatype="string">(22,20);(22,70)</location>
    </Line>
  </hasGeometry>
</Road>
<Campus rdf:ID="UFRGS">
  <hasGeometry rdf:resource="#Pol1"/>
  <hasName rdf:datatype="string">Campus do Vale</hasName>
  <POBox rdf:datatype="string">38492</POBox>
  <hasAdministrator rdf:datatype="string">
    Hanemann</hasAdministrator>
</Campus>
<Lesson rdf:ID="LSS_GIS">
  <hasPlace>
    <Classroom rdf:ID="UFRGS_B1_CL202">
      <hasNumber rdf:datatype="string">202</hasNumber>
      <hasCapacity rdf:datatype="int">30</hasCapacity>
      <hasGeometry rdf:resource="#Pt2"/>
    </Classroom>
  </hasPlace>
  <hasName rdf:datatype="string">Geogr. Inf. Systems</hasName>
</Lesson>
<GradStudent rdf:ID="GradStudent_2">
  <hasStartYear rdf:datatype="int">2007</hasStartYear>
  <hasSalary rdf:datatype="float">1000.0</hasSalary>
  <hasName rdf:datatype="string">Carl Wright</hasName>
  <hasOffice>
    <Office rdf:ID="GradStdOffice_2">
      <hasNumber rdf:datatype="string">105</hasNumber>
      <hasGeometry>
        <Point rdf:ID="Pt7">
          <location rdf:datatype="string">(15,55)</location>
        </Point>
      </hasGeometry>
    </Office>
  </hasOffice>
  <hasPositionName rdf:datatype="string">
```

```
    Master student</hasPositionName>
  <hasResearchArea rdf:datatype="string">
    ontologies</hasResearchArea>
  <hasTitle rdf:datatype="string">bsc</hasTitle>
</GradStudent>
<Lab rdf:ID="NetLab">
  <hasSubject rdf:datatype="string">Networks</hasSubject>
  <hasNumber rdf:datatype="string">207</hasNumber>
  <hasName rdf:datatype="string">Network lab</hasName>
  <hasCapacity rdf:datatype="int">20</hasCapacity>
  <hasGeometry>
    <Point rdf:ID="Pt4">
      <location rdf:datatype="string">(14,49)</location>
    </Point>
  </hasGeometry>
</Lab>
<Building rdf:ID="UFRGS_B2">
  <numFloors rdf:datatype="int">1</numFloors>
  <hasGeometry rdf:resource="#Pol3"/>
  <hasNumber rdf:datatype="string">68</hasNumber>
</Building>
<Researcher rdf:ID="Researcher_1">
  <hasTitlerdf:datatype="string">PhD</hasTitle>
  <hasHireYear rdf:datatype="int">2006</hasHireYear>
  <hasSalary rdf:datatype="float">4000.0</hasSalary>
  <hasName rdf:datatype="string">Carol Kerr</hasName>
  <hasResearchArea rdf:datatype="string">
    Databases</hasResearchArea>
  <teaches rdf:resource="#Lss_DB"/>
  <hasPositionName rdf:datatype="string">
    Associate researcher</hasPositionName>
  <hasOffice rdf:resource="#ResearchersOffice"/>
</Researcher>
<Professor rdf:ID="Professor_1">
  <hasResearchArea rdf:datatype="string">GIS</hasResearchArea>
  <hasTitle rdf:datatype="string">PHD</hasTitle>
  <teaches rdf:resource="#LSS_GIS"/>
  <hasHireYear rdf:datatype="int">1998</hasHireYear>
  <hasPositionName rdf:datatype="string">
    Full professor</hasPositionName>
  <hasSalary rdf:datatype="float">6500.0</hasSalary>
  <hasName rdf:datatype="string">Alan Gonzales</hasName>
  <hasOffice rdf:resource="#ProfOffice_1"/>
</Professor>
```

6.3 Instance parsing and concept creation

The *InstanceParser* module is in charge of reading the ontology's OWL instance tags. From them it extracts the useful information to be used by the *ConceptCreator* module. At this moment we are interested in the tag's label and attributes, not in the values, which are specific for each instance. Each tag corresponds to either a concept or a property. If the tag

has an attribute `rdf:ID` (for example, `<Professor rdf:ID="Professor.1">`, then it corresponds to a concept. All the nested tags, i.e., tags within the opening and closing tag that define the instance, are considered as properties. However, if we find another tag with the attribute `rdf:ID` this means that a new concept is being instantiated. Furthermore, the nested concept is associated to the broader one. In the encoding above, we have the `Office` tag nested to the `GradStudent` tag.

In an ontology, a property may be either a *data type* or an *object type*. Data type properties are the ones that accept literal values as possible ranges, such as *integer*, *double*, *boolean*, and *string*, just as attributes in a database. On the other hand, object type properties accept as their range instances of other concepts, i.e., have the semantics of relationships between concepts. A data type property is identified by the tag attribute `rdf:datatype` (e.g., `<hasTitle rdf:datatype = "string">`), while an object type property is represented in OWL by the tag attribute `rdf:resource` (e.g., `<teaches rdf:resource = "#LSS.GIS">`). Furthermore, in case of an object type property, the value of the `rdf:resource` attribute is a reference to the associated concept.

As already mentioned, a relationship holding between two concepts can be also encoded in OWL by nested concepts, such as `Office` and `GradStudent` in the example. In that case, the tag located immediately before the tag that defines the nested concept represents the object type property that relates them to one another. In the example, the tag `hasOffice` indicates that an object type property must be associated to `GradStudent` and can have `Office` in its range.

The next step executed by the algorithm is the analysis of the data structure received as input. For each line (or tuple), each different label outside the brackets generates a concept. The first element inside the brackets is the name of the instance from which the concept was extracted. The other elements form the concept's context. Each one is read as a pair*(property,range)*, and therefore originate a property.

The first time a label is found, a concept is created representing it. If the same label is found more than once (e.g., `Professor`), the concept's context is updated if needed. For example, suppose that an instance *i*, of label *c*, has values for the properties *p1*, *p2* and *p3*. Therefore, when the concept *c* is created, only the properties *p1*, *p2* and *p3* are created and attached to *c*'s context. However, when parsing an instance i' of the same label *c*, values for properties *p1*, *p2* and *p4* are found. As the concept *c* already exists, a new concept is not created. Instead, as a value for a new property is associated to i' (*p4*), a new property is created and attached to the context of the concept *c*.

When processing a property, two are the possible classification for the *range* component: it may be a *resource* (when `"resource:"` is found) or a *datatype* (when `"datatype:"` is found). In the first case it indicates that the property plays the role of a relationship between the concept and another concept (object type property). Otherwise, it indicates that the property plays the role of an attribute (data type property). In case of a relationship property, what follows the `":"` symbol is the name of the related instance. Then the *ConceptCreator* module has to find the concept that the instance represents by searching the first element of the concept's tuple. For example, in the case of the `Graduate` concept, the `(advisor,resource:CI)` pair indicates that the property `advisor` associates it with the concept instantiated by `CI`. By searching for *CI*, the concept `Professor` is found. Therefore, the algorithm infers that the range for the property `advisor`, in the context of the concept `Graduate`, is the concept `Professor`.

It may happen that two different instances have, for the same object type property,

different ranges. Furthermore, it may also happen that for a given instance, a property has more than one range, and the associated instances belong to different concepts. For example, the concept `journal` has twice the property `author`, one with value `resource:CI` and the other with value `resource:GNH`. By searching for the concepts corresponding to the instances `CI` and `GNH`, two concepts are found: `Professor` and `Graduate`, respectively. Therefore, in the definition of the concept's context, multiple ranges have to be allowed for the property.

The output of the second step of the algorithm for the running example is graphically depicted in Figure 6.2 and is encoded in Figure 6.3.

6.4 Inferring topological relationships

In the specific case of geographic ontologies, it is possible to discover topological relationships between the ontology concepts besides than the conventional relationships. To do that we have to analyze one of the spatial characteristics of the instances: their geometries. However, not always the geometry of an instance is explicitly defined. Our proposal aims at being compliant to the OGC recommendation as well as to the geographic ontology community, as discussed in Section 2.1. Therefore the reverse engineering method we propose here is capable of discovering topological relationships when the instances of geometric concepts are explicitly connected to the instances of geographic concepts, but also when the geometries are not part of the ontology. In the first case, as the OGC recommends, there is a `hasGeometry` object type property associating the geographic object instance to the geometry instance. Furthermore, is the geometry instance that holds the position (coordinates) through the `location` data type property. If the geographic ontology is built without the geometry concepts and instances, we can discover the concepts geometries by analyzing the property associated to the instance that holds the geographical coordinates. In this case, however, as none standard is followed, the user has to inform which is the property representing the location of the instance, i.e., the geographic coordinates.

To carry out the discovery of topological relationships we assume the following two premises to be true:

- All the instances from the input ontology are described by the same set of metadata. In other words, they have the same values for scale, projection system and coordinate reference system;
- All the instances from a given geographic concept have the same geometry;

The procedure for inferring the topological relationships is as follows:

1. Discover the instances geometries: this step is executed only if the instances are not explicitly associated to instances of geometry concepts. Based on the values for the geographic coordinates, a geometry is inferred and associated to the concept being created. If it has only one pair of coordinates `(x,y)`, then a *point* geometry is given. If it has two or more pair of coordinates (x_1, y_1), (x_2, y_2), but they do not form a ring, then the concept's geometry is defined as being a *line*. Finally, if the instance has more than two pairs of coordinates and compose a ring, the geometry is set to *polygon*.

2. Once each concept has an associated geometry, we can proceed with the process, by executing a function `eRelate(geom1,geom2)`. `geom1` and `geom2` are,

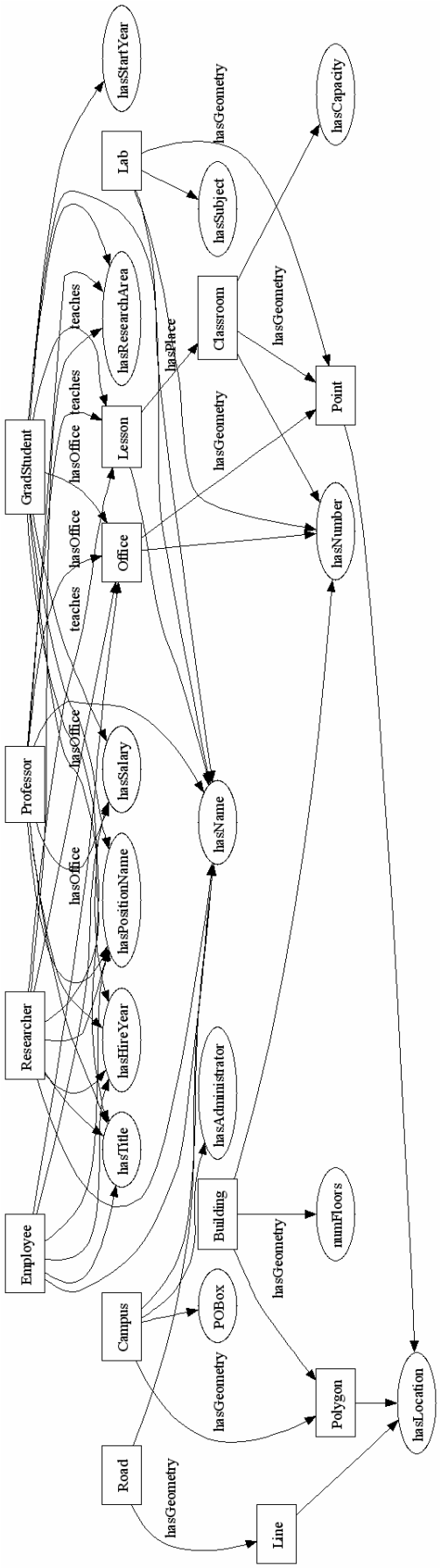

Figure 6.2: The extracted ontology

C_g = $Lesson, Employee, GradStudent$ (domain)
= $Professor, Researcher$ (domain)
$Road, Campus, Building$ (geographicdomain)
$Office, Classroom, Lab$ (geographicdomain)
$Line, Polygon, Point$ (geometry)
P_g = $numFloors, hasCapacity, hasNumber, POBox, hasPlace$ (conventional)
$hasName, hasResearchArea, hasStartYear, hasSalary$ (conventional)
$hasSubject, teaches, hasOffice, hasTitle$ (conventional)
$hasHireYear, hasPositionName, hasAdministration$ (conventional)
$inside, contains, crosses$ (spatial)
$hasGeometry$ (geometric)
$hasLocation$ (positional)
A = $hasGeometry(Campus, Polygon)$
$hasGeometry(Road, Line)$
$hasGeometry(Building, Polygon)$
$hasGeometry(Office, Point)$
$hasGeometry(Classroom, Point)$
I_g =
M =

Figure 6.3: Parsed concepts and properties

respectively, the geometries of the two concepts being compared. The output of this function is a list of possible topological relationships for the two given geometries, based on Egenhofer's 9-intersection model (EGENHOFER; FRANZOSA, 1991).

3. For each one of the relationships returned by the `eRelate(geom1,geom2)` function, test if it really occurs between two instances of the concepts being compared. This test is based on the instances spatial position (i.e., location). If the topological relation occurs at least once, define that relationship as existing between the concepts the instances belong to.

In the example we are using to illustrate the process the ontology is OGC compliant. Therefore, only steps 2 and 3 of the algorithm have to be executed.

The final step of inferring topological relationships consists on eliminating redundant relationships. In the example, for the concept `Office` we have that it is `inside Building`, and `inside Campus` as well. For the concept `Building` there is also a property `inside Campus`. By definition, the `inside` property is transitive and, therefore, we do not have to explicitly define `Office inside Campus`. The output of this phase, i.e., the inferred topological relationships, is encoded as the axioms of the ontology model, as presented in Figure 6.4:

6.5 Rebuilding hierarchies

The last phase of the reverse engineering algorithm execution consists on (re)building and enriching the ontology's hierarchy. The goal is to put all the concepts in a unique hierarchy with more than two levels, i.e., the root level, with the `Thing` concept and the level where all concepts are included, as children of `Thing`. For that purpose this

$$
\begin{aligned}
A \;=\; & crosses(Road, some(Campus)) \\
& inside(Building, some(Campus)) \\
& inside(Classroom, some(Building)) \\
& inside(Office, some(Building)) \\
& contains(Campus, some(Building)) \\
& contains(Building, some(Office)) \\
& contains(Building, some(Classroom)) \\
& hasGeometry(Campus, Polygon) \\
& hasGeometry(Road, Line) \\
& hasGeometry(Building, Polygon) \\
& hasGeometry(Office, Point) \\
& hasGeometry(Classroom, Point)
\end{aligned}
$$

Figure 6.4: Axioms representing the inferred topological relationships

module combines some techniques from the database to ontology reverse engineering field (STOJANOVIC; STOJANOVIC; VOLZ, 2002; ASTROVA, 2004) with the support of a reference ontology. The basic issue here is to find the shortest path between two given concepts. The number of *parent-child* relationships composing this path is the distance between the two given concepts.

When the concepts are created, they are all sub-concepts of the general concept `Thing`. In order to enrich the ontology's taxonomy, the concept wrapper algorithm executes two steps. Firstly, it verifies the hierarchical relationships holding between the ontology concepts in the reference ontology, if there is one. For that purpose the concepts inferred from the instances are compared against the ones in the reference ontology in a simple matching process. Then, based on the properties associated to the concepts, the taxonomy is refined. In case of not existing a reference ontology, only the second phase of the process is executed, i.e., the ontology hierarchy is built based only on the concepts properties.

6.5.1 Reference ontology search

Rebuilding the ontology based on a reference ontology consists on comparing each concept c from the input ontology against each concept c_r from the reference ontology. This comparison is a very simple matching process: if the concepts c and c_r are defined by the same label they are considered equivalent to one another. If the labels are different, an external dictionary is searched to verify if the labels of c and c_r are synonyms. If so, the concepts are considered equivalent as well. No other features (i.e., properties or axioms) are considered. The goal here is to accelerate the comparison process.

Once c and cr are identified as equivalent, the algorithm resumes as follows:

1. Search, in the reference ontology, the concept that is the direct superclass of c_r, i.e., the concept c_{rx} which holds a $subClassOf$ relationship with c_r. If a concept c_{rx} other than `Thing` is found, check if there is, among the concepts, a concept c_x equivalent to c_{rx}. The matching strategy is the same as for c and c_r. If there is a match, than create a *parent-child* relationship between c_x and c (i.e., define that c is a *subclassOf* c_x).

2. For each concept c that in step 1 was not identified a direct superclass, take its

equivalent concept c_r and verify its ancestral concepts in the reference ontology. Take the first ancestral c_{rx} (i.e., the one closest in the hierarchy to c_r) and compare it among the ontology concepts. If there is a match with a concept c_x, then create a *parent-child* relationship between c_x and c. Otherwise, take the next concept c_{rx} (i.e., go one level up in the hierarchy of the reference ontology) and search again among the concepts from the input ontology. Repeat until there is a match between c_x and c_{rx} or until the root element from the reference ontology is reached.

3. If, at the end of the process, the concepts from the input ontology are organized in two or more taxonomies, a common root must be found. Therefore, the top concept c_i from each hierarchy is taken and the reference ontology is searched to find the concept c_{rx} that represents the closest common superclass for all the c_i concepts. If there is such c_{rx} concept and it is not the concept `Thing`, add it to the input ontology and finish the rebuilding by establishing a *parent-child* relationship between c_{rx} and each c_i.

To exemplify how does this process work, suppose that the instance parsing algorithm extracted the following concepts: $Professor$, $GradStudent$, $Employee$ and $Researcher$. Furthermore, suppose we have a reference ontology, such as depicted in Figure 6.5.

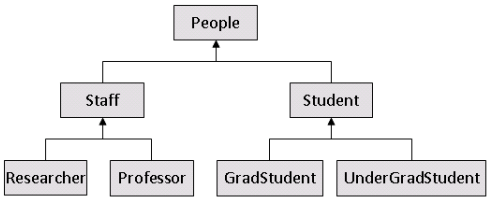

Figure 6.5: Reference ontology

In step one of the taxonomy enrichment algorithm, the goal is to find *parent-child* (denoted as *subclassOf*) relationship between concepts. As result, the following relations are produced [1]:

- `Employee` ***subClassOf*** `Thing`
- `GradStudent` ***subclassOf*** `Thing`
- `Professor` ***subClassOf*** `Employee`
- `Researcher` ***subClassOf*** `Employee`

In step two, the goal is to define *parent-child* (denoted as *subclassOf*) relationship between concepts based on ancestral concepts among the extracted concepts and also present in the reference ontology. In the example no other hierarchical relationships were identified.

Finally, the third step aims at establishing a unique root concept, other than `Thing`, if possible. Figure 6.6 presents the final taxonomy to the concepts of the example. The concept `People` was created (imported from the reference ontology) because it was the superconcept for both `Staff` and `GradStudent`.

[1] we consider here that the concept `Employee` of the input ontology is equivalent to the concept `Staff` of the reference ontology.

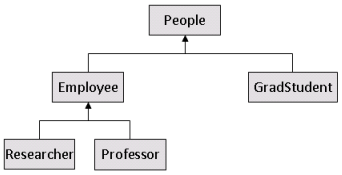

Figure 6.6: Re-built ontology

6.5.2 Hierarchy based on properties

In many cases the reference ontology may not be available. Moreover, even if there is such a reference ontology, it is possible that the concept from the input ontology does not match any concept in the reference ontology. In these cases the reconstruction of the ontology hierarchy with support of a reference ontology is not possible. This limitation is compensated for in the second step of the process of rebuilding the ontology hierarchy, in which we use reverse engineering techniques to infer *parent-child* as well as *sibling* relationships between the concepts. We bring some principles and rules from the field of relational databases to that of an ontology reverse engineering (STOJANOVIC; STOJANOVIC; VOLZ, 2002; ASTROVA, 2004) in order to establish whether either a specialization/generalization - or a sibling - relationship holds between two concepts. These rules depends, basically, on the set of properties associated to each concept.

Let's consider two concepts c_1 and c_2, both extracted by the concept wrapper or defined in the input ontology. Let's now consider that $P(c_1)$ is the set of properties associated to c_1 and $P(c_2)$ the set of properties associated to c_2. Adapting (ASTROVA, 2004), we can define:

- **Property quality**, if $P(c_1) \equiv P(c_2)$, i.e., if all properties associated to c_1 are also associated to c_2 and vice-versa;
- **Property containment**, if $P(c_1) \subset P(c_2)$, i.e., if all properties associated to c_1 are also associated to c_2, but not every property associated to c_2 is also associated to c_1;
- **Property overlap**, if $P(c_1) \cap P(c_2) \neq \emptyset, P(c_1) - P(c_2) \neq \emptyset, P(c_2) - P(c_1) \neq \emptyset$ i.e., if there are some properties associated to c_1 that are also associated to c_2, but also some properties associated to c_1 not associated to c_2 and vice-versa.

The definitions above are used during the execution of the algorithm we propose for defining the ontology hierarchy. The following steps consider the comparison of two concepts c_1 and c_2 without the presence of a reference ontology.

1. Define *parent-child* relationships. Compare the set of properties of c_1 against the set of properties associated to c_2. If there is a property containment relation (i.e., $P(c_1) \subset P(c_2)$) then we define that c_2 is a subclass of $c1$.

2. Eliminate redundant superclasses. Suppose that, in step 1, when comparing $P(c_1)$ against $P(c_2)$ there was a containment relation. Furthermore, when comparing $P(c_1)$ against $P(c_3)$ there was also a containment relation. Therefore, both c_2 and c_3 are defined as subclasses of c_1. Moreover, when comparing $P(c_2)$ against $P(c_3)$, a containment relation was also discovered. Consequently, c_3 is defined as being a subclass of c_2. Analyzing these relationships, we can identify a redundant *parent-child* relationship between c_1 and c_3. As c_2 is a subclass of c_1 and c_3 is a subclass of

c_2, the transitive property of a *parent-child* relationship guarantee that c_3 is also a subclass of c_1. Therefore, we must eliminate the *parent-child* relationship holding between c_1 and c_3, as shown in Figure 6.7.

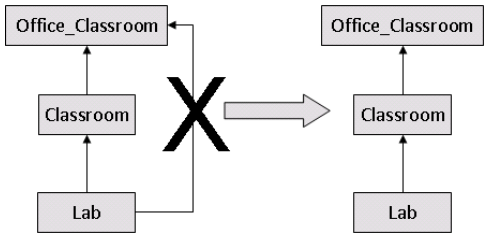

Figure 6.7: Eliminating redundant *parent-child* relationships

3. Define siblings. If no *parent-child* relationship was found in step 1, but $P(c_1)$ overlaps $P(c_2)$ with the equivalent properties representing at least 70% of the properties of both concepts and, moreover, neither c_1 nor c_2 have a superclass other than `Thing` they are considered as siblings. A new concept c_{new} is, thus, created and both c_1 and c_2 are defined as being subclasses of c_{new}. The properties associated to c_{new} are the ones c_1 and c_2 have in common (i.e., $P(c_1) \cap P(c_2)$). The label of c_{new} is given by $label(c_1)\_label(c_2)$, where *label(c)* is a function that returns the name of a concept.

4. Repeat step 3 until there are no more concepts.

For the concepts extracted in the example of Sections 7.3 and 6.4, the axioms representing tge taxonomy produced is encoded in Figure 6.8.

The result of the application of the reverse engineering technique for the example OWL ontology is depicted in Figure 6.9. In Figure 6.10 is presented the respective encoding according to our geographic ontology model.

6.6 Publications

The paper accepted for publication in the International Conference on Advanced Geographic Information Systems & Web Services (HESS; IOCHPE; CASTANO, 2009) presents an overview of the concept wrapper and a short description. The geographic ontology reverse engineering method we propose here was published in details in the ACM GIS 2008 conference (HESS; IOCHPE, 2008).

$$
\begin{aligned}
A \;=\; & isa(Line, Point\_Line\_Polygon) \\
& isa(Polygon, Point\_Line\_Polygon) \\
& isa(Point, Point\_Line\_Polygon) \\
& isa(Employee, Employee\_GradSudent) \\
& isa(GradStudent, Employee\_GradSudent) \\
& isa(Resarcher, Employee) \\
& isa(Professor, Employee) \\
& isa(Office, Office\_Classroom) \\
& isa(Classroom, Office\_Classroom) \\
& isa(Lab, Classroom) \\
& crosses(Road, some(Campus)) \\
& inside(Building, some(Campus)) \\
& inside(Classroom, some(Building)) \\
& inside(Office, some(Building)) \\
& contains(Campus, some(Building)) \\
& contains(Building, some(Office)) \\
& contains(Building, some(Classroom)) \\
& hasGeometry(Campus, Polygon) \\
& hasGeometry(Road, Line) \\
& hasGeometry(Building, Polygon) \\
& hasGeometry(Office\_Classroom, Point)
\end{aligned}
$$

Figure 6.8: Rebuilt ontology hierarchy

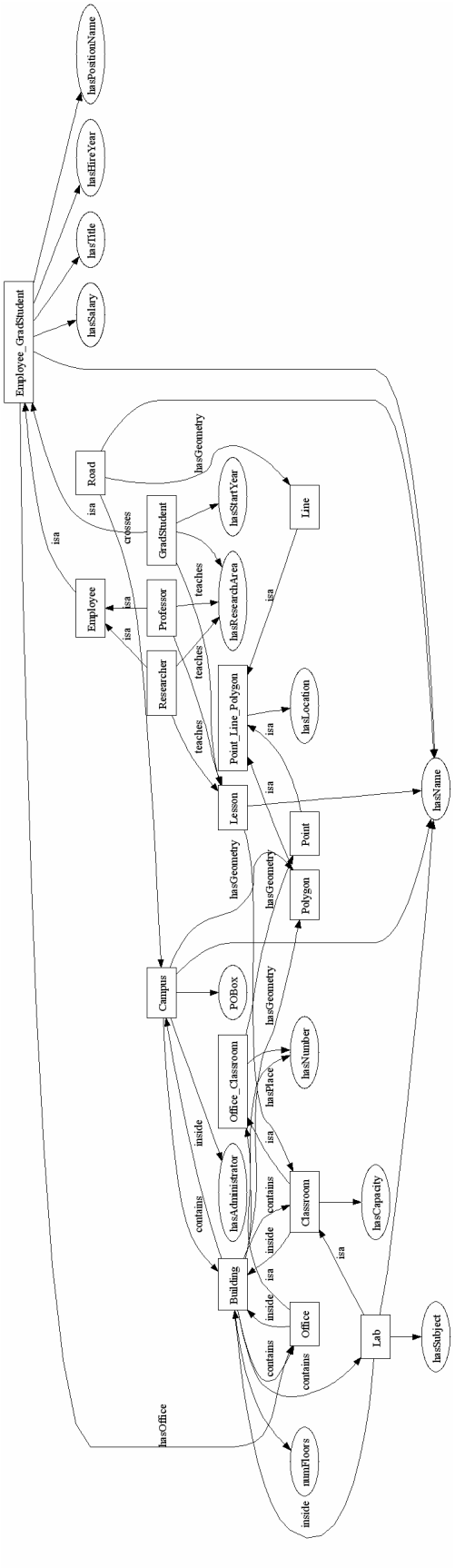

Figure 6.9: Produced ontology

C_g = $Lesson, Employee_GradStudent, Employee, GradStudent$ (domain)
= $Professor, Researcher$ (domain)
$Road, Campus, Building, Office_Classroom$ (geographicdomain)
$Office, Classroom, Lab$ (geographicdomain)
$Point\_Line\_Polygon, Line, Polygon, Point$ (geometry)

P_g = $numFloors, hasCapacity, hasNumber, POBox, hasPlace$ (conventional)
$hasName, hasResearchArea, hasStartYear, hasSalary$ (conventional)
$hasSubject, teaches, hasOffice, hasTitle$ (conventional)
$hasHireYear, hasPositionName, hasAdministration$ (conventional)
$inside, contains, crosses$ (spatial)
$hasGeometry$ (geometric)
$hasLocation$ (positional)

A = $isa(Line, Point\_Line\_Polygon)$
$isa(Polygon, Point\_Line\_Polygon)$
$isa(Point, Point\_Line\_Polygon)$
$isa(Employee, Employee\_GradSudent)$
$isa(GradStudent, Employee\_GradSudent)$
$isa(Resarcher, Employee)$
$isa(Professor, Employee)$
$isa(Office, Office\_Classroom)$
$isa(Classroom, Office\_Classroom)$
$isa(Lab, Classroom)$
$crosses(Road, some(Campus))$
$inside(Building, some(Campus))$
$inside(Classroom, some(Building))$
$inside(Office, some(Building))$
$contains(Campus, some(Building))$
$contains(Building, some(Office))$
$contains(Building, some(Classroom))$
$hasGeometry(Campus, Polygon)$
$hasGeometry(Road, Line)$
$hasGeometry(Building, Polygon)$
$hasGeometry(Office\_Classroom, Point)$

I_g =
M =

Figure 6.10: Rebuilt ontology structure

7 IG-MATCH SOFTWARE ARCHITECTURE

In order to evaluate the algorithms and metrics developed in this dissertation, a software architecture called IG-MATCH was designed and implemented. IG-MATCH was designed as a three-layer software architecture, namely concept wrapper, concept matcher and instance matcher layer respectively, as depicted in Figure 7.1. Each one of these layers implements one of the proposed algorithms for the geographic ontology matching process. Furthermore, these three layers are independent from one another, which means that it is possible to use them separately. For example, if the user wants only to match ontologies at the concept-level, he/she can use only the concept matcher layer.

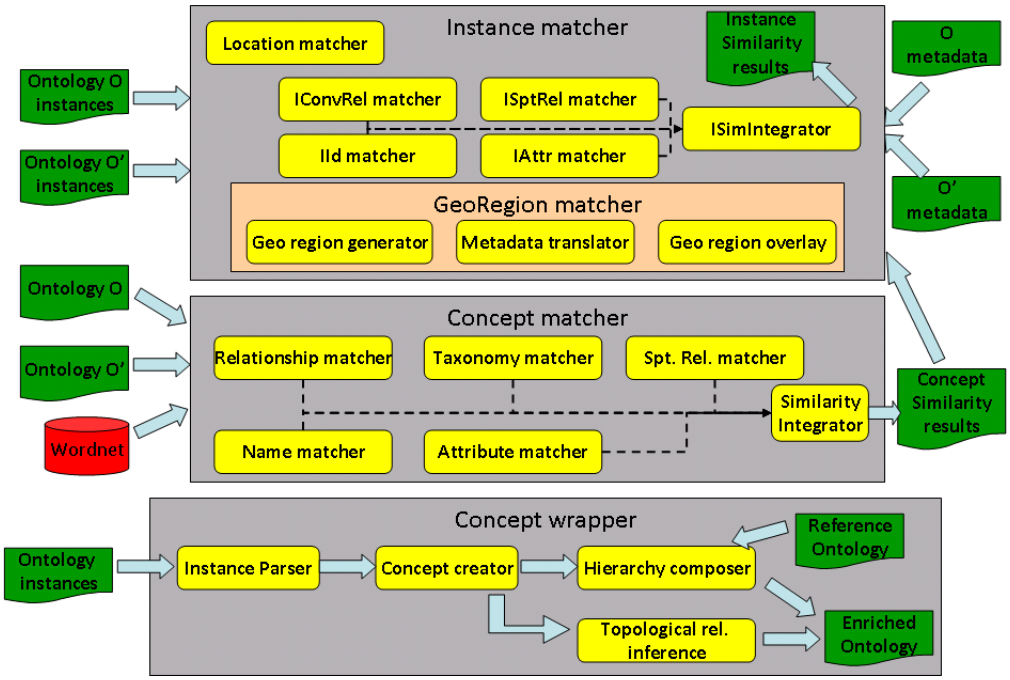

Figure 7.1: IG-MATCH architecture

Considering IG-MATCH as a whole, the ontology matching process starts by taking as input two geographic ontologies O and $O^{'}$ to be matched. The concept wrapper layer, which implements the concept wrapper algorithm, is the first one to be executed. The goal is to enrich ontologies O and $O^{'}$ with, at least, topological relationships and parent-child relationships as well.

The next step is the execution of the concept matching algorithm, which is implemented in the concept matcher layer. In this phase the concepts defined in the two input ontologies are compared. A similarity value is assessed for each pair of concepts $(gc, gc^{'})$,

or $(c, c^{'})$ if they are not geographic. The output of the concept matcher layer is a list of pairs of matching (equivalent) concepts. Each entry of this list is a pair of concepts $c, c^{'}$. Furthermore, for each pair of matching concepts, a list of equivalent properties of these two concepts is produced. Each entry of this list is a pair of properties $p, p^{'}$.

The instances from the two ontologies O and $O^{'}$ together with the concepts equivalence list and the metadata of both ontologies constitute the input for the instance matcher layer, which implements the instance matching algorithm. Only instances belonging to concepts previously identified as equivalent are compared. In other words, two instances gi and $gi^{'}$, or i and $i^{'}$ if they are not geographic, are compared only if the concepts gc, or c, and $gc^{'}$, or $c^{'}$, they, respectively, instantiate were identified as matching in the concept matching phase or manually informed by the user. The output produced by the instance matcher is a list of matching (equivalent) pairs of instances.

Figure 7.2 presents the UML activity diagram for the overall process executed by iG-MATCH.

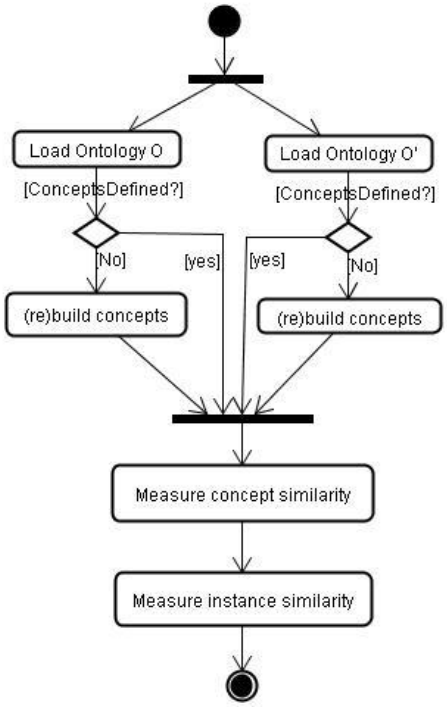

Figure 7.2: iG-MATCH general UML activity diagram

iG-MATCH was implemented in Java, using the Protege OWL API for reading and parsing the ontologies. Two other APIs were imported for measuring the similarity regarding concepts, instances and attributes names: the JWNL API was imported to measure the linguistic affinity using WordNet (MILLER, 1995) and the SIMMETRICS API for the string-distance metric we adapted. The Oracle 10g XE (eXpress Edition) was used to store the results of the matching process, especially regarding the ontology's spatial characteristics. It allows the creation of a column of type `sdo_geometry` to store the spatial position (coordinates) of the data. Furthermore, it provides the Oracle Spatial Java API (sdoapi), which has a number of methods to perform spatial operations over the data. We basically used the method which verifies if two geometries overlap or not. We used that operation for checking the intersection area of two *GeoRegions* as well as to check if two given instances have or do not have overlapping areas, as described in Chapter 5.

7.1 Concept-level layer

The concept matcher layer implements the concept matching algorithm and respective mathematical expressions proposed in Chapter 4. Figure 7.3 presents the UML deployment diagram of the concept matcher layer.

The inputs for this layer are the two ontologies O and $O^{'}$ to be matched and an external thesaurus or ontology. In this implementation, we exploit the WordNet lexical system (MILLER, 1995). The output generated by the concept matcher layer is a list of equivalent pairs of concepts (gc,$gc^{'}$), or (c,$c^{'}$) if they are not geographic, where gc (or c) belongs to O and $gc^{'}$ (or $c^{'}$) belongs to $O^{'}$.

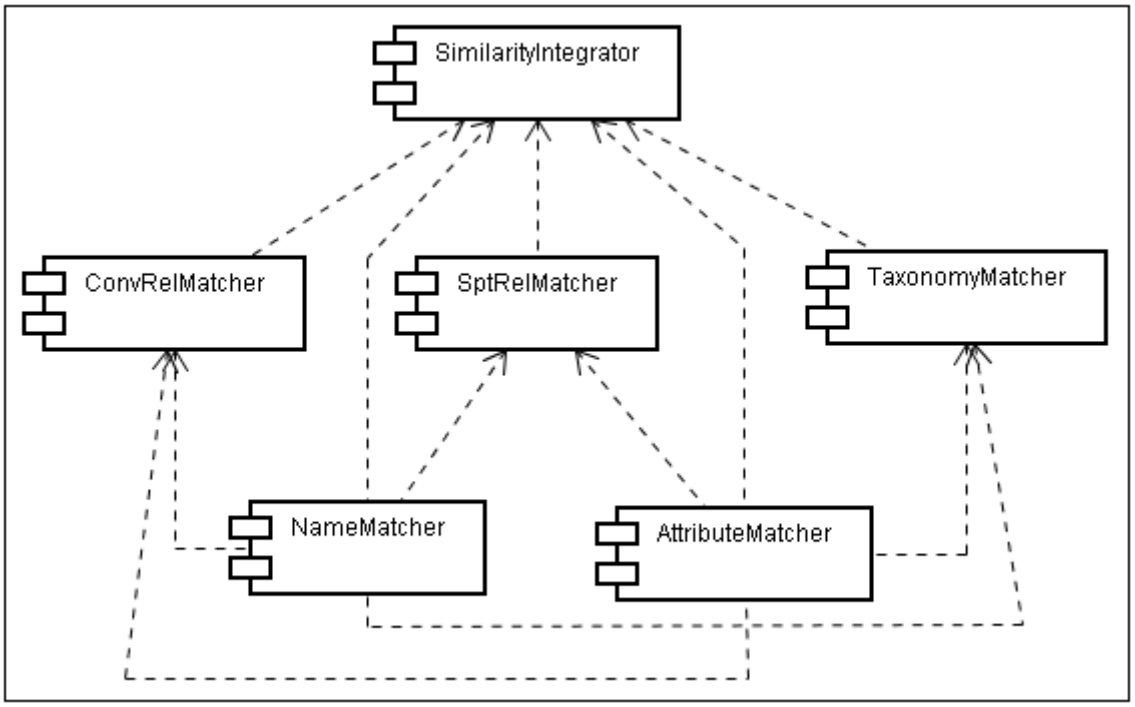

Figure 7.3: Concept matcher layer

Internally, the concept matcher layer is composed by 6 modules, which are part of the algorithm presented in Chapter 4. The *NameMatcher* module implements the part of the algorithm and the metrics for measuring the name similarity between two given concepts. The *AttributeMatcher* is responsible for the part of the algorithm which has the metrics for the matching of properties representing attributes. The *TaxonomyMatcher* module performs the part of the concept-level algorithm for the similarity measurement of the hierarchies in which the two compared concepts are comprised. Spatial relationships and conventional relationships are addressed by the part of the matching algorithm which is implemented, respectively, in the *SptRelMatcher* and *ConvRelMatcher* modules. Finally, the *SimilarityIntegrator* module computes the overall similarity value.

As the concept-matcher algorithm is a two-phase algorithm, in the first phase the *NameMatcher* and the *AttributeMatcher* modules are executed. The *TaxonomyMatcher*, *SptRelMatcher* and *ConvRelMatcher* modules are executed in the second phase of the algorithm. The results produced by the first phase are used as parameters in the second phase.

7.2 Instance-level layer

The instance matcher layer implements the instance-level matching algorithm and respective mathematical expressions proposed in Chapter 5. Figure 7.4 depicts the UML deployment diagram for the instance matcher layer.

The inputs for this layer are the two sets of ontology instances to be matched as well as the metadata from ontologies O and $O^{'}$, and the concept-level matching results. The latter is the set of equivalent pairs of concepts $(gc, gc^{'})$, or $(c, c^{'})$. The output of the instance matcher layer is a list of equivalent pairs of instances (gi,$gi^{'}$), or (i,$i^{'}$) if they are not geographic, where gi (or i) belongs to O and $gi^{'}$ (or $i^{'}$) belongs to $O^{'}$.

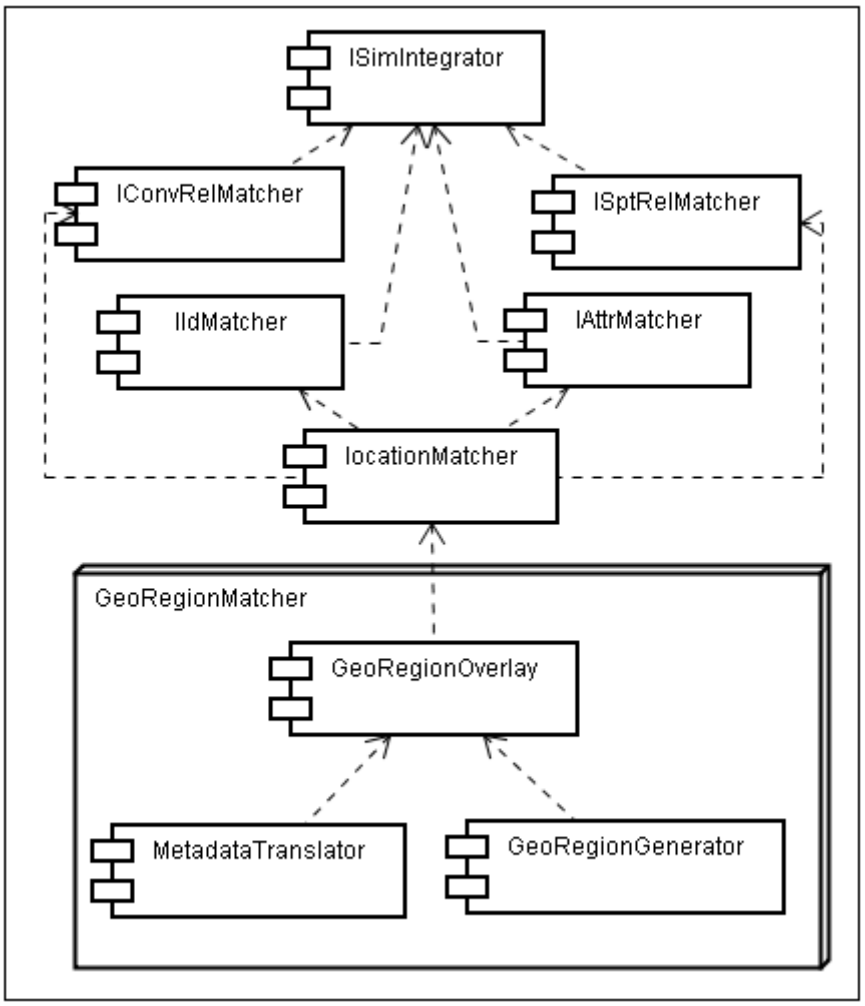

Figure 7.4: Instance matcher layer

The instance layer architecture, is composed by nine modules. The *GeoRegionGenerator*, *MetadataTranslator* and *GeoRegionOverlay* modules compose a sub-layer called *GeoRegionMatcher*, which implements the algorithm for the geographic context region creation and elimination of geographic isolated instances.This algorithm is detailed in Section 5.2.

The actual instance matching algorithm and metrics are implemented in the *IIdMatcher*, *IAttrMatcher*, *IConvRelMatcher*, *ISpatialRelMatcher*, *LocationMatcher* and *ISimIntegrator* modules. The first four implement, respectively, the instance identifier

matching, the attributes values matching, the conventional relationships property values matching and the spatial relationships property values matching. The *LocationMatcher* implements the part of the algorithm which deals with the similarity measurement regarding the instances' spatial position and, finally, the overall instance similarity computation is implemented in the *ISimIntegrator* module. These modules are used in the instance-level matching algorithm, presented in Chapter 5.

7.3 Concept wrapper layer

The concept wrapper layer is IG-MATCH's layer which implements the concept wrapper algorithm presented in Chapter 6. The goal is to enrich the ontology with topological relationships as well as with the inference of new, implicit, taxonomic relationships. Furthermore, it can also (re)build the ontology structure (concept and properties) by analyzing the ontology's instances. The UML deployment diagram fo Figure 7.5 presents the concept wrapper layer components.

The inputs for this layer are the set of instances of an ontology and, if available, a reference ontology. The output an OWL file corresponding to the ontology structure (definition), rebuilt or, at least, enriched with hierarchical relationships and topological relationships as well.

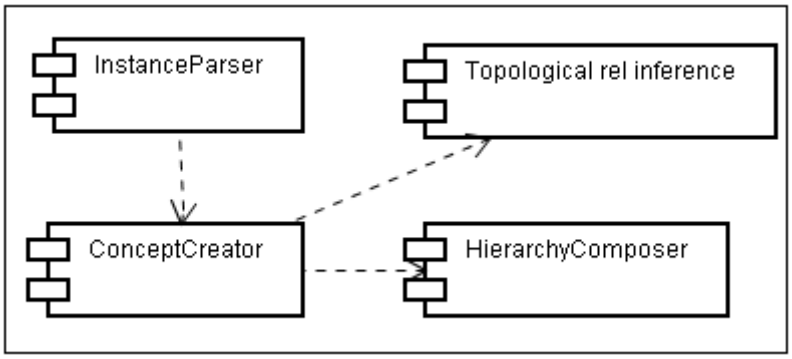

Figure 7.5: Concept wrapper modules

The concept wrapper is implemented in the conceptWrapper layer, which is composed by four modules, namely *InstanceParser*, *ConceptCreator*, *Topological rel. inference* and *HierarchyComposer*. The input for this layer is the set of instances of an ontology an, if available, a reference ontology to aid in the process of composing the ontology's taxonomy.

The *InstanceParser* and *ConceptCreator* modules implement the part of the concept wrapper algorithm which extracts, from the ontology's instances, the concepts and the properties associated to their contexts. The *Topological rel. inference* module implements the algorithm for dicovering topological relationships and, at last, the *HierarchyComposer* module implements the part of the algorithm which is responsible for inferring the implicit parent-child and sibling relationships among the concepts.

7.4 Publications

In the International Spatial and Temporal Databases (SSTD) 2007 (HESS; IOCHPE; CASTANO, 2007c) a first version of IG-MATCH was presented, in a two layer-

architecture. The concept-level and instance-level matchers were described. The paper accepted for publication in the The International Conference on Advanced Geographic Information Systems & Web Services (HESS; IOCHPE; CASTANO, 2009) also presented IG-MATCH as a whole, including the concept wrapper layer.

8 CONCLUSIONS

Interchanging geographic information is an actual issue nowadays. The semantic web is increasing the interest as well as research on ontologies. When integrating geographic ontologies, one of the steps is the identification of semantically equivalent pieces of information, at the concept-level and instance-level as well. This implies the existence of proved techniques for measuring the similarity between geographic ontologies consisting of concepts and instances in a process called matching. In this Ph.D. dissertation we proposed algorithms and mathematical expressions (metrics) to match geographic ontologies.

Because of the lack of a widely accepted standard model for describing geographic ontologies, the first contribution done in this work was the proposal of a geographic ontology model. Based on existing data models and frameworks for conceptual modeling of geographic databases as well as on the Open GIS Consortium recommendation, we defined a (meta-) model with concepts as well as properties, and axioms to describe and create geographic ontologies. The definition of a geographic ontology model was mandatory in order to homogenize the way the ontologies are described and, therefore, make the matching process possible.

The main contributions of this research are the algorithms and metrics tailor made for geographic ontology matching, at the concept-level as well as at the instance-level. The concept matching algorithm is novel because it combines conventional, non-geographic features with specific features describing geographic phenomena. The majority of the existing approaches for geographic ontology matchers basically consider the concept label, the ontology taxonomy and, or geographic properties, such as topological relationships and geometry, or conventional properties, such as attributes and relationships. Furthermore, the tailor made metrics defined in this work and the way they are combined in a balanced sum are also part of the contribution.

Probably, the instance-level matcher layer is the part of this research that contains a larger number of novelties. To the best of our knowledge, is the first attempt to consider the metadata when measuring semantic similarity of geographic data. Although it seems simple, none of the previous matchers considered that the input instances could be described using different perspectives (for example, coordinate system, scale, projection system) and, thus, a translation must be performed prior to actually comparing the data. If one thinks that the ontologies may come from different sources and organizations, this is a situation that can easily happen. It is also a contribution the introduction of a *Geographic Region* concept to accelerate the matching process. As instances of an ontology usually represent individuals from an enclosed region, by first eliminating the instances that are outside the common geographic area covered by both ontologies, it is possible to save time preventing non-useful comparisons.

An additional contribution is the integration of property values in the matching process. Most of existing approaches consider only the spatial location or, at most, the spatial location and the instance label. Our proposal, in addition, considers the attributes values and relationships ranges as well. Finally, our proposal is not limited to only one type of geometry, or requires the instances matching to be associated with the same geometry. It transforms the instances to the same geometry by reducing them to point or creates MBRs to generate polygons.

Although not in a large scale, the tests we run showed that our algorithms and metrics are a potential good proposal for geographic ontology matching especially for middle-large to large ontologies. The results obtained with the tests, when compared to a human-centered approach were very satisfactory. Furthermore, when comparing the new matching algorithms with available non-geographic matchers, it did achieve better results with geographic ontologies. Unfortunately, the proposals for the geographic matchers referenced in this research did not have a prototype tool available for comparison.

Another contribution of this research is the creation of a technique to enrich geographic ontologies, by inferring topological relationships from the instances locations and discovering taxonomic relationships using reverse engineering techniques. It parses the ontology's instances property values to discover implicit topological relationships, *parent-child* as well as *sibling* relationships. Furthermore, it also translates the input ontologies into the geographic ontology model we propose, by creating the geometry concepts, and the `hasLocation` and the `hasGeometry` properties when needed. As a side effect, it also allows the complete (re)construction of the concepts of an ontology from the OWL instances tags.

Finally, to evaluate the proposed algorithms, we designed a software architecture called IG-MATCH. It consists in a three-layer software system which implements the concept-level matching algorithm as well as the instance-level matching algorithm, and the geographic ontology reverse engineering algorithm. Each one of the algorithms was implemented in one separate layer.

One of the main difficulties faced during the production of this research is related to the lack of a standard ontology model for describing geographical resources, especially the spatial components. To overcome these limitations we proposed a geographic ontology model specially designed for matching purposes, to which the ontologies are converted before starting the matching process. At the moment, this ontology model is expressive enough to describe all the features we are considering in our matching algorithms.

Another difficulty so far is the lack of a technique to properly set the parameters to balance the overall similarity measures, at both concept and instance-levels. As this tuning depends on a number of variables, as discussed in Chapter 4, we did not yet manage to get an automatic weight combination.

8.1 Future work

During the research effort that lead to the results achieved in this work, some obstacles have been identified that need more time to be investigated properly:

1. **Intelligent weights tuning**: At the concept-level as well as at the instance-level, the overall similarity is measured in a balanced sum. However, depending on the features of the input ontologies, i.e., depth of the taxonomy, number of data type properties, number of object type properties, etc., a different weight combination

may produce better matching results. Therefore, to develop a technique to (semi-) automatic tune this weights is planned for the future.

2. **To perform more tests**: So far we performed a limited number of tests. Therefore, more tests, at both the concept and instance-levels have to be performed. Tests with large ontologies, with large sets of instances are of special interest.

3. **Metadata implementation**: The algorithm that performs the matching at the instance-level predicts the metadata translation to prevent incorrect comparisons. However, it is not yet implemented. Currently, if needed, this translation is done manually.

4. **Extend to moving objects**: Moving objects is an emerging interest area to the GIS community. Therefore, the extension/adaptation of the algorithms as well as the development of specific mathematical expressions to deal with the particularities of the moving objects particular features is also planned for the future.

5. **Extend the algorithms to cover temporality**: Although not exclusive for the geographic field, temporality is an intrinsic feature of many geographic phenomena. Presently, neither the geographic ontology model nor the proposed matching algorithms deal with it. Future works may extend the geographic ontology model to represent temporality and also the algorithms to consider its effects when measuring the similarity between two concepts or two instances. Once again, the development of specific metrics to deal with temporality is needed.

REFERENCES

ALEKSOVSKI, Z. et al. Matching Unstructured Vocabularies Using a Background Ontology. In: INTERNATIONAL CONFERENCE - MANAGING KNOWLEDGE IN A WORLD OF NETWORKS, EKAW, 15., 2006, Podebrady, Czech Republic. **Proceedings...** Berlin: Springer, 2006. p.182–197. (Lecture Notes in Computer Science, v.4248).

ARONOFF, S. **Geographic Information Systems**: a management perspective. [S.l.]: WDL Publications, 1991.

ARPINAR, B. et al. Geospatial Ontology Development and Semantic Analytics. **Transactions in GIS**, [S.l.], v.10, n.4, p.551–575, July 2006.

ASTROVA, I. Reverse Engineering of Relational Databases to Ontologies. In: EUROPEAN SEMANTIC WEB SYMPOSIUM, ESWS, 1., 2004, Heraklion, Crete, Greece. **The Semantic Web**: Research and Applications: Proceedings. Berlin: Springer, 2004. p.327–341. (Lecture Notes in Computer Science, v.3053).

ASTROVA, I.; STANTIC, B. An HTML-Form-Driven Approach to Reverse Engineering of Relational Databases to Ontologies. In: IASTED INTERNATIONAL CONFERENCE ON DATABASES AND APPLICATIONS, 2005, Innsbruck, Austria. **Proceedings...** [S.l.]: IASTED/ACTA Press, 2005. p.246–251.

BAGLIONI, M. et al. Building Geospatial Ontologies from Geographical Databases. In: INTERNATIONAL CONFERENCE ON GEOSPATIAL SEMANTICS, GEOS, 2., 2007, Mexico City, Mexico. **Proceedings...** Berlin: Springer, 2007. p.195–209. (Lecture Notes in Computer Science, v.4853).

BEERI, C. et al. Finding corresponding objects when integrating several geo-spatial datasets. In: ACM INTERNATIONAL WORKSHOP ON GEOGRAPHIC INFORMATION SYSTEMS, ACM-GIS, 13., 2005, Bremen, Germany. **Proceedings...** New York: ACM, 2005. p.87–96.

BELUSSI, A.; CATANIA, B.; PODESTÀ, P. Towards topological consistency and similarity of multiresolution geographical maps. In: ACM INTERNATIONAL WORKSHOP ON GEOGRAPHIC INFORMATION SYSTEMS, ACM-GIS, 13., 2005, Bremen, Germany. **Proceedings...** New York: ACM, 2005. p.220–229.

BENSLIMANE, S. M. et al. Acquiring owl ontologies from data-intensive web sites. In: INTERNATIONAL CONFERENCE ON WEB ENGINEERING, ICWE, 6., 2006, Palo Alto, CA, USA. **Proceedings...** New York: ACM, 2006. p.361–368.

BISHR, Y. et al. Probing the Concepts of Information Communities: A First Step Towards Semantic Interoperability. In: GOODCHILD, M. et al. (Ed.). **Interoperating Geographic Information Systems**. [S.l.]: Kluwer-Academic, 1999. p.55–69.

BITTNER, T.; SMITH, B. Granular Spatio-Temporal Ontologies. In: AAAI SPRING SYMPOSIUM ON FOUNDATIONS AND APPLICATIONS OF SPATIO-TEMPORAL REASONING, FASTR, 2003. **Proceedings...** [S.l.: s.n.], 2003.

BORGES, K. A. V.; DAVIS, C. A.; LAENDER, A. H. F. OMT-G: an object-oriented data model for geographic applications. **GeoInformatica**, [S.l.], v.5, n.3, p.221–260, 2001.

BRODEUR, J.; BÉDARD, Y.; MOULIN, B. A geosemantic proximity-based prototype for the interoperability of geospatial data. **Computers, Environment and Urban Systems**, [S.l.], v.29, n.6, p.669–698, 2005.

CASATI, R.; SMITH, B.; VARZI, A. C. Ontological Tools for Geographic Representation. In: FORMAL ONTOLOGY IN INFORMATION SYSTEMS, 1998. **Proceedings...** [S.l.]: IOS Press, 1998. p.77–85.

CASTANO, S.; FERRARA, A.; MONTANELLI, S. Matching Ontologies in Open Networked Systems: techniques and applications. **Journal of Data Semantics V**, [S.l.], p.25–63, 2006.

CHAUDHRI, V. K. et al. OKBC: a programmatic foundation for knowledge base interoperability. In: AAAI/IAAI, 1998. **Proceedings...** [S.l.: s.n.], 1998. p.600–607.

CRUZ, I. F.; SUNNA, W.; CHAUDHRY, A. Semi-automatic Ontology Alignment for Geospatial Data Integration. In: INTERNATIONAL CONFERENCE ON GEOGRAPHIC INFORMATION SCIENCE, GISCIENCE, 3., 2004, Adelphi, MD, USA. **Proceedings...** Berlin: Springer, 2004. p.51–66. (Lecture Notes in Computer Science, v.3234).

DOBRE, A.; HAKIMPOUR, F.; DITTRICH, K. R. Operators and Classification for Data Mapping in Semantic Integration. In: INTERNATIONAL CONFERENCE ON CONCEPTUAL MODELING, ER, 22., 2003, Chicago, IL, USA. **Proceedings...** Berlin: Springer, 2003. p.534–547. (Lecture Notes in Computer Science, v.2813).

DUCKHAM, M.; WORBOYS, M. F. An algebraic approach to automated geospatial information fusion. **International Journal of Geographical Information Science**, [S.l.], v.19, n.5, p.537–557, 2005.

EGENHOFER, M. J.; FRANZOSA, R. D. Point Set Topological Relations. **International Journal of Geographical Information Systems**, [S.l.], v.5, p.161–174, 1991.

FAGIN, R. et al. Composing schema mappings: second-order dependencies to the rescue. **ACM Trans. Database Syst.**, New York, v.30, n.4, p.994–1055, 2005.

FONSECA, F. et al. Using ontologies for integrated geographic information systems. **Transactions in Geographic Information Systems**, [S.l.], v.6, n.3, 2002.

FONSECA, F.; MARTIN, J. Learning the Differences Between Ontologies and Conceptual Schemas Through Ontology-Driven Information Systems. **Journal of the Association for Information Systems (JAIS) - Special Issue on Ontologies in the Context of IS**, [S.l.], v.8, n.2, 2007.

FONSECA, F. T.; DAVIS, C. A.; CAMARA, G. Bridging Ontologies and Conceptual Schemas in Geographic Information Integration. **GeoInformatica**, [S.l.], v.7, n.4, p.355–378, 2003.

FORNARI, M. R.; COMBA, J. L. D.; IOCHPE, C. Query optimizer for spatial join operations. In: ACM INTERNATIONAL SYMPOSIUM ON GEOGRAPHIC INFORMATION SYSTEMS, ACM-GIS, 14., 2006, Arlington, Virginia, USA. **Proceedings...** New York: ACM, 2006. p.219–226.

FRANK, A. U. Qualitative Spatial Reasoning about Distances and Directions in Geographic Space. **Journal of Visual Languages and Computing**, [S.l.], v.3, p.343–371, 1992.

FU, G.; JONES, C. B.; ABDELMOTY, A. I. Building a Geographical Ontology for Intelligent Spatial Search on the Web. **Databases and Applications (DBA2005)**, [S.l.], p.167–172, 2005.

GIUNCHIGLIA, F.; SHVAIKO, P.; YATSKEVICH, M. S-Match: an algorithm and an implementation of semantic matching. In: SEMANTIC INTEROPERABILITY AND INTEGRATION, 2005. **Proceedings...** Germany: IBFI: Schloss Dagstuhl, 2005. (Dagstuhl Seminar Proceedings, v.04391).

GRUBER, T. R. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In: FORMAL ONTOLOGY IN CONCEPTUAL ANALYSIS AND KNOWLEDGE REPRESENTATION, 1993, Deventer, The Netherlands. **Proceedings...** [S.l.]: Kluwer Academic Publishers, 1993.

GUARINO, N. Understanding and building, using ontologies. **Int. J. Hum.-Comput. Stud.**, [S.l.], v.46, n.2, p.293–310, 1997.

HAKIMPOUR, F.; GEPPERT, A. Global Schema Generation Using Formal Ontologies. In: INTERNATIONAL CONFERENCE ON CONCEPTUAL MODELING, ER, 21., 2002, Tampere, Finland. **Proceedings...** Berlin: Springer, 2002. p.307–321. (Lecture Notes in Computer Science, v.2503).

HARIHARAN, R.; SHMUELI-SCHEUER, M.; LI, C.; MEHROTRA, S. Quality-driven approximate methods for integrating GIS data. In: ACM INTERNATIONAL WORKSHOP ON GEOGRAPHIC INFORMATION SYSTEMS, ACM-GIS, 13., 2005, Bremen, Germany. **Proceedings...** New York: ACM, 2005. p.97–104.

HESS, G. N.; IOCHPE, C. Syntactic and Semantic GDB Conceptual Schemas Integration. In: AGILE CONFERENCE ON GISCIENCE, AGILE, 8., 2005, Estoril, Portugal. **Proceedings...** Portugal: Instituto Geográfico Português, 2005. p.271–280.

HESS, G. N.; IOCHPE, C. Geo-ontology enrichment through reverse engineering. In: ACM SIGSPATIAL INTERNATIONAL CONFERENCE ON ADVANCES IN GEOGRAPHIC INFORMATION SYSTEMS, ACM-GIS, 16., 2008, Irvine, CA, USA. **Proceedings...** New York: ACM, 2008.

HESS, G. N.; IOCHPE, C.; CASTANO, S. An Algorithm and Implementation for GeoOntologies Integration. In: BRAZILIAN SYMPOSIUM ON GEOINFORMATICS, 8., 2006, Campos do Jordão, SP, BRAZIL. **Proceedings...** São Paulo: INPE, 2006. p.109–120.

HESS, G. N.; IOCHPE, C.; CASTANO, S. Towards a Geographic Ontology Reference Model for Matching Purposes. In: BRAZILIAN SYMPOSIUM ON GEOINFORMATICS, 9., 2007, Campos do Jordão, SP, BRAZIL. **Proceedings...** São Paulo: INPE, 2007. p.35–47.

HESS, G. N.; IOCHPE, C.; CASTANO, S. An algorithm and implementation for geoontologies alignment. In: **Advances in geoinformatics**. Berlin: Springer-Verlag, 2007. p.151–164.

HESS, G. N.; IOCHPE, C.; CASTANO, S. Geographic Ontology Matching with iG-Match. In: INTERNATIONAL SYMPOSIUM ON SPATIAL AND TEMPORAL DATABASES, SSTD, 10., 2007, Boston, MA, USA. **Advances in Spatial and Temporal Databases**: Proceedings. Berlin: Springer, 2007. p.185–202. (Lecture Notes in Computer Science).

HESS, G. N.; IOCHPE, C.; CASTANO, S. IG-MATCH : towards effective geo-ontology matching. To appear in the International Conference on Advanced Geographic Information Systems & Web Services, GEOWS, 2009, Cancun, Mexico.

HESS, G. N.; IOCHPE, C.; FERRARA, A.; CASTANO, S. Towards Effective Geographic Ontology Matching. In: INTERNATIONAL CONFERENCE ON GEOSPATIAL SEMANTICS, GEOS, 2., 2007, Mexico City, Mexico. **Proceedings...** Berlin: Springer, 2007. p.51–65. (Lecture Notes in Computer Science, v.4853).

KALFOGLOU, Y.; SCHORLEMMER, M. Ontology mapping: the state of the art. **Knowledge Engineering Review**, New York, NY, USA, v.18, n.1, p.1–31, 2003.

KAVOURAS, M.; KOKLA, M.; TOMAI, E. Comparing categories among geographic ontologies. **Computers & Geosciences**, [S.l.], v.31, n.2, p.145–154, March 2005.

KLIEN, E. et al. An Architecture for Ontology-Based Discovery and Retrieval of Geographic Information. In: AGILE CONFERENCE ON GISCIENCE, AGILE, 7., 2004, Heraklion, Greece. **Proceedings...** [S.l.: s.n.], 2004.

KOKLA, M.; KAVOURAS, M.; TOMAI, E. Semantic Information in Geo-Ontologies: extraction, comparison and reconciliation. **Journal on Data Semantics III**, [S.l.], v.3534, p.125–142, 2005.

KOLAS, D.; DEAN, M.; HEBELER, J. Geospatial semantic Web: architecture of ontologies. In: IEEE AEROSPACE CONFERENCE, 2006. **Proceedings...** [S.l.: s.n.], 2006.

KUHN, W. Modeling the Semantics of Geographic Categories through Conceptual Integration. In: INTERNATIONAL CONFERENCE ON GEOGRAPHIC INFORMATION SCIENCE, GISCIENCE, 2., 2002, Boulder, CO, USA. **Proceedings...** Berlin: Springer, 2002. p.108–118. (Lecture Notes in Computer Science, v.2478).

LI, M.; DU, X.-Y.; WANG, S. Learning ontology from relational database. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING AND CYBERNETICS, 2005. **Proceedings...** [S.l.: s.n.], 2005.

LISBOA FILHO, J.; IOCHPE, C. Specifying Analysis Patterns for Geographic Databases on the Basis of a Conceptual Framework. In: INTERNATIONAL SYMPOSIUM ON ADVANCES IN GEOGRAPHIC INFORMATION SYSTEMS, ACM-GIS, 7., 1999, Kansas City, USA. **Proceedings...** New York: ACM, 1999. p.7–13.

MAEDCHE, A.; STAAB, S. Semi-automatic Engineering of Ontologies from Text. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING AND KNOWLEDGE ENGINEERING, SEKE, 12., 2000. **Proceedings...** [S.l.: s.n.], 2000.

MANOAH, S.; BOUCELMA, O.; LASSOUED, Y. Schema Matching in GIS. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE: METHODOLOGY, SYSTEMS, AND APPLICATIONS, AIMSA, 11., 2004, Varna, Bulgaria. **Proceedings...** Berlin: Springer, 2004. p.500–509. (Lecture Notes in Computer Science, v.3192).

MILLER, G. A. WordNet: a lexical database for english. **Communications of the ACM**, [S.l.], v.38, n.11, p.39–41, 1995.

NAVARRETE, T.; BLAT, J. An Algorithm for Merging Geographic Datasets Based on the Spatial Distributions of Their Values. In: INTERNATIONAL CONFERENCE ON GEOSPATIAL SEMANTICS, GEOS, 2., 2007, Mexico City, Mexico. **Proceedings...** Berlin: Springer, 2007. p.66–81. (Lecture Notes in Computer Science, v.4853).

NOY, N. F. Tools for Mapping and Merging Ontologies. In: STAAB, S.; STUDER, R. (Ed.). **Handbook on Ontologies**. [S.l.]: Springer, 2004. p.365–384. .

OGC. **GO-1 Application Objects (document OGC 03-064r10)**. [S.l.]: Open GIS Consortium, 2005. Available at <http://portal.opengeospatial.org>. Visited on: 2008.

QUIX, C. et al. Matching Schemas for Geographical Information Systems Using Semantic Information. In: SEMANTIC BASED GEOGRAPHIC INFORMATION SYSTEMS, SEBGIS, 2006. **Proceedings...** Berlin: Springer, 2006. p.1566–1575.

RAHM, E.; BERNSTEIN, P. A. A survey of approaches to automatic schema matching. **VLDB Journal: Very Large Data Bases**, [S.l.], v.10, n.4, p.334–350, 2001.

RODRIGUEZ, M. A.; EGENHOFER, M. J. Determining Semantic Similarity among Entity Classes from Different Ontologies. **IEEE Trans. Knowl. Data Eng.**, [S.l.], v.15, n.2, p.442–456, 2003.

RODRÍGUEZ, M. A.; EGENHOFER, M. J. Comparing geospatial entity classes: an asymmetric and context-dependent similarity measure. **International Journal of Geographical Information Science**, [S.l.], v.18, n.3, p.229–256, 2004.

SCHARFFE, F.; BRUIJN, J. de. A language to specify mappings between ontologies. In: INTERNATIONAL CONFERENCE ON SIGNAL-IMAGE TECHNOLOGY AND INTERNET-BASED SYSTEMS, SITIS, 1., 2005, Yaounde, Cameroon. **Proceedings...** [S.l:]: Dicolor Press, 2005. p.267–271.

SCHWERING, A.; RAUBAL, M. Spatial Relations for Semantic Similarity Measurement. In: ER WORKSHOPS AOIS, BP-UML, COMOGIS, ECOMO, AND QOIS, 2005, Klagenfurt, Austria. **Perspecives in Conceptual Modeling**: Proceedings. Berlin: Springer, 2005. p.259–269. (Lecture Notes in Computer Science, v.3770).

SCHWERING, A.; RAUBAL, M. Measuring Semantic Similarity Between Geospatial Conceptual Regions. In: INTERNATIONAL CONFERENCE ON GEOSPATIAL SEMANTICS, GEOS, 1., 2005, Mexico City, Mexico. **Proceedings...** Berlin: Springer, 2005. p.90–106. (Lecture Notes in Computer Science, v.3799).

SEHGAL, V.; GETOOR, L.; VIECHNICKI, P. Entity resolution in geospatial data integration. In: ACM INTERNATIONAL SYMPOSIUM ON GEOGRAPHIC INFORMATION SYSTEMS, ACM-GIS, 14., 2006, Arlington, Virginia, USA. **Proceedings...** New York: ACM, 2006. p.83–90.

SOTNYKOVA, A.; CULLOT, N.; VANGENOT, C. Spatio-temporal Schema Integration with Validation: a practical approach. In: OTM CONFEDERATED INTERNATIONAL WORKSHOPS AND POSTERS, AWESOME, CAMS, GADA, MIOS+INTEROP, ORM, PHDS, SEBGIS, SWWS, AND WOSE, 2005, Agia Napa, Cyprus. **On the Move to Meaningful Internet Systems**: Proceedings. Berlin: Springer, 2005. p.1027–1036. (Lecture Notes in Computer Science, v.3762).

SOTNYKOVA, A. et al. Semantic Mappings in Description Logics for Spatio-temporal Database Schema Integration. **Journal on Data Semantics III**, [S.l.], p.143–167, 2005.

SOUZA, D.; SALGADO, A. C.; TEDESCO, P. A. Towards a Context Ontology for Geospatial Data Integration. In: OTM CONFEDERATED INTERNATIONAL WORKSHOPS AND POSTERS, AWESOME, CAMS, COMINF, IS, KSINBIT, MIOS-CIAO, MONET, ONTOCONTENT, ORM, PERSYS, OTM ACADEMY DOCTORAL CONSORTIUM, RDDS, SWWS, AND SEBGIS, 2006, Montpellier, France. **On the Move to Meaningful Internet Systems**: Proceedings. Berlin: Springer, 2006. p.1576–1585. (Lecture Notes in Computer Science, v.4278).

SPACCAPIETRA, S. et al. On Spatial Ontologies. In: BRAZILIAN SYMPOSIUM ON GEOINFORMATICS, 6., 2004, Campos do Jordão, SP, Brazil. **Proceedings...** São Paulo: INPE, 2004.

SPECIA, L.; MOTTA, E. Integrating Folksonomies with the Semantic Web. In: EUROPEAN SEMANTIC WEB CONFERENCE - THE SEMANTIC WEB: RESEARCH AND APPLICATIONS, ESWC, 4., 2007, Innsbruck, Austria. **Proceedings...** Berlin: Springer, 2007. p.624–639. (Lecture Notes in Computer Science, v.4519).

STOILOS, G.; STAMOU, G. B.; KOLLIAS, S. D. A String Metric for Ontology Alignment. In: INTERNATIONAL SEMANTIC WEB CONFERENCE, ISWC, 4., 2005, Galway, Ireland. **Proceedings...** Berlin: Springer, 2005. p.624–637. (Lecture Notes in Computer Science, v.3729).

STOIMENOV, L.; DJORDJEVIC-KAJAN, S. Realization of GIS Semantic Interoperability in Local Community Environment. In: AGILE CONFERENCE ON GISCIENCE, AGILE, 6., 2003, Lyon, France. **Proceedings...** [S.l.: s.n.], 2003.

STOIMENOV, L.; DJORDJEVIC-KAJAN, S. An Architecture for Interoperable GIS use in a Local Community Environment. **Computers and Geosciences**, [S.l.], v.31, p.211–220, 2005.

STOJANOVIC, L.; STOJANOVIC, N.; VOLZ, R. Migrating Data-intensive Web Sites into the Semantic Web. In: ACM SYMPOSIUM ON APPLIED COMPUTING, SAC, 2002, Madrid, Spain. **Proceedings...** [S.l.]: ACM, 2002. p.1100–1107.

SUNNA, W.; CRUZ, I. F. Structure-Based Methods to Enhance Geospatial Ontology Alignment. In: INTERNATIONAL CONFERENCE ON GEOSPATIAL SEMANTICS, GEOS, 2., 2007, Mexico City, Mexico, 2007. **Proceedings...** Berlin: Springer, 2007. p.82–97. (Lecture Notes in Computer Science, v.4853).

TOMAI, E.; KAVOURAS, M. From "Onto-GeoNoesis" to "Onto-Genesis": the design of geographic ontologies. **GeoInformatica**, Boston, v.8, n.3, p.285–302, 2004.

TRINKUNAS, J.; VASILECAS, O. Building Ontologies from Relational Databases Using Reverse Engineering Methods. In: INTERNATIONAL CONFERENCE ON COMPUTER SYSTEMS AND TECHNOLOGIES, COMPSYSTECH, 2007, Rousse, Bulgaria. **Proceedings...** [S.l.: s.n.], 2007.

VISSER, U.; STUCKENSCHMIDT, H.; SCHLIEDER, C. Interoperability in GIS - Enabling Technologies. In: AGILE CONFERENCE ON GEOGRAPHIC INFORMATION SCIENCE, 5., 2002, Palma, Spain. **Proceedings...** [S.l.: s.n.], 2002.

VOLZ, S. Data-Driven Matching of Geospatial Schemas. In: INTERNATIONAL CONFERENCE ON SPATIAL INFORMATION THEORY, COSIT, 2005, Ellicottville, NY, USA. **Proceedings...** Berlin: Springer, 2005. p.115–132. (Lecture Notes in Computer Science, v.3693).

WORBOYS, M. F.; DUCKHAM, M. Integrating Spatio-Thematic Information. In: INTERNATIONAL CONFERENCE ON GEOGRAPHIC INFORMATION SCIENCE, GISCIENCE, 2., 2002, Boulder, CO, USA. **Proceedings...** Berlin: Springer, 2002. p.346–362. (Lecture Notes in Computer Science, v.2478).

XU, W.; HUANG, H.-K.; LIU, X.-H. Spatio-temporal ontology and its application in geographic information system. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING AND CYBERNETICS, 5., 2006. **Proceedings...** [S.l.: s.n.], 2006. p.1487–1492.

ONTOLOGIES

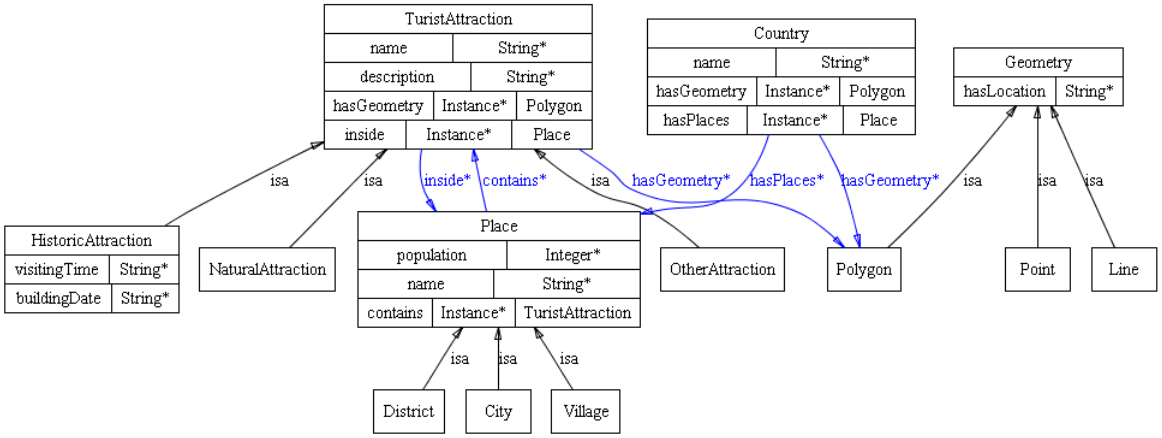

Figure 1: Tourist attraction ontology produced by human expert 1

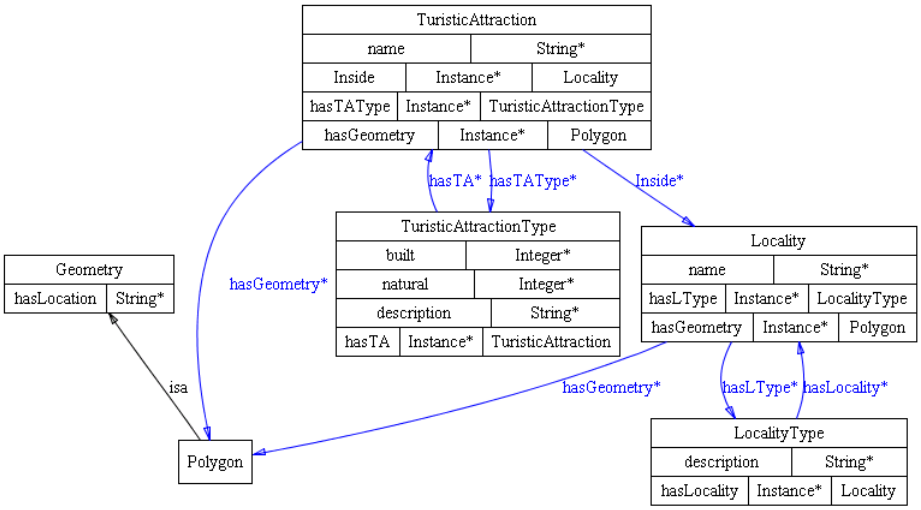

Figure 2: Tourist attraction ontology produced by human expert 2

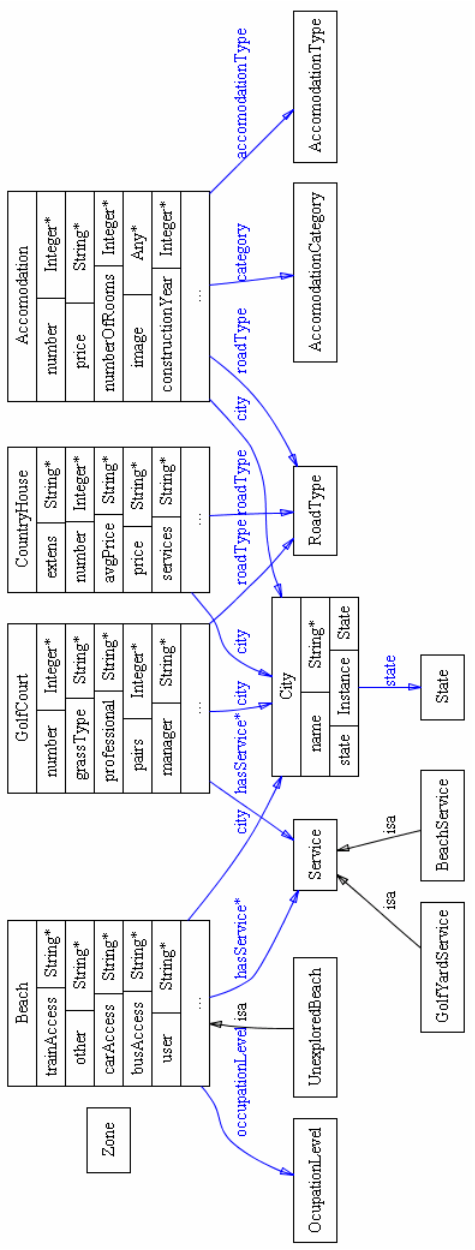

Figure 3: Andalucia-tourism ontology

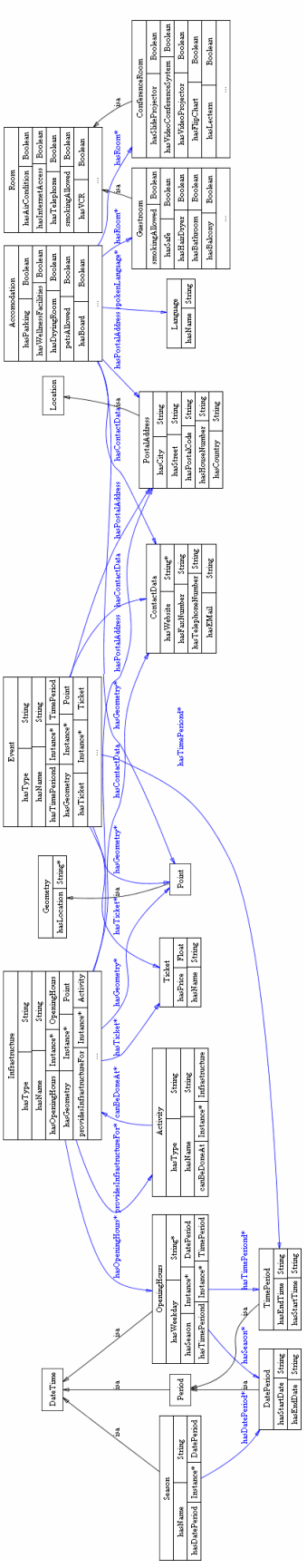

Figure 4: e-tourism ontology

E-MAIL SENT TO GIS EXPERTS

—– Original Message —–
From: <hess@inf.ufrgs.br>
To: <claudioruschel@terra.com.br>; <miguel.fornari@gmail.com>; <luvargas@terra.com.br>; <vargas@inf.ufrgs.br>
Sent: Tuesday, August 12, 2008 3:21 PM Subject: Modelagem geográfica

Luciana, Miguel e Ruschel

Tudo bem?

Estou tentando terminar minha tese de doutorado, na qual proponho um método/técnica para comparar ontologias geográficas. Para testar o algoritmo e as métricas que proponho me faltam casos de teste. Desta forma, gostaria de pedir a ajuda de vocês para modelar uma ontologia (pode quase ser vista como um diagrama de classes ou E-R, mas sem métodos). Assim, abaixo passo a descrição do domínio a ser modelado.

Se cada um de vocês construir a ontologia, terei 3 para comparar, mais uma que eu criei. Isso me dará 6 possibilidades de comparação 2 a 2, que já ajudará muito.

Não pensei em nada muito extenso. A modelagem não precisa ser muito complexa e nem é necessário preocupar-se em demasia se todo o domínio está sendo coberto.

Preferencialmente pediria para modelarem no Protege (protege.stanford.edu), mas se vocês não o conhecem, não percam tempo com ele. Pode ser modelado como um diagrama UML baseado, por exemplo, no GeoFrame. Inicialmente não precisam ser colocados dados. Somente a definição das classes (conceitos), suas propriedades e relacionamentos é suficiente.

Muito obrigado pela ajuda. Peço que me respondam se poderão me auxiliar e para quando acham que poderão ter a modelagem pronta.

Um abraço,

Guillermo

A descrição segue abaixo:
==========
Deve ser criado uma ontologia para representar o domínio geográfico, especialmente para

descrição dos pontos turísticos de uma localidade, a qual pode ser, por exemplo, uma cidade, um povoado, uma centro histórico, etc. Os pontos turísticos podem ser naturais (ex. lagos, colinas, parques, etc.) ou construídos (ex.: museus, monumentos, templos, restaurantes, etc.). Não necessariamente devem estar assim classificados. Além dos pontos turísticos propriamente ditos, todos os demais conceitos (classes) que forem considerados necessários para descrição do domínio podem ser modelados. A classificação é livre. Por se tratar de uma ontologia, devem ser descritos conceitos geográficos e não geográficos envolvidos no domínio. Os conceitos devem estar definidos, na medida do possível, em hierarquias. Para os conceitos modelados, podem/devem ser descritos seus atributos, relacionamentos e relacionamentos espaciais (topológicos e direcionais, se for o caso). Um conceito geográfico deve, obrigatoriamente, estar associado a uma geometria, que pode ser ponto, linha ou polígono. Se for possível, descreva a ontologia em inglês.

RESUMO EXPANDIDO

Desde o surgimento dos Sistemas de Informação Geográfica (SIG), novas áreas de pesquisa vêm surgindo, em função das particularidades dos dados geográficos, os quais são distintos dos dados ditos convencionais (alfanuméricos). Além dos componentes descritivos, tais como generalização/especialização, relacionamentos e atributos, os dados geográficos se caracterizam por três outros componentes. São eles a geometria, localização espacial e possibilidade de representação de relacionamentos espaciais (ARONOFF, 1991; FONSECA; DAVIS; CAMARA, 2003). Dados geográficos podem ainda ter um componente temporal (SOTNYKOVA et al., 2005), embora isto não possa ser apontado como uma especificidade dos dados geográficos. Adicionalmente, dados geográficos são descritos utilizando-se alguns metadados particulares, os quais dão importantes informações adicionais sobre os dados, tais como o sistema de coordenadas, o sistema de projeção, a escala de projeção, a data de aquisição do dados, etc.

Os relacionamentos espaciais são relacionamentos que podem ocorrer entre dois objetos (dados) geográficos, em função de suas geometrias e posições espaciais. Os relacionamentos espaciais estão classificados em três categorias:

- **Relacionamentos topológicos**: São relacionamentos que podem ocorrer entre dois objetos dependendo de suas geometrias. Exemplos deste tipo de relacionamentos são *contem (contains)*, *disjunto (disjoint)*, *dentro (inside)*, *cruza (crosses)*, entre outros. Para geometrias regulares, que são usadas neste trabalho, um dos modelos mais amplamente aceitos é o *9-intersection model* de Max Egenhofer (EGENHOFER; FRANZOSA, 1991).

- **Relacionamentos direcionais**: São relacionamentos que não dependem da geometria dos objetos geográficos associados. Dependem da posição espacial relativa de um objeto em relação ao outro, quando ambos são comparados. Existem doze relacionamentos direcionais (FRANK, 1992), como por exemplo *Ao_Norte*, *Ao_Sudeste*, *Acima* e *A_Esquerda*.

- **Relacionamentos métricos**: São relacionamentos que representam a distância entre dois objetos geográficos. Geralmente não calculados pelas ferramentas de SIG em tempo de execução, não sendo modelados ou armazenados.

Quando informações geográficos são armazenadas, comparadas ou apresentadas, existem alguns metadados que devem ser associados, de modo a permitir uma correta interpretação da informação. Entre eles, destacamos:

- Data de captura e atualização da informação. Se possível, o periodo de validade desta informação geográfica.

- Sistema de coordenadas, sistema de projeção e escala de representação da informação.
- Informação sobre o equipamento e modo de captura da informação.
- Formato de armazenamento: matricial ou vetorial.
- Unidades de medidas utilizadas, tais como milhas ou quilômetros, metros, pés ou jardas, etc.

Atualmente, SIGs são usados diariamente, seja em aplicações especializadas, seja para realizar atividades quotidianas. Exemplos são os sistemas de posicionamento global (GPS) usados em veículos, as ferramentas Google Earth e Google Maps, geradores de mapas na web, etc. Contudo, produzir mapas e/ou dados geográficos em geral consome muito tempo e é custoso financeiramente. Além disso, em muitos casos os dados necessários podem já estar disponibilizados em um outro sistema ou em uma outra organização. Ao mesmo tempo, a popularização da Internet propicia o intercâmbio de informações ao redor do mundo.

Por um lado, esta possibilidade de intercâmbio oferece inúmeros benefícios, tais como o reuso de informações e compartilhamento de conhecimento. Por outro lado, gera uma necessidade de tratar das heterogeneidades entre as informações a serem integradas que foram obtidas de diversas fontes. Este problema é complexo de ser resolvido em virtude da pouca documentação e da semântica que está implícita nos dados, bem como pela diversidade de fontes de dados. A web 2.0, conhecida como a web semântica, tem como objetivo incluir semântica explícita nos dados a serem intercambiados, de modo a que computadores sejam capazes de identificar recursos úteis.

Um campo de pesquisa que surgiu com a web semântica é o das ontologias. Uma ontologia é uma especificação explícita de uma conceituação (GRUBER, 1993). Mais especificamente, uma ontologia é uma teoria lógica que corresponde ao significado intencional de um vocabulário formal, isto é, um comprometimento ontológico com uma conceitualização específica do mundo (GUARINO, 1997). Uma ontologia consiste em axiomas lógicos que contém o significado dos termos para uma comunidade específica. Os axiomas lógicos representam a hierarquia entre os conceitos, bem como os relacionamentos entre eles. Uma ontologia é específica para uma comunidade e deve ser aceita em comum acordo pelos membros desta comunidade. (BISHR et al., 1999).

Uma ontologia é composta por conceitos organizados em uma taxonomia, por propriedades, axiomas e instâncias dos conceitos. Os conceitos descrevem os elementos que devem ser representados, enquanto as propriedades representam as características que podem ser associadas aos conceitos, tais como atributos e relacionamentos. Os axiomas são os relacionamentos do tipo *pai-filho* presentes na taxonomia, ou restrições aplicadas às propriedades quando no contexto de um conceito particular. As instâncias, por sua vez, representam os indivíduos pertencentes aos conceitos.

Ontologias vêm sendo largamente utilizadas para armazenamento e intercâmbio de informações através da Internet, uma vez que podem funcionar de forma muito semelhante aos bancos de dados, mas com semântica explícita associada aos seus elementos, e necessitando de um espaço de armazenamento reduzido. Quando do intercâmbio de ontologias, um dos principais desafios consiste em identificar quais são as estruturas (conceitos ou instâncias) equivalentes e, adicionalmente, mensurar quão similar elas são. Este processo é chamado de medida de similaridade ou *matching*.

No âmbito das ontologias convencionais, já existem boas propostas para técnicas/ferramentas de *matching*, tais como as apresentadas em (CASTANO; FERRARA; MONTANELLI, 2006; GIUNCHIGLIA; SHVAIKO; YATSKEVICH, 2005; NOY, 2004). Estes *matchers* trabalham basicamente no nível de conceitos e não são capazes de considerar as particularidades da informação geográfica. No que diz respeito às soluções existentes para *matchers* geográficos, ainda há uma lacuna em propostas que sejam eficazes e completas. Isto quer dizer que existem técnicas ou ferramentas que consideram algumas das particularidades dos dados geográficos, tanto em nível de conceitos (FONSECA et al., 2002; FONSECA; DAVIS; CAMARA, 2003; STOIMENOV; DJORDJEVIC-KAJAN, 2005; QUIX et al., 2006; KAVOURAS; KOKLA; TOMAI, 2005; VISSER; STUCKENSCHMIDT; SCHLIEDER, 2002; SUNNA; CRUZ, 2007; RODRÍGUEZ; EGENHOFER, 2004) quanto em nível de instâncias (SEHGAL; GETOOR; VIECHNICKI, 2006; HARIHARAN et al., 2005; NAVARRETE; BLAT, 2007). Contudo, nenhuma delas é completa, especialmente considerando o nível de instâncias. As poucas propostas existentes que dizem abordar tanto o nível de conceitos quanto o nível de instâncias são ainda mais limitadas (DOBRE; HAKIMPOUR; DITTRICH, 2003; WORBOYS; DUCKHAM, 2002; MANOAH; BOUCELMA; LASSOUED, 2004).

O cenário apresentado motivou esta pesquisa, na qual é proposta uma solução para o problema de identificação da similaridade, ou equivalência, entre conceitos ou instâncias geográficas. Para tanto foram desenvolvidas técnicas de *matching* tanto para o nível de instâncias quanto para o nível de conceitos, de modo a que possa ser possível medir o grau de similaridade entre os conceitos das ontologias a serem comparadas, bem como entre suas instâncias. As técnicas propostas consistem em algoritmos e expressões matemáticas (métricas), considerando tanto as características convencionais, não geográficas, quanto as características específicas de ontologias geográficas. Algumas das métricas foram adaptadas da literatura existente para *matching* de ontologias não geográficas, enquanto outras foram desenvolvidas especialmente para as características geográficas.

Um dos principais fatores que podem limitar as técnicas existentes que se propõem a medir a similaridade entre ontologias geográficas é, justamente, a falta de um modelo padrão no qual as ontologias devem ser descritas. Cada proposta utiliza um modelo próprio. De modo a superar essa limitação, nesta pesquisa nós propomos e formalizamos um modelo genérico para descrição de ontologias geográficas estáticas, isto é, não temporais. Este modelo foi pensado especificamente para fins de *matching*. Ele é baseado em propostas existentes para ontologias geográficas e também foi concebido de modo a representar todos aqueles elementos que devem ser considerados no processo de medida de similaridade. O modelo para ontologia geográfica proposto é, adicionalmente, compatível com as recomendações do *Open GIS Consortium* (OGC).

Um dos diferenciais do modelo para ontologias geográficas proposto neste trabalho é a possibilidade de associar metadados aos conceitos e, principalmente, instâncias da ontologia. Isto é importante para evitar conclusões errôneas no processo de medida de similaridade ocasionadas por diferenças não nos valores das propriedades associadas aos conceitos e instâncias, mas nos valores dos metadado que os descrevem. Os metadados que podem ser associados à ontologia, atualmente, são o sistema de coordenadas e a escala de representação.

Resumidamente, o modelo para ontologia proposto é uma extensão do modelo OKBC (CHAUDHRI et al., 1998), com criação de alguns conceitos, propriedades e axiomas específicos para representação correta e completa da informação geográfica. O modelo é apresentado formalizado. Com base no modelo também são formalizadas as

heterogeneidades que podem ocorrer quando da comparação de duas ontologias. Estas heterogeneidades são o ponto de partida no desenvolvimento dos algoritmos e expressões matemáticas para a medida de similaridade propriamente dita.

As técnicas para *matching* de conceitos e instâncias geográficas constituem as principais contribuições deste trabalho. No nível de conceitos, é proposto um algoritmo de duas passagens que mede a similaridade entre pares de conceitos de ontologias distintas. Este algoritmo aplica uma série de métricas para avaliar a similaridade dos conceitos de acordo com as várias características que estes apresentam. São consideradas tanto as características que os conceitos geográficos têm em comum com os conceitos não geográficos, tais como atributos, relacionamentos e hierarquias, bem como características particulares da informação geográficas, especificamente os relacionamentos espaciais dos tipos topológicos e direcionais. Desta forma, diferentemente dos *matchers* geográficos existentes, a técnica aqui apresentada não está limitada a comparar conceitos geográficos. Ela também pode comparar conceitos não geográficos, o que é importante, uma vez que podem existir conceitos convencionais em uma ontologia geográfica.

Outro diferencial da técnica proposta nesta pesquisa para o *matching* em nível de conceitos está no fato que a medida final de similaridade é uma soma ponderada das várias características encontradas. Uma vez que, dependendo das ontologias a serem comparadas, elas podem estar mais ou menos detalhadas em um ou outro aspecto (exemplo, muitos atributos e poucos relacionamento espaciais), a possibilidade de ponderar qual métrica terá mais peso e qual terá menos, possibilita que resultados mais eficazes sejam alcançados. Conforme já mencionado anteriormente, algumas das métricas utilizadas foram adaptadas das existentes na literatura, enquanto outras foram criadas especialmente para as características geográficas.

No nível de instâncias, a técnica apresentada possui ainda mais contribuições. As instâncias também são comparadas duas a duas, mas somente se elas pertencerem a conceitos previamente identificados como equivalentes. Contudo, antes da execução do algoritmo de *matching* propriamente dito, é introduzido nesta pesquisa o conceito de região de geográfica de contexto. O objetivo é eliminar da comparação instâncias da ontologia O que estejam geograficamente distantes das instâncias da ontologia $O^{'}$,pois, desta forma, certamente não serão encontrados pares de instâncias equivalentes. A região geográfica de contexto é dada pela operação de *overlay* entre os mínimos retângulos envolventes (MBR - *minimum bounding rectangle*) formados pelos conjuntos de instâncias da duas ontologias em questão.

Ainda antes de executar o algoritmos de medida da similaridade entre instâncias, é necessário deixar homogêneos os valores das propriedades afetadas pelos metadados. Deste modo, os valores das propriedades das instâncias de $O^{'}$ são convertidos de modo a que fiquem representados usando o mesmo conjunto de metadados das instâncias da ontologia O. Atualmente, são considerados os metadados de escala de projeção e sistema de coordenadas. Outros metadados importantes, tais como as unidades de medida e data de aquisição dos dados, devem ser suportados no futuro.

O algoritmo de *matching* de instâncias propriamente dito é executado em três etapas, sempre considerando a comparação de instâncias duas a duas. Primeiramente, é verificada a distância entre as duas instâncias. Se esta distância for maior que um determinado limiar, o par é eliminado. No momento, todas as instâncias geográficas são convertidas para a geometria de ponto para a comparação das coordenadas geográficas, através de sua centróide. Nada impede, contudo, que em uma futura implementação seja usado o MBR de cada geometria para o teste da similaridade posicional.

Para os pares de instâncias que passaram pelo teste de similaridade posicional, a segunda etapa consiste em verificar a similaridade dos identificadores das instâncias. Esta medida é usada na terceira etapa, que consiste em medir a similaridade para os valores das propriedades, ou seja, atributos, relacionamentos convencionais e relacionamentos espaciais. O valor final da similaridade entre instâncias é uma soma ponderada entre a similaridade de identificador e a similaridade dos valores das propriedades. Pares de instâncias que tiverem similaridade medida abaixo de um determinado limiar são automaticamente descartados.

Apesar da informação geográfica se caracterizar pela geometria, pela posição espacial e pelos relacionamentos espaciais, muitas vezes uma ontologia geográfica não descreve estas características explicitamente. Desta forma, nesta pesquisa também é apresentada uma técnica para o enriquecimento de ontologias geográficas via engenharia reversa. O objetivo principal é acrescentar relacionamentos topológicos entre os conceitos e granularidade à taxonomia da ontologia pela análise dos valores das propriedades associadas às instâncias, mas também é possível re-definir toda a estrutura da ontologia. Adicionalmente, a ontologia fica sendo compatível com os padrões do OGC.

A técnica de enriquecimento semântico da ontologia via engenharia reversa é, em parte, inspirada nos trabalhos de engenharia reversa para página web baseadas em dados (*data intensive web pages*) (STOJANOVIC; STOJANOVIC; VOLZ, 2002; ASTROVA, 2004). A partir das instâncias da ontologia descrita em OWL (*Web Ontology Language*) os conceitos são reconstruídos com suas propriedades. Adicionalmente, pelo valor das coordenadas geográficas é possível inferir os relacionamentos topológicos que ocorrem entre duas instâncias e generalizá-los para os conceitos por elas instanciados, vinculando estes relacionamentos às geometrias dos conceitos. Por fim, a taxonomia da ontologia pode ser enriquecida de duas formas: via comparação com uma ontologia de referências ou, pela comparação das propriedades dos conceitos.

Os testes executados demonstraram que os algoritmos propostos, bem como as métricas utilizadas, são indicadas para o *matching* de ontologias geográficas, pois obtiveram melhores resultados que quando a medida de similaridade foi efetuada utilizando-se *matchers* para ontologias convencionais. Não foi possível comparar os resultados com *matchers* geográficos, pois não estavam disponíveis protótipos do mesmos, nem sequer os algoritmos e expressões matemáticas utilizadas.

As técnicas apresentadas nesta pesquisa foram implementadas em uma arquitetura de software chamada IG-MATCH . Utilizou-se a linguagem Java com a API do *Protege* para acesso à ontologias em OWL, a API *simmetrics*, a API *JWNL* para acesso a WordNet e a API *sdoapi* para acesso ao Oracle Spatial 10g XE.

GLOSSARY

A	The axioms of an ontology.
$AH(c, c^{'})$	Attribute heterogeneity between the concepts c and $c^{'}$.
C	The concepts of an ontology.
c	A concept.
$CH(c, c^{'})$	Concept heterogeneity between the concepts c and $c^{'}$.
$ctx(c)$	The context of a concept.
$eqTop(top_A, top_B)$	Test of equivalence between the topological relationships top_A and top_B.
gc	A geographic concept, which specializes a concept.
$DH(c, c^{'})$	Directional relationship heterogeneity between the concepts c and $c^{'}$.
ge	A geometric property, labeled `hasGeometry`, which relates a geographic concept gc and a geometry concept geo.
geo	A specialization of a concept, representing a geometry.
$GH(c, c^{'})$	Geometric heterogeneity between the concepts c and $c^{'}$.
giim	Geographic information integration or mapping.
$HH(c, c^{'})$	Hierarchy heterogeneity between the concepts c and $c^{'}$.
I	The set of instances of an ontology.
i	An instance.
$ICH(i, i^{'})$	Instance coordinate heterogeneity between instances i and $i^{'}$.
$IIH(i, i^{'})$	Instance identifier heterogeneity between instances i and $i^{'}$.
$IH(i, i^{'})$	Instance heterogeneity between instances i and $i^{'}$.
$IRH(i, i^{'})$	Instance relationship heterogeneity between instances i and $i^{'}$.
M	The metadata of an ontology.
$maxCard(p)$	The maximum cardinality of a property p.

$mdv(i)$ A metadata value associated to the instance i.

$minCard(p)$ The minimum cardinality of a property p.

$NH(c, c^{'})$ Name heterogeneity between the concepts c and $c^{'}$.

O An ontology.

P The set of properties of an ontology.

p A property.

$p(c)$ A property in the context of the concept c.

pd The domain of a property.

pos The spatial position of a geographic instance i.

$pv(i)$ A property value associated to the instance i.

R A geographic context region.

$RH(c, c^{'})$ Relationship heterogeneity between the concepts c and $c^{'}$.

$Sim(c, c^{'})$ The overall similarity measure between concepts c and $c^{'}$.

$SimHier(c, c^{'})$ Hierarchy similarity measure between concepts c and $c^{'}$.

$SimAt(c, c^{'})$ Attribute similarity measure between concepts c and $c^{'}$.

$SimIAtN(i, i^{'})$ Instance numeric attribute similarity measure between instances i and $i^{'}$.

$SimIAtS(i, i^{'})$ Instance text attribute similarity measure between instances i and $i^{'}$.

$SimIDir(i, i^{'})$ Instance directional relationship similarity measure between instances i and $i^{'}$.

$SimIId(i, i^{'})$ Instance identifier similarity measure between instances i and $i^{'}$.

$SimIR(i, i^{'})$ Instance relationship similarity measure between instances i and $i^{'}$.

$SimInst(i, i^{'})$ Overall instance similarity measure between instances i and $i^{'}$.

$SimITop(i, i^{'})$ Instance topological relationship similarity measure between instances i and $i^{'}$.

$SimName(c, c^{'})$ Name similarity measure between concepts c and $c^{'}$.

$SimPos(i, i^{'})$ Instance spatial location similarity measure between instances i and $i^{'}$.

$SimRel(c, c^{'})$ Conventional relationship similarity measure between concepts c and $c^{'}$.

$SimSpt(c, c^{'})$ Spatial relationship similarity measure between concepts c and $c^{'}$.

$t(c)$ The label of a concept.

$TH(c, c^{'})$ Topological relationship heterogeneity between the concepts c and $c^{'}$.

$x(c)$ An axiom in the context of the concept c.

WA Weight for the attribute similarity, in the concept matching algorithm.

WH Weight for the hierarchy similarity, in the concept matching algorithm.

WN Weight for the name similarity, in the concept matching algorithm.

WR Weight for the conventional relationship similarity, in the concept matching algorithm.

WS Weight for the spatial relationship similarity, in the concept matching algorithm.