Building Geospatial Ontologies from Geographic Database Schemas in Peer Data Management Systems

Danúbia Lima[1], Antonio Mendonça[1], Ana Carolina Salgado[2], Damires Souza[1]

[1] Federal Institute of Education, Science and Technology of Paraiba, Brazil

[2] Federal University of Pernambuco, Brazil

{danubialima@gmail.com, tony2415@gmail.com, acs@cin.ufpe.br, damires@ifpb.edu.br}

Abstract. *One key issue in Peer Data Management Systems (PDMSs) is the heterogeneity of the peer schemas. To help matters, ontologies may be used as uniform conceptual representation of these schemas. In this work, we are working with geographic databases to be used in a PDMS. When dealing with geospatial data, specific problems with representation and usage occur. In this sense, we have developed an approach and a tool, named GeoMap, which builds a peer ontology from a geographic database schema. In order to provide geospatial semantics when mapping, we have defined and used a reference geospatial ontology. We present the principles underlying our approach and examples illustrating how they work by means of the tool.*

1. Introduction

Peer Data Management Systems (PDMSs) came into the focus of research as a natural extension to distributed databases in the peer-to-peer (P2P) setting [Lodi *et al*. 2008, Sung *et al*. 2005]. PDMSs are considered the result of blending the benefits of P2P networks, such as lack of a centralized control, with the richer semantics of a database [Zhao 2006]. They can be used for data exchanging, query answering and information sharing. For instance, in the areas of scientific research, the idea of setting up a PDMS to share research data among peers has already been widely discussed [Lodi *et al*. 2008, Zhao 2006].

A PDMS consists of a set of inter-related peers (data sources). Each peer has an associated schema within a domain of interest. However, PDMSs do not consider a single global schema. Instead, each peer represents an autonomous data source and exports either its entire data schema or a portion of it. Such schema, named exported schema, represents the data to be shared with the other peers of the system.

Data management in PDMSs is a challenging problem given the heterogeneity of their schemas. Due to the fact that ontologies provide good support for understanding the meaning of data, they have been used as an uniform metadata representation, i.e., each data source schema is represented by a local ontology (named *peer ontology*) [Souza *et al*. 2011, Xiao 2006]. In addition, due to semantic heterogeneity, research on PDMSs has also considered the use of ontologies as a way of providing a domain reference [Souza *et al*. 2011, Xiao 2006]. Considering a given knowledge domain, an agreement on its terminology can occur through the definition of a domain ontology which can be used as a semantic reference or background knowledge to enhance processes such as ontology matching and query answering.

One of the most representative realms of diversity of data representation is the geospatial domain. Geospatial data, besides hierarchical and descriptive components (relationships and attributes), are featured by other ones such as geometry, geospatial location and capability of holding spatial relationships (e.g., topological) [Hess 2008, Fonseca *et al.* 2003]. Furthermore, geospatial data are often described according to multiple perceptions, different terms and with different levels of detail. Syntactical aspects have been addressed by interoperability standards, such as Geography Markup Language [GML 2007]. However, the most hard-facing problem is still concerned with semantic heterogeneity.

In this work, we are working with geographic databases to be used in a PDMS called SPEED (Semantic PEEr-to-Peer Data Management System) [Pires 2009]. In order to uniformly deal with geospatial data without worrying about their specific heterogeneity restrictions (syntactic or semantic), we use ontologies as uniform conceptual representation of peer schemas. When a peer asks to enter the system, its schema (e.g., represented according to the relational or object-relational database model) must be automatically exported to a *peer ontology*. Due to the special semantics of geospatial data, this automatic extraction becomes more complex. Thus, in this work, we present an approach and an implemented tool, named *GeoMap*, for automatically building a geospatial peer ontology as a semantic view of data stored in a geographic database. During the ontology building process, a set of correspondences (relationships) between the generated ontology components and the original database schema is also automatically generated. The produced peer ontology will be later used for matching and querying processes in the PDMS. The set of correspondences will be used to translate ontological queries into the database query language (e.g., SQL) and retrieve corresponding instances from the geographic databases.

This paper is organized as follows: Section 2 introduces the SPEED system; Section 3 presents the *GeoMap* approach; Section 4 describes the developed *GeoMap tool* with some peer ontology generation examples. Related work is discussed in Section 5. Finally, Section 6 draws our conclusions and points out some future work.

2. The SPEED System

The SPEED system is a PDMS which works with three distinct types of peers, namely: data peers, integration peers, and semantic peers. A data peer represents a data source sharing structured or semi-structured data with other data peers in the system. Data peers are grouped within semantic clusters according to their semantic interest. A semantic interest includes the peer's interest theme and a local peer ontology. The interest theme is an abstract description of the peer's semantic domain, whereas the local peer ontology (LO) describes the peer's exported schema. Each semantic cluster has a special type of peer named integration peer. Actually, integration peers are data peers with higher availability, network bandwidth, processing power, and storage capacity. Such peers are responsible for tasks like managing data peers' metadata, query answering, and data integration. An integration peer maintains a cluster ontology (CLO), which is obtained through the merging of the local ontologies representing data peers' and integration peer's exported schemas. Integration peers communicate with a semantic peer, which is responsible for storing and offering a community ontology (CMO) containing elements of a particular knowledge domain (i.e., a domain ontology). Semantic peers are responsible for managing integration peers' metadata. A set of clusters sharing semantically similar interests composes a semantic community. An overview of the SPEED architecture, with its kinds of peers and used ontologies is presented in Figure 1.

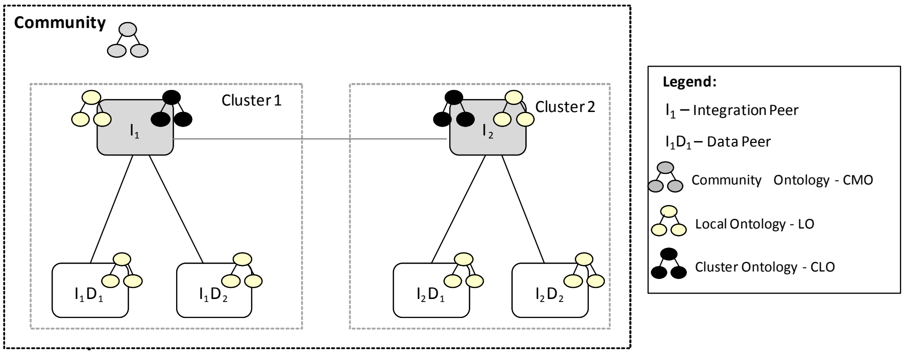

Figure 1: An Overview of SPEED Architecture

3. The *GeoMap* Approach

The database-to-ontology mapping approaches are usually classified into two main categories [Ghawi and Cullot 2009]: (i) approaches which create an ontology from a database and, (ii) approaches which map a database to an existing ontology. In the former, the objective is the creation of an ontology from a database and may include both the metadata and the data. The mappings, in this case, are the correspondences between each created ontology component (e.g., concept, property) and its original database schema concept (e.g., table, column). In the latter, the goal is to create a set of mappings between the existing ontology and the database. In our approach, we focus on the former, i.e., we build an ontology from a geographic database. Particularly, we build an ontology from the geospatial metadata. Nevertheless, the generated ontology does not contain data, i.e., the data remains in the database.

In our work, geospatial data are represented by means of the vector model. As a result, they are expressed as objects and are stored as points, lines or polygons, depending on the scale of their capture. Thus, the heterogeneity of data sources (databases) is even greater than in conventional databases: data may have multiple representations (the same data can be represented as a point in a given data source or as a polygon in another one), data may have different resolutions and different coordinate systems as well as temporal properties associated. In addition, since existing spatial database systems do not follow the same spatial data model (for instance, PostGIS [PostGis 2011] is based on the OGC specification, although Oracle is not [Oracle 2010]), there are differences when dealing with metadata from most of them. In this sense, the syntactic, semantic and spatial data format heterogeneity should be considered when creating an ontology from a geographic database.

On the other hand, an ontology is composed by concepts, properties (defined by means of domain and range information), axioms and, optionally, instances. Since an ontology is a knowledge representation technique based on Description Logics (DL) [Baader *et al*. 2003], it is usually coded using OWL (Web Ontology Language) model [Horrocks 2005]. As a result, there is a gap in terms of concept and relationship definitions between the ontology model and the database schema model. Regarding geospatial data, there has been a lot of research looking for spatial ontology definitions [Hess *et al.* 2007, Arpinar *et al.* 2004] as well as for extensions to SPARQL language in the geospatial realm [Zhai *et al.* 2010]. Nevertheless, recently, there has been published an OGC candidate document which aims to specify a geographic query language for RDF data named GeoSPARQL [GeoSPARQL 2011]. The OGC GeoSPARQL standard will define a vocabulary for representing geospatial data in RDF as well as an extension to the SPARQL query language for processing geospatial data. We have defined specific constructs in OWL to deal with geospatial concepts and relationships, considering OWL as an extensible XML-format.

In the following, we introduce our approach by means of its main architecture. Then, we present a reference ontology which has been used to guide the mapping and the steps underlying our generation ontology process.

3.1 Architecture

Our approach, named *GeoMap*, is based on the architecture depicted in Figure 2. From a geographic database, a *peer ontology* (i.e., an application ontology) is built by means of the *GeoMap* components: at first, the database schema is extracted, then its elements are classified into spatial and non spatial ones, then its respective geospatial OWL construct is identified and, finally, the peer ontology is generated. This ontology represents, through ontological concepts and properties, the structure of the database. In order to provide semantics when accomplishing the OWL construct identification, we use a geospatial domain ontology (a reference ontology). During the generation process, an OWL document is also automatically produced to record the set of correspondences (relationships) between the generated ontology components and the original database metadata. This document will later be used to translate ontological queries into the database query language and retrieve corresponding instances.

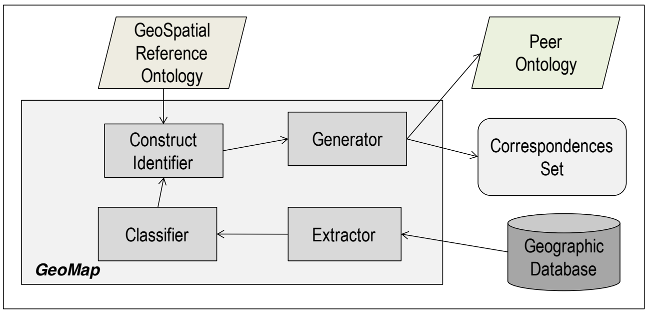

Figura 2. *GeoMap* Architecture

3.2 Reference Ontology

Using OWL as a common representation model, the SPEED system minimizes the problems of heterogeneity by transforming the schemas of data sources using ontologies. Nevertheless, regarding geospatial metadata, the set of predefined constructs in OWL does not include specifications for the description of geographical concepts and properties. Therefore, it was necessary to use some kind of background knowledge which could be a reference at the mapping process, when generating such geospatial constructs (tags). Although existing geospatial ontologies are available, we could not find out one which completely fitted our purposes. Thus, we have defined a geospatial domain ontology to be used as a reference in our process.

The reference ontology has been developed using the Protégé 3.4.4 tool [Protégé 2011]. An excerpt from this ontology is depicted in Figure 3, using the OntoViz notation [OntoViz 2011]. In this format, subtypes are associated with their supertypes through the *isa* relationship. Relationships between concepts are also presented through edges.

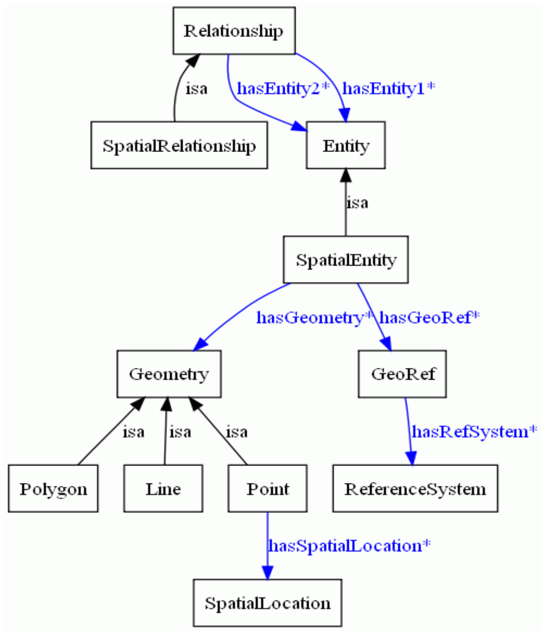

Figure 3. Excerpt from the Geospatial Reference Ontology

The reference ontology should be able to represent an abstraction of geospatial and non geospatial metadata, including, for example, entities, spatial entities, relationships, spatial relationships and geometry. In order to represent such concepts, we have defined specific high level concepts: Entity as the main concept; SpatialEntity as a specialization of Entity representing a geographic phenomenon; Relationship as a general kind of association between entities and SpatialRelationship as a specialization of Relationship regarding specific geospatial ones. A SpatialEntity has a Geometry (point, line or polygon) and is associated with a geospatial reference (a Reference System). Points are represented through spatial location, lines are defined through points and polygons are defined through lines.

3.3 Mapping Process

The mapping process used in *GeoMap* approach is based on the particular aspects described in the previous sections. An overview of its steps is presented in Figure 4. During this process, rules are applied to transform the geospatial schema elements from the database into geospatial ontology components. These rules include: (i) after connecting to the database, it extracts the database schema; (ii) it identifies, among the obtained tables, which are non-spatial, i.e., tables that have no geographic columns; (iii) meanwhile, it also identifies the spatial ones. Among the obtained tables, (iv) it identifies simple type properties (e.g., varchar and number types) and (v) it identifies relationships which are mapped into object type properties, with the specification of domain and range, (vi) it also classifies these relationships as spatial or nonspatial ones. After processing all these steps, the tool produces an OWL file (i.e., the peer ontology) representing the geographic database schema.

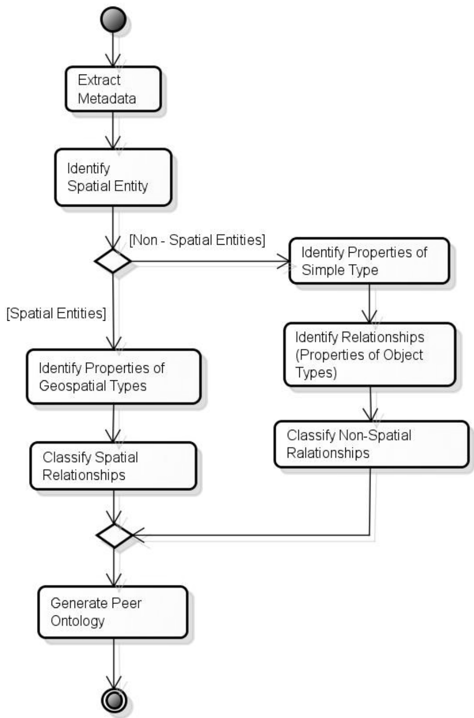

Figure 4 . Peer Ontology Generation Process

In next section, we provide some implementation issues for *GeoMap* and present the tool underlying our approach through an example.

4. The *GeoMap* Tool: Experiments and Results

The *GeoMap* tool has been implemented in JAVA as an extension of an object-relational database ontology generation tool [Franco 2009]. In this current version, *GeoMap* uses geographic databases coded in Oracle DBMS [Oracle 2010]. The Protégé-OWL API [Protégé 2011] and the Jena framework [Jena 2011] have been used for ontology manipulation.

In Figure 5, we present a use case diagram which shows the functional requirements that have been considered in the *GeoMap* tool implementation. There are two actors in the diagram. The first one is the *GeoMap* tool itself that starts the whole process of ontology mapping by connecting to the database. The database, in turn, characterizes the second actor. It is worth mentioning that, in this current version, the mapping options are initiated by a "user", i.e., manually by a *GeoMap* user. In a future version, the tool will be set in SPEED system as a service to be called whenever a peer (with geographic database) requests entering the system.

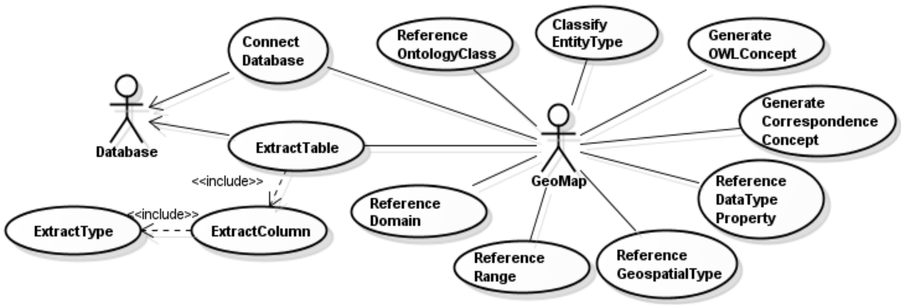

Figure 5. Use Case Diagram for the *GeoMap* Tool

After connecting to the database, *GeoMap* retrieves the existing geospatial types and tables from the database schema. Through this recovery, it is possible to extract all the columns, identifying what is a simple attribute and what represents a geometry. Then, the tool identifies the geometry type of geospatial tables - whether it is point, line or polygon. Based on the recovery of all entities and their properties (simple or objects), the tool makes use of the domain ontology as a reference of terms and creates the specific geospatial tags (i.e., constructs). Thus, when creating the spatial entities representation and referring the types of geometry, we use the reference domain ontology. When referencing the domain, we identify the domain of properties. When referencing the geographical range for a column, we specify the geometry types (line, polygon or point) present in the reference ontology.

During the mapping of these metadata, it is also possible to identify the equivalence correspondences of the generated ontology components and the existing database schema entities and properties. In order to define this set of equivalence correspondences, we build an OWL document composed by a specific construct named *IsEquivalentTo.* Such construct has been defined and used to indicate which ontology concept is equivalent to the database schema element. Also, it indicates which ontology properties are equivalent to the database schema attributes and relationships. An excerpt from a produced set of correspondences is depicted in Figure 6. In this example, we are working with a database regarding districts in São Paulo city. In the owl file, for instance, the SPATIAL_DATA column in the database schema corresponds to the DITRITOSSP_SPATIAL_DATA concept in the peer ontology, i.e., they are equivalent.

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:j.0="d:\base2.owl#" >
  <rdf:Description rdf:about="http://localhost:1521#DENOMINACAO">
    <j.0:isEquivalentTo>d:\base2.owl#DISTRITOSSP_DENOMINACAO</j.0:isEquivalentTo>
  </rdf:Description>
  <rdf:Description rdf:about="http://localhost:1521#BAIRRO">
    <j.0:isEquivalentTo>d:\base2.owl#BAIRROSSP_BAIRRO</j.0:isEquivalentTo>
  </rdf:Description>
  <rdf:Description rdf:about="http://localhost:1521#DISTR">
    <j.0:isEquivalentTo>d:\base2.owl#BAIRROSSP_DISTR</j.0:isEquivalentTo>
  </rdf:Description>
  <rdf:Description rdf:about="http://localhost:1521#CLASSE">
    <j.0:isEquivalentTo>d:\base2.owl#DRENAGEMSP_CLASSE</j.0:isEquivalentTo>
  </rdf:Description>
  <rdf:Description rdf:about="http://localhost:1521#BAIRROSSP">
    <j.0:isEquivalentTo>d:\base2.owl#BAIRROSSP</j.0:isEquivalentTo>
  </rdf:Description>
  <rdf:Description rdf:about="http://localhost:1521#SPATIAL_DATA">
    <j.0:isEquivalentTo>d:\base2.owl#DISTRITOSSP_SPATIAL_DATA</j.0:isEquivalentTo>
  </rdf:Description>
  <rdf:Description rdf:about="http://localhost:1521#SIGLA">
    <j.0:isEquivalentTo>d:\base2.owl#DISTRITOSSP_SIGLA</j.0:isEquivalentTo>
  </rdf:Description>
```

Figure 6. Excerpt from an Obtained Set of Correspondences

At end, the *GeoMap* tool produces two outputs: (i) the peer geospatial ontology and (ii) the owl document with the set of equivalence correspondences. As a way to present *GeoMap* tool main steps execution, we provide some examples in the following.

4.1 *GeoMap* in Practice

Figure 7 shows a screenshot of the tool's main window that is split into four parts: (i) area which identifies the database name, (ii) area showing the structure obtained from the database schema, (iii) area with the *generated peer ontology* and (iv) area with the set of produced correspondences. In this example, we use a geographic database that stores attributes and geometries from the neighborhoods, districts and drainage map of São Paulo city. For instance, the database table named "DistritoSP" is represented as a polygon through the Oracle type MDSYS.SDO_GEOMETRY. Its structure is presented in an expanded view also shown in Figure 7.

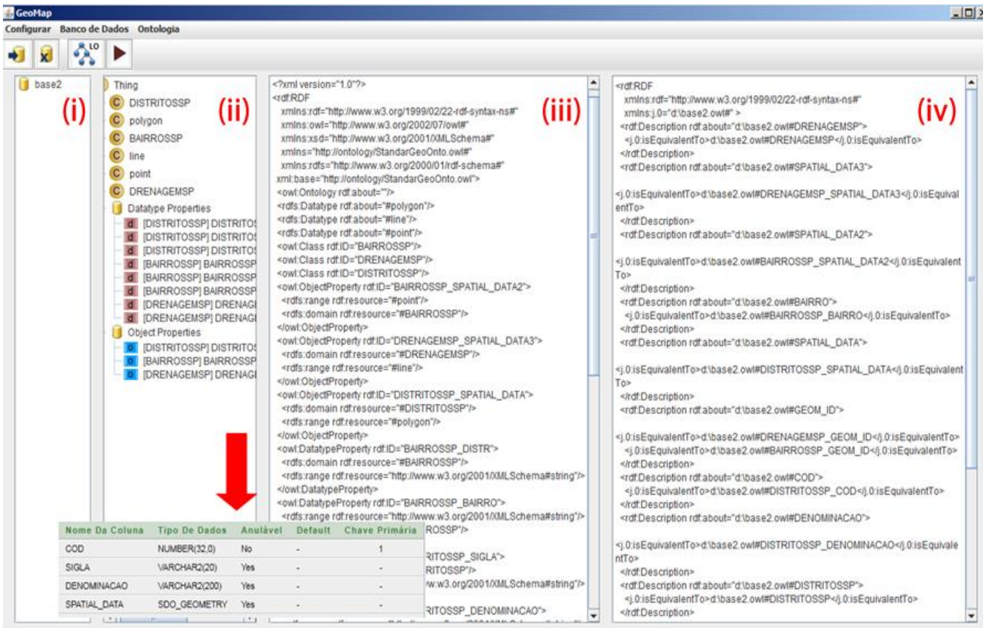

Figure 7. *Geomap* Interface and an Example of Database Schema

Particularly, another example regards using a geographic database that stores *laboratories* from IFPB Institute. According to the process explained in Section 3, the tool identifies spatial and non-spatial entities and properties from the database schema and generates the peer ontology and the set of equivalence correspondences. For the sake of visibility, we present the database schemas for the tables *Laboratórios*, *Corredor* and *TV* apart from the interface (Figure 8a). In addition, we depict the generated peer ontology from such database and the set of obtained correspondences in Figure 8b.

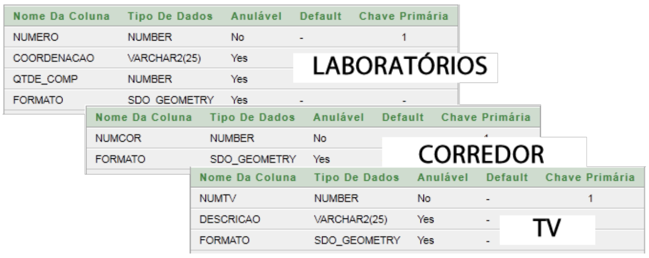

Figure 8a. Laboratories Database Schema

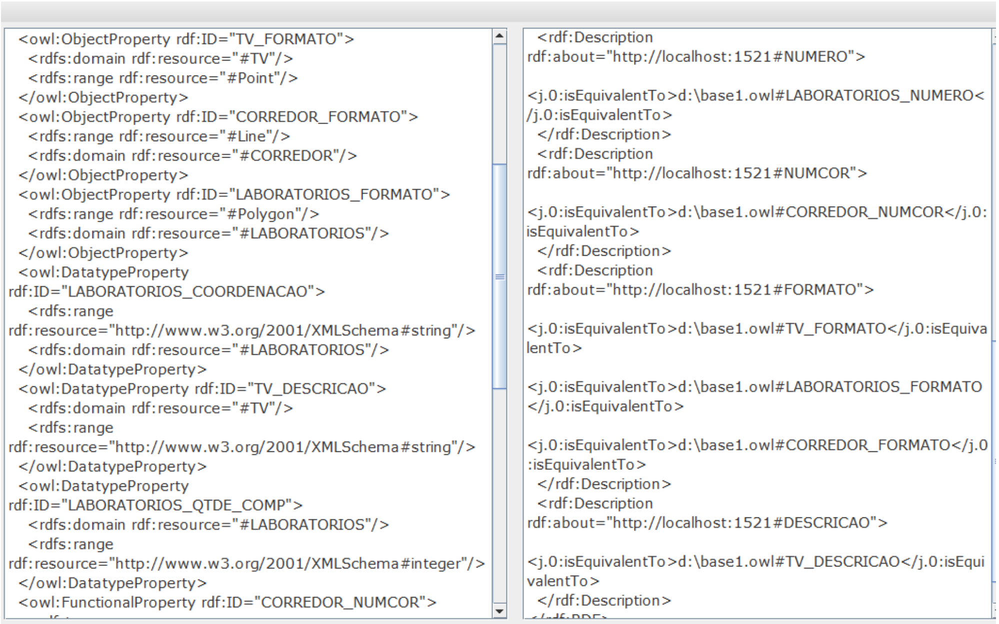

Figure 8b. Produced Peer Ontology and Correspondences

We have accomplished some initial experiments with the *GeoMap* tool. The goal of our experiments is to check if we can assess *completeness* in terms of the produced peer ontology. According to some quality information criteria [Batista and Salgado 2007, Wang and Strong 1996], regarding PDMS systems, we have defined *completeness* as the degree to which entities and properties of the peer data source (i.e. the database schema) are not missing in the generated peer ontology. In order to measure such criterion, we have invited some users (knowledgeable about the Geospatial domain and OWL/RDF constructs) to produce a manual peer ontology from the geographic database schemas. These "gold ontologies" were compared with our produced peer ontologies. As result, we could verify that our produced peer ontologies are quite complete (ninety percent on average) in terms of the existing database elements, i.e., they include most of all the schema elements from the database. The different components obtained from the "gold" ontologies and ours regarded semantic interpretations when defining the geometry types: the expert users defined point, line and polygon as owl:class while our tool is still generating them as rdf:datatype. In fact, it indicates a probable mistake that will be corrected. Furthermore, we intend to accomplish additional experiments with other experts and other databases in order to obtain a more concrete result.

5. Related Work

Currently, there are many approaches and tools which build ontologies from databases [Franco 2009, Cerbah 2008, Cullot *et al.* 2007, Baglioni *et al.* 2007]. However, most of them are concerned with relational databases. As an example, the DB2OWL tool map relational databases to OWL ontologies, considering particular table cases during the mapping process [Cullot *et al.* 2007]. Another example regarding relational databases is the RDBToOnto tool [Cerbah 2008]. This tool produces an ontology and allows refining its generated version. To this end, RDBToOnto provides a visual interface for accomplishing manual changes. Lubyte and Tessaris define a framework for extracting from a relational database an ontology that is to be used as a conceptual view over the data [Lubyte and Tessaris 2007]. In this work, the semantic mapping between the database schema and the ontology is captured by associating a view over the source data to each element of the ontology (i.e., by means of a GAV approach) [Halevy 2001]. Regarding object-relational databases, the work of Franco [2009] implements a tool which provides the ontology built from such kind of database.

Particularly, in the geospatial realm, Cruz *et al.* [10] developed a semi-automatic method to generate mappings between ontologies of local databases and a global one. The generated mappings are then used for query rewriting. In a closer scope, Baglioni *et al.* [2007] defined a method to access spatial database through an ontology layer. To this end, they developed a semi-automatic tool which builds an application ontology from a geographical database. They also enrich the generated ontology with semantics from a domain ontology by finding correspondences between the classes and properties of the two ontologies. This work is the most similar to ours.

Comparing these works with ours, we are able to produce the peer ontology in an automatic way, by using the semantics provided by the reference ontology. In this light, we can use any background knowledge that may support the geospatial semantics we need. Thus, for instance, we will be able to use the OGC GeoSparql standard (when it is ready for use) as our domain reference. Furthermore, in our approach, we do not need the user intervention, since this current tool will be stated as a service in SPEED system which will be dynamically executed at peer arriving time.

6. Conclusions and Future Work

This work has presented the *GeoMap* approach and tool. *GeoMap* accomplishes the extraction of metadata from a geographic database representing them in terms of a peer ontology. Also, it identifies the equivalence correspondences between the generated ontology components and the existing database schema entities and properties. To this end, it takes into account the identification and classification of geospatial and non spatial entities and properties, figuring out how they can be represented in terms of an OWL ontology. A geospatial reference ontology is used as a way to provide the semantics of geospatial relationships and types, absent from the set of existing concepts in the OWL model.

Currently, this version generates ontologies from Oracle databases. As future work, it will be extended to also extract metadata from other DBMSs, such as PostGIS and other data sources such as GML. Another important future work concerns identifying correspondences between geospatial peer ontologies. Thereby, we will be able to reformulate and execute geospatial queries among the existing peers in the PDMS.

References

Arpinar I. B., Sheth A., Ramakrishnan C., Usery E. L., Azami M. and Kwan M. "Geospatial Ontology Development and Semantic Analytics". In: Handbook of Geographic Information Science. Blackwell Publishing (2004).

Baader F., Horrocks I., and Sattler U. (2007) "Description Logics". In Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors, Handbook of Knowledge Representation. Elsevier, 2007.

Baglioni M., Masserotti M., Renso C., Spinsanti L. (2007) "Building Geospatial Ontologies from Geographical Databases". GeoS 2007: 195-209.

Batista M. C., Salgado A.C. (2007) "Data Integration Schema Analysis: An Approach with Information Quality". In Proceedings of the 12th International Conference on Information Quality (ICIQ), MIT, Massachusetts, USA, October 2007.

Cerbah, F. (2008) "Learning Highly Structured Semantic Repositories from Relational Databases – The RDBtoOnto Tool". In: 5th European Semantic Web Conference (ESWC'08), pages 777-781, Tenerife, Spain (2008).

Cruz, I. F., Sunna, W., Chaudhry, A. (2004) "Semi-automatic ontology alignment for geospatial data integration". In: Egenhofer, M.J., Freksa, C., Miller, H.J. (eds.) GIScience 2004. LNCS, vol. 3234, pp. 51–66. Springer, Heidelberg (2004).

Cullot N., Ghawi R., and Yétongnon, K. (2007) "DB2OWL: A Tool for Automatic Database-to-Ontology Mapping". In: 15th Italian Symposium on Advanced Database Systems (SEBD'07), pages 491-494, Torre Canne di Fasano (BR), Italy (2007).

Fonseca, F. T., Davis Jr. C. A., Camara G. (2003) "Bridging Ontologies and Conceptual Schemas in Geographic Applications Development". *Geoinformatica*, v. 7, p. 355-378, 2003.

Franco, F. (2009) "Uma Ferramenta de Transformação de Esquemas Objeto-Relacionais para Ontologias". Trabalho de Graduação, Centro de Informática, Universidade Federal de Pernambuco (UFPE), Recife, PE, Brasil.

Geography Markup Language (2007). Available at: http://www.opengeospatial.org/standards/gml. Last access on august 15th, 2011.

GeoSPARQL documentation (2011). Available at: http://www.opengeospatial.org/projects/groups/geosparqlswg. Last access on august 15th, 2011.

Ghawi, R., and Cullot, N. "Building Ontologies from Multiple Information Sources". In 15th Conference on Information and Software Technologies (IT2009) (Kaunas, Lithuania, April 2009).

Halevy A. (2001) "Answering queries using views: A survey". VLDB Journal, 10(4):270{294, 2001.

Hess G. (2008) "Towards effective geographic ontology semantic similarity assessment". PhD Thesis. UFRGS.

Hess G., Iochpe C., Castano S. "Towards a Geographic Ontology Reference Model for Matching Purposes". GeoInfo 2007: 35-47.

Horrocks I. (2005) "Applications of description logics: State of the art and research challenges". Proc. of the 13th Int. Conf. on Conceptual Structures (ICCS'05) April 2005.

Jena (2011). Jena - A Semantic Web Framework for Java. http://jena.sourceforge.net/

Lodi, S., Penzo, W., Mandreoli, F., Martoglia, R., and Sassatelli, S. "Semantic Peer, Here are the Neighbors You Want!" In: 11th Extending Database Technology (EDBT'08), Nantes, France, pp. 26-37 (2008).

Lubyte L., Tessaris S. (2007) "Extracting ontologies from relational databases". In Proc. of the 20th Int. Workshop on Description Logics (DL'07).

OntoViz (2011). Available at: http://protegewiki.stanford.edu/wiki/OntoViz. Last access on august 18th, 2011.

OpenGIS (2011). Available at: http://www.opengeospatial.org/standards. Last access on august 15th, 2011.

Oracle 10g documentation (2010). Available at: http://www.oracle.com/technetwork/indexes/documentation/index.html. Last access on august 15th, 2011.

OWL API (2011) – available at: http://protege.stanford.edu/plugins/owl/api/. Last access on august 15th, 2011.

Pires C.E.S. (2009) "Ontology-Based Clustering in a Peer Data Management System". PhD thesis, Center for Informatics, UFPE, 2009.

PostGIS documentation (2011). Available at: http://postgis.refractions.net/documentation/. Last access on august 15th, 2011.

Souza D., Pires C. E., Kedad Z., Tedesco P. C., Salgado A. C. (2011) "A Semantic-based Approach for Data Management in a P2P System". To be published in LNCS Transactions on Large-Scale Data- and Knowledge-Centered Systems, 2011.

Sung L. G. A., Ahmed N., Blanco R., Li H., Soliman M. A., Hadaller D. (2005) "A Survey of Data Management in Peer-to-Peer Systems". School of Computer Science, University of Waterloo.

Wang R. Y., Strong D. M. (1996) "Beyond Accuracy: What Data Quality Means to Data Consumers". Journal of Management Information Systems, Vol. 12, N. 4, pages 5-33, 1996.

Xiao H. (2006) "Query processing for heterogeneous data integration using ontologies". PhD Thesis in Computer Science. University of Illinois at Chicago, 2006.

Zhai X., Huang L., and Xiao, Z. (2010) "Geo-spatial query based on extended SPARQL". In Proceedings of Geoinformatics. 2010, 1-4.

Zhao J. (2006) "Schema Mediation and Query Processing in Peer Data Management Systems". Master Thesis, University Of British Columbia, October, 2006.