

Querying a Geographic Database using an Ontology-Based Methodology

Renata Viegas¹, Valéria Soares^{1,2}

¹ Programa de Pós-Graduação em Sistemas e Informação – DIMAp – UFRN
Campus Universitário - Natal, RN, Brasil

² Departamento de Informática – UFPB
Campus Universitário – João Pessoa – PB

renata@ppgsc.ufrn.br, valeria@di.ufpb.br

***Abstract.** This work presents a mechanism that allows different community users access the same geographic database without knowing its particular internal structure. We use geographic ontologies to support a common and shared understanding of a specific domain: the coral reefs. Using these ontologies' descriptions that represent the knowledge of the different communities, mechanisms are created to handle with such different concepts. We use equivalent classes mapping, and a semantic layer that interacts with the ontologies and the geographic database, and that gives to the user the answers about his/her queries, independently of the used terms.*

1. Introduction

Geographic Information Systems (GIS) are multidisciplinary systems that could be used by different community users, each one with their own objectives and interests. So, different visions of the same reality must be combined to support the community's necessities.

Different people recognize differently the same geographic region. Geographic features are collected and stored in GIS that were modeled by some specific conceptual model. If we need an efficient search on geographic databases, in some cases, it is necessary to associate meaning with the data. So, current GIS must be able to solve the semantic interoperability due to the fact that a geographic feature could have more than one description. The term semantic refers to the meaning of these features.

According to Fonseca [Fonseca 2001], information systems must be able to understand the user's models and their meanings. In other words, we could say that it is necessary to develop systems that exceed the information barriers to give to users not only the data, but also their meaning.

For spatial data interchange happens without missing information it is necessary a high degree of interoperability between GIS's [Fonseca 2001]. Regarding to interoperability between different GIS formats, some alternatives have been proposed to prevent this problem, as the creation of standards like SDTS (Spatial Data Transfer Standard) [Sondheim et al. 1999] and SAIF (Spatial Archive and Interchange Format) [Surveys 1994].

Although standards for exchange of data are necessary and useful, they do not have the capacity to transfer the meaning associated with these data. Nowadays there is an increasing interest about how to reach interoperability with the use of ontologies as being a knowledge database type. The ontologies could specify a specific vocabulary domain relative, and could define entities, classes, properties, predicates, functions, and the relationships between these components.

The term ontology comes from Philosophy, meaning the representation of existence through a systematic explanation, as being the conception of all things that may “exist” or “be”. In the computer science area, ontology began to be used by Artificial Intelligence, as being a “formal and explicit specification that tries, in the best way, to make the defined world structure to be closer by a concept” [Guarino 1998]. On this definition, “formal” means computer readable; “explicit” is concerning concepts, properties, relations, functions that are explicit defined; “concept” concerns an abstract model of some phenomenon of the real world [Guarino 1998].

An explicit formalization of our mental model is generally called ontology (with a lower-case “o”). The basic description of the real things of the world, the description of what would be true, is called Ontology (with an upper case “O”). Thus, there is only one Ontology, but several ontologies [Fonseca 2001]. Each community that offers information and accesses them has his own ontology. Each one of these ontologies may be divided on small ontologies. The details level of ontologies tells the geographic information details level [Fonseca 2001].

A geographic ontology is a conceptualization of a phenomenon or geographic object in the real world. It is necessary to store all characteristics referred to a geographic object. This is what differs from a geographic ontology and other types of ontologies. Besides the domain being geographic, characteristics of geographic objects (location, topology, direction) are embodied to this ontology. The search for geographic data semantics is important for the interoperability among GIS.

The development of ontology is an iterative process. To build ontology consists on learning and understanding the concepts and visions that are relevant for the different users of a GIS.

This aim work is to provide a mechanism to allow that different communities’ users access the same Geographic Database without knowing its internal structures, and using only specific terms of each research area. Although we have only one database implementation, through the definition of different communities’ ontologies, anyone could search the database, in a transparent way, using a specific interface. No one will need to know how the database is, in fact, implemented. But, everyone could search for information, and will have their queries attended. Through the relationships between the defined ontologies, different descriptions and names about the same data could be merged and implemented with appropriated mechanisms. These mechanisms use equivalent classes mapping, and an intelligent GIS layer that interact with the ontologies and with the geographic database, and that give to the user the answers about his queries, independent of the used terms.

The presented article is organized as follows: section 2 shows a gather of the related works. Section 3 explains the system architecture of our work, Section 4 present

the application in a specific geographic domain, and finally Section 5 concludes this paper.

2. Related Works

Researches about interoperability in information systems are motivated by the increasing heterogeneity in computer world.

Guarino [Guarino 1998] proposed the use of ontologies on information systems, conducting to ontology-driven information systems – ODIS.

Heterogeneous data on GIS is not an exception, but the complexity and diversity of geographic data and the difficulty of its representation make this search for interoperability on this kind of system more complex.

Fonseca [Fonseca 2001] defends the use of geographic ontologies for the semantic integration of data in GIS. On his work, he defines ODGIS (Ontology-Driven Geographic Information Systems), which are systems that use translated ontologies on software components.

Egenhofer report on [Egenhofer et al. 2000] the creation of the Semantic Geospatial Web: a framework for geospatial information retrieval based on the semantics of spatial and terminological ontologies. This framework enable users to retrieve more precisely the data they need, based on the semantics associated with these data.

Others ontology-based works for query formulation could be found in [Dongilli et. al 2004], [Franconi and Tessalis 2005]. In [Cardoso et. al 2005] it is presented an architecture that binds the web semantic concepts with regular expression techniques whose objective is to recover and mine data from web pages.

An algorithm is proposed in [Noy and Musen 1999] to align and merge ontologies, by name similarities between classes. In this algorithm, for each class of all the ontologies, the SMART algorithm automatically executes a series of actions, asking the users if classes with the same name could be aligned in a same ontology.

In this work, we propose to create a semantic layer between the geographic database and the users, using ontologies. With a unique database, we could have a low cost implementation because we have only one data collect, and different users, with different knowledge, could access this same database. Furthermore, we develop a case study on environmental and marine modeling, what increases the difficulty of collecting the information, and none of the proposal quoted on literature is similar to this.

3. System Architecture

We propose a solution based on geographic ontologies to provide that different professionals of distinct research areas access the same Geographic Database (GDB).

From the different terms stored on these ontologies, the system could infer necessary information to the users' queries, allowing that different community users access and interact with the system, without know specific characteristics about the internal structure of the database.

We create a semantic layer that intermediate the users' queries with the geographic database. Each one of the community users could interact with the system using only specific terms of his research area, and could receive his queries answers in an appropriate way. It is transparent to the user how this database was effectively implemented. The user only has to worry about his necessity, and what he wants to look for on the database. Through the semantic layer, the ontologies will be activated, and these users' queries will be translated to an appropriate SQL clause.

We could see the complete system architecture on Figure 1 that shows the semantic layer, between the users, the application and the database.

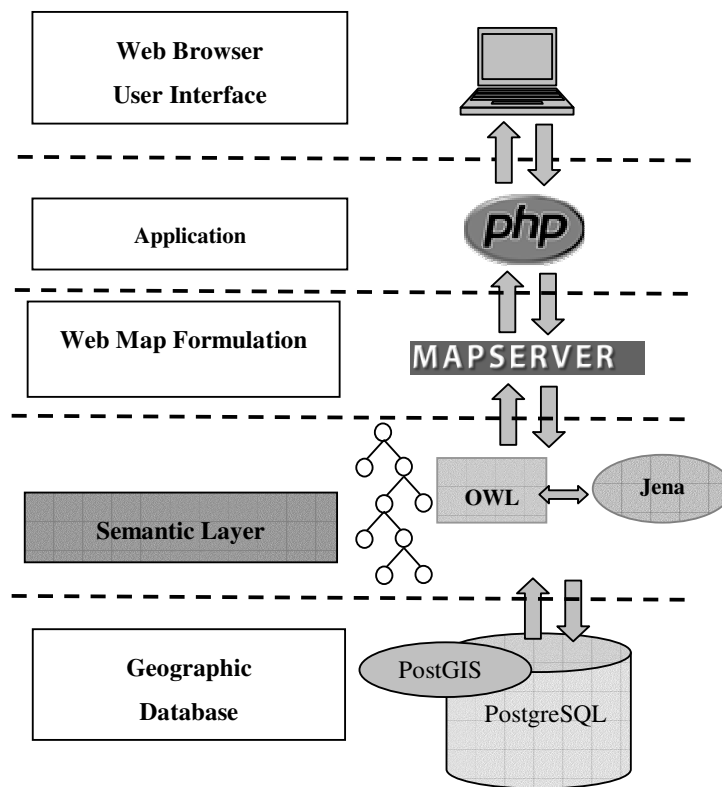


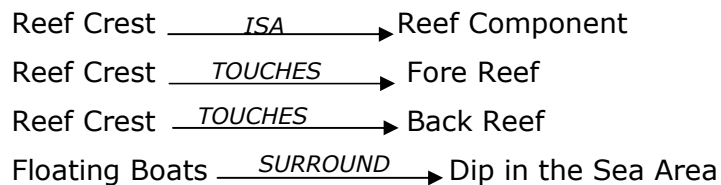
Figure 1. System Architecture with the Semantic Layer

In this work, like we can see in the Figure 1, we developed the geographic database with the DBMS PostGIS [Postgis 2006], developed the ontologies on the language OWL [OWL 2004], with the editor Protégé [Protégé 2005], and use the API Jena [Jena 2004] to generate the ontologies graphs. The package jena.ontology [Jena 2004] has the classes *OntClass* and *OntModel* that allows that graphs components could be accessed by many ways.

3.1 Geographic Ontologies

There are many definitions about ontologies in the literature [Guarino 1998] [Fonseca 2001] [Breitman and Casanova 2005]. Some of these references provide definitions for what a geographic ontology is. Sometimes a geographic ontology is defined as simply an ontology whose domain is geographic. In this work we propose to add some more characteristics to these ontologies to differentiate them to others. For this, we decided to use typical geographic relationships to relate the ontologies classes with each others, like topological, metrics and directions relationships.

We defined relationships between each ontology classes, using the *properties* function of the ontology specification on Protégé Editor [Protégé 2005]. These properties are used to spatial relate a concept to another, based with the basic operations defined by the 9-Intersection Model [Egenhofer and Herring 1991], and also specified in the OpenGIS Consortium [OGC 1999]. This can be verified in the following examples, where we present some concepts that were spatially related with another, such as:



This spatial information in the ontologies' relationships and properties could be useful in the solution of some users' queries, as well as in the complement of others. Through pop-up windows, for each submitted query, the application also supplies a dictionary that gives detailed information (descriptive and spatial) about the instances of the GDB, that were effectively used in the solution of the queries, as well as its relationships. The objective to use these spatial relationships is to give support to users about their submitted queries, giving them additional information about the geographic features used in the solution of his queries.

Each user' community has a pay-define query interface. This specific interface was created to allow the user interact with the system using only the specific knowledge to its area of performance.

We can see another vision of the system architecture in Figure 2. The query will be processed on this following way: the user query will be submitted, and the system will recognize the used terms, and relate with the specific ontology of this type of user at this moment. So, we have to identify the user query based on the concepts and terms that had been used in this submission.

After we had identified the query, the next step is to look for the terms and concepts used in this query, in the users' ontology, comparing this ontology with the GDB ontology, looking for equivalents concepts. Through the ontologies' URLs, the Jena API [Jena 2004] will be used to construct the graphs of the ontologies.

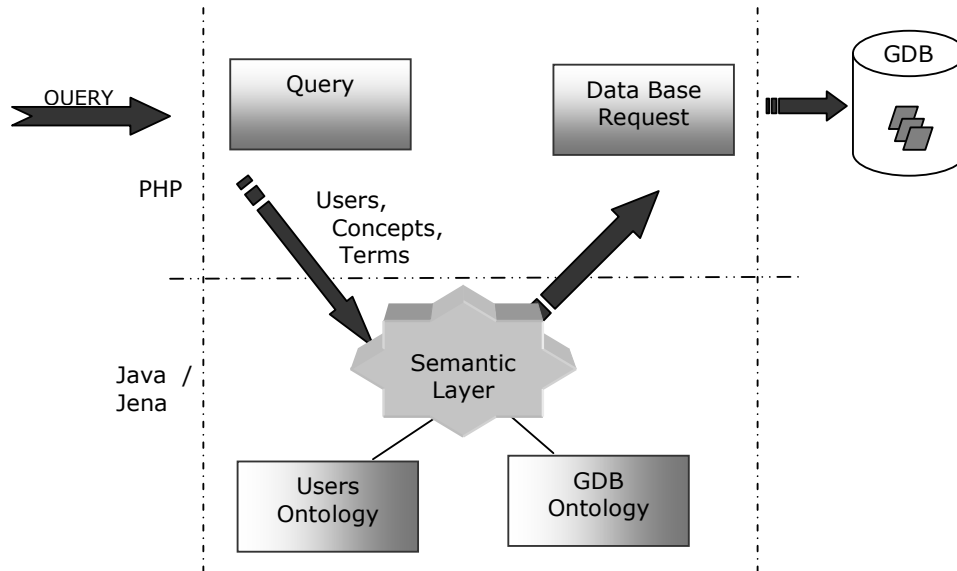


Figure 2. General system architecture

3.2 Detailing the Semantic Layer

The next step, as well as define the ontologies, is the binding definition between the classes of the different ontologies. The result of this binding process is a formal structure with expressions that show which terms of determinate ontology is related to others terms of another ontology.

We have done a mapping with synonyms classes, to help the localization of classes whose information is relevant to answer the user queries. To compare and join the ontologies, in this work we use not only classes that have the same name in distinct ontologies, but also classes with different names, too. We consider synonym classes those whose concepts have the same meaning, independent of their given names, that is related of the specific knowledge of each community.

After we have done this mapping on the defined ontologies, we have also to manipulate them. So, it is necessary to go along the terms of each ontology looking for similar concepts. We use the Jena API [Jena 2004] to generate the ontologies graphs. It is possible to generate graphs RDF, which is represented by resources, properties and literals. From the Jena API methods we can manipulate and compare the ontologies.

A detailing of the semantic layer could be seen in Figure 3. The layers' modules are:

a) Users' Management Module:

The first step to submit a query is to inform which type of user wants to interact with the system. With this module, the user will be able to choose which type of user interface he wants to interact with the system. Depending on the user' choice, the system will shown a pay-define queries' interface, with only specific terms of this type of users, based on the defined ontologies.

b) Ontologies' Management Module:

With the choice of the user's type and the pay-defined query's interface, the next module of the semantic layer will be activated, and will be responsible for the ontologies' activation. In our application, the activated ontologies will always be from the active user, as well as the ontology that represents the contents of the GDB. The ontologies are stored in ontologies' server, and are accessed through its URLs.

At the moment of the activation of the specific ontologies, some methods of the API Jena will be used to construct the graphs (models) of the ontologies.

The API Jena has object classes that represent graphs, resources, properties and literals. A graph is called model, and is represented by the Model interface. These models will keep the activated ontologies for the application. With the ontologies represented in graphs' form, we can make sweepings, looking for the desired terms.

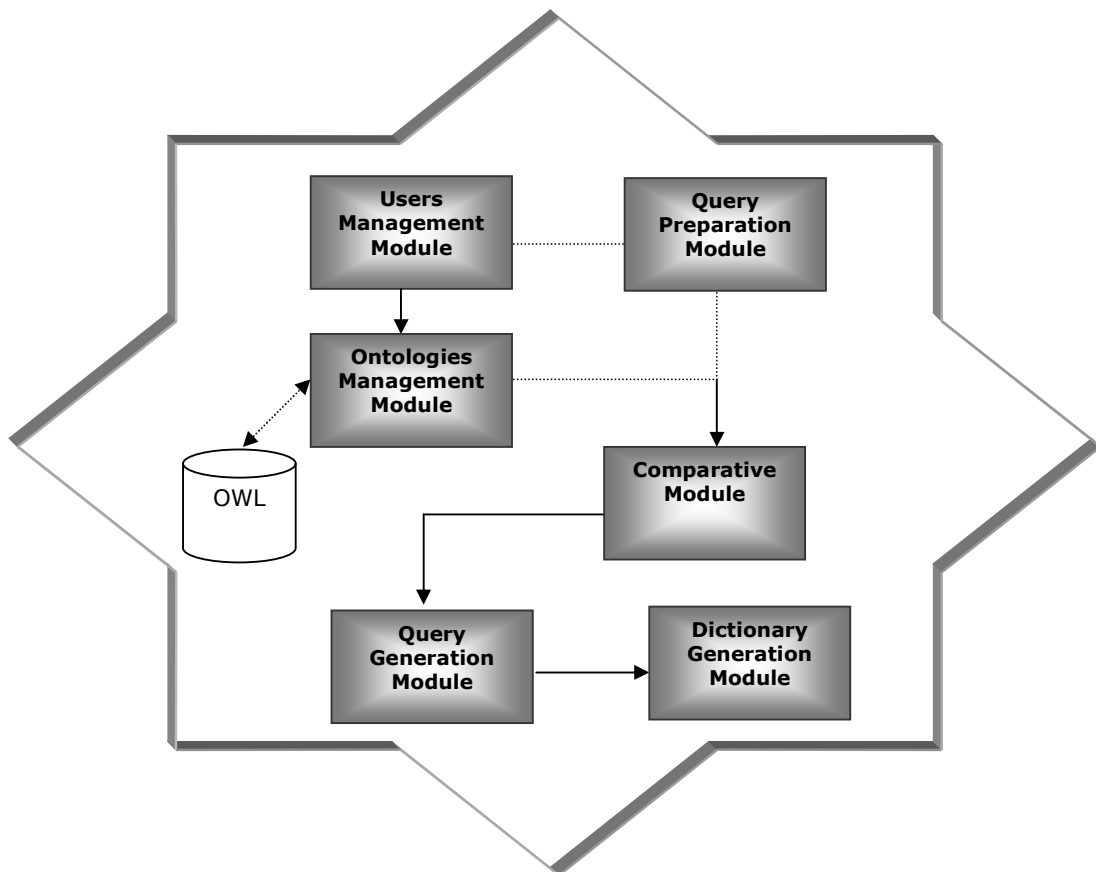


Figure 3. Semantic Layer Modules

c) Query Preparation Module:

With the user-defined interface, the user will formulate his query, choosing what he desires to search. After that, the Query Preparation Module will be activated. This module will identify the key terms of the query.

d) Comparative Module

The ontologies' construction is an iterative process. To accomplish this it is necessary to realize some interviews with professionals of the domain area to collect important terms and the relationships between them, and construct and model the ontologies.

With the ontologies created and modeled, the next step of our methodology is to continue discovering similar terms in the ontologies. In this work, the ontologies will be compared, searching similar terms. This similarity is defined manually, based on the interviews with professionals of the area. In accordance with the point of view of each professional, classes with the same meaning can be nominated with different terms.

One of the most important modules of our application is the Comparative Module. It is responsible by the search of similar terms in the ontologies. With the graphs (models) of the available ontologies, the search will be for classes that have similar concepts to the user's term.

The linking between the equivalent concepts is represented in OWL language through tags *equivalentClass* and *sameAs*, however has a difference between these tags. The `<owl: equivalentClass>` tag is used to indicate that two classes are equivalents if, and only if, they possess, necessarily, the same instances. On the other hand, the `<owl: sameAs>` tag is used when we have different nomenclatures that are mentioned to a same class. A typical use of *owl:sameAs* is for ontologies' unification, to say that two individuals classes, defined in different documents, are equals [OWL 2004].

Thus, we use the `<owl: sameAs>` tag. Manually, each class of each ontology is compared with the classes of another, using the related terms that have the same meaning, but different nomenclatures. For each term found with this characteristic, the `<owl: sameAs>` tag is applied.

The next step to the Comparative Module is to use methods of the API Jena to treat the similarities, binding classes of an ontology to another one. The `getSameAs()` method, by the `OntResource` interface, is used to find the similar classes in the ontologies.

This method looks for the class passed as query parameter, in the user ontology, and search to the terms in the database ontology that have some *owl:sameAs* tag, and that the similarity is accurately the term that passed as parameter. This method also looks for in the ontologies OWL documents to the tag `<owl: sameAs>` that are present in the activated ontologies for the application. If this tag is found, the conceptual similarity is established.

e) Query Generation Module

The Query Generation Module will mount the query that will be submitted to the database. For this, will be used the terms found in the search for the similarity, as well

as the relationship used in the query interface. In next section we give an example of how the system generates the query clause.

f) Dictionary Generation Module

The query will be returned for the user, locating in the map where he searched, beyond an explain window, showing all the information on its research. The Dictionary Generation Module will mount a detailed text, with the key terms of the query, supplying to the user descriptions about the geographic features involved in his queries.

4. Application: Coral Reef Domain

The geographic domain of our application is the coral reefs. We choose this domain because this work is part of a research project of the UFRN that involves researchers of distinct areas on the region of Maracajau reefs situated on the north littoral of the RN state.

In this project we have basically three different communities: the geologists, the biologists and the tourists. Because of that, we developed three different ontologies for each one of these communities.

The main objective of this work primarily was to model the mental worlds of each one of these communities by using ontologies. We choose a natural environment of coral reefs because there were not any works in literature about that use semantic terms with geographic databases.

Besides, we think that all these different communities must have the right to access the information stored on this database. So, we developed ontologies that will give support to the construction of adaptable interfaces for each community involved in the Project. This was done because we know that, although there is only one reality of coral reefs in the region of Maracajaú, each one of these communities think differently about that.

According to Fonseca [Fonseca 2001], if we have a body of water, for example, for a biologist it could be a habitat for some fish, but for a firefighter it could be an emergency source of water. So, it depends on the point of view of each person, and his necessity at this time.

The first ontology that we have developed was the geologist one. This is presented on Figure 4, and was named by O_{geo} . We have to say that this ontology is much bigger than we present here, but we had to summarize to put in this paper.

On the ontology showed in Figure 4, we could see the benthic region. This region could be defined as the deepest layer of a body of water, like a lagoon, a river or the ocean. We could simply say that the benthic region is the minerals and organisms that compound the bottom of reefs.

We developed also ontology for the biologist's community and named it by O_{bio} , and ontology for the tourists community, that we call O_{tur} . Through the semantic layer that we present in the last section, we defined some equivalent classes between these ontologies.

We also have developed different interfaces for each one of these communities. So, if a biologist wants to interact with the system, he will submit a query using only specific biologist terms, without worrying about the database contents.

Next section we could show how users can access the geographic database, submitting their queries through the appropriate interfaces, and which kind of results it is generated.

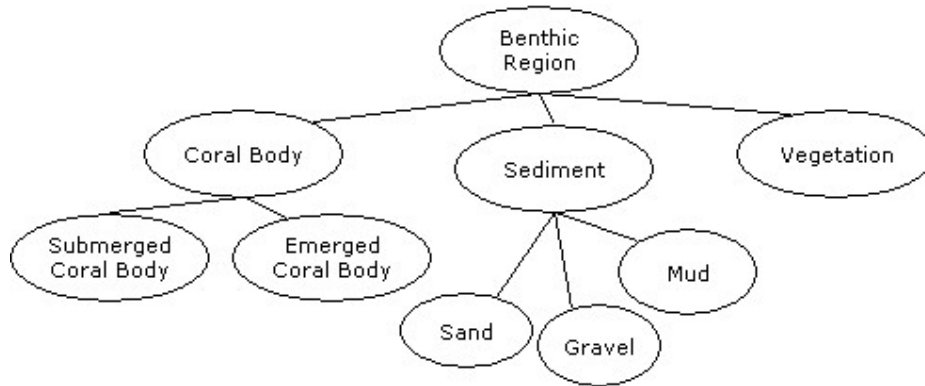


Figure 4. Geographic Ontology for the Coral Reefs Domain

4.1 Prototype Query Examples

We have already developed all the architecture modules presented on section 3, and, based on the coral reef application ontologies we present now two different examples of query submissions.

a) A Tourist Query Submission

Suppose that a tourist wants to find the best area for dip in the sea nearby the coral reefs. This information is not stored on the database. On the other hand, we could find this information using the developed semantic layer that defined equivalences between the classes of the specific ontologies.

The activated ontologies for this query are the geologist (O_{geo}) and the tourist (O_{tur}). So, the first step to solve this user query is to look for the term *dip in the sea* on the tourist ontology. We defined in the O_{tur} that a tourist could dip in the sea around the floating boats on within the natural pools. The floating boats are some boats whose geographic position do not change, and is used as a base point for tourists and researchers that work in the area. After this term was found on the tourist ontology, the system will look for an equivalent term on the geologist ontology.

No similar class to “dip in the sea” could be found in the O_{geo} at this point. So, we have to go down one more step on the geologist ontology, to verify if there is some more information in the relationships between classes or subclasses.

On the other hand, going along the O_{tur} ontology, we found the relations “Dip in the sea in Natural pools” and “Dip in the sea near by floating boats”. That is, the term *dip in the sea* appears in the tourist ontology with properties that bind this class to others

classes in the ontology. The property “*within*” bind the class “dip in the sea” (domain) to the class “natural pool” (range). And the property “*surround*”, binds the class “dip in the sea” (domain) to the class “floating boats” (range).

Then, the system will search now for similar classes to “natural pool” and “floating boats” on the geologist ontology. The class “floating boats” is defined both on the tourist ontology O_{tur} , and on the geologist ontology O_{geo} .

The problem is the term “natural pool” found in the O_{tur} that we still have to search for conceptual similarity on the O_{geo} .

No conceptual similarity is found in the O_{geo} , and thus the *Comparative Module* goes down one more step in the graph generated from the tourist ontology, looking for relationships between classes. Finally at this point, the relationship is found: “Natural pools” surround “Coral Reef”;

“Coral Reef” is situated inside of “Coral Region” on the Geologist Ontology. So, we can infer that for each coral reef body that appears in the coral region, we could find natural pools around it. The information about “Coral Reef Bodies” is stored in the Geographic Database.

So, the query that will be submitted to the geographic database, after we had inferred between the two ontologies is: “show all the areas that surround the floating boats and the coral reefs bodies.” And then, the generated query clause is:

```
SELECT buffer(flutuante.flutuante_geom, 10) AS
  flutuante_geom,
  buffer(corpo_coralineo.geom_cabeco, 10)
  AS geom_cabeco
FROM flutuante, corpo_coralineo
AS foo USING UNIQUE oid USING SRID = -1
```

The operator `buffer` returns the area situated surrounded some geometry, based on a radius value that is passed as a parameter. In this case, we use a radius value equal to 10 meters. The result of this query is shown on Figure 5.

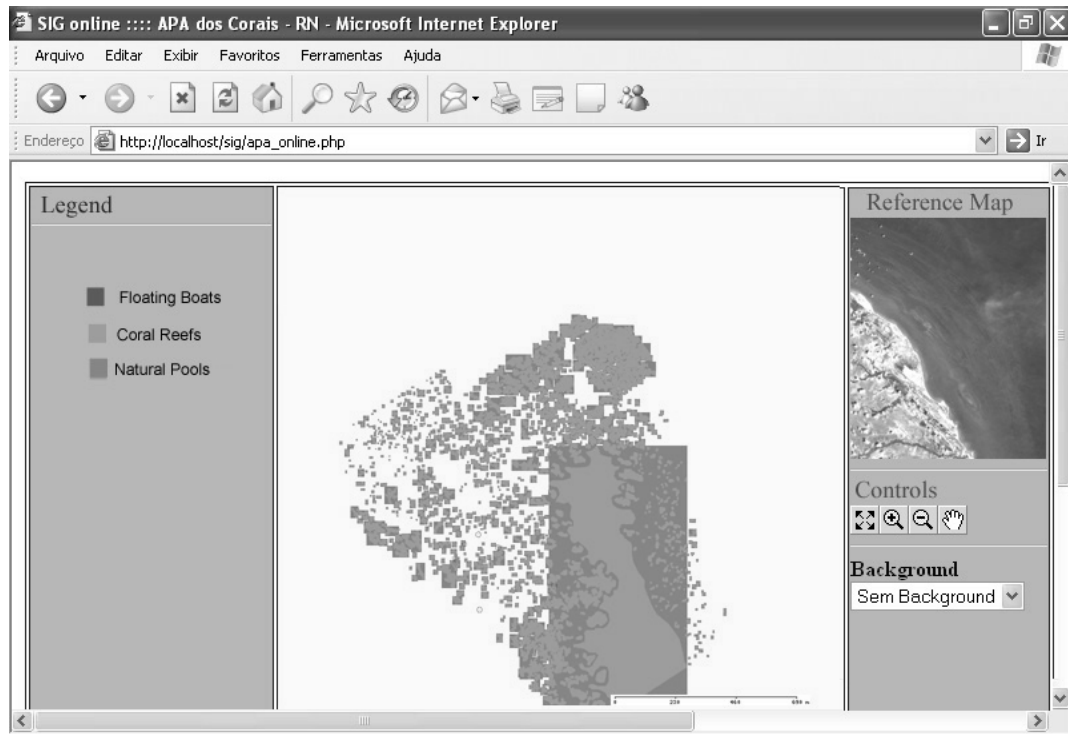


Figure 5. Result set of the submitted query

Finally, using the information stored on the ontologies, and used to generate the queries, the system can show detailed information about the geographic features that were used on the queries, like we can see on Figure 6.

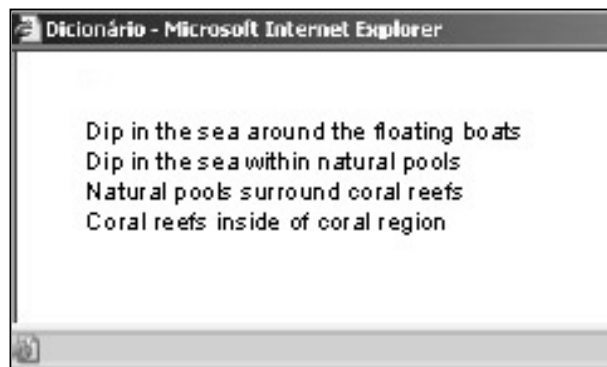


Figure 6. Pop-Up window that shows detailed information about geographic features used in the Tourist Query.

b) A Biologist Query Submission

In this scenario we suppose that a biologist wants to know where he can find no consolidate substrates on the coral reef region. This information is not stored on the database like this.

In this example, the activate ontologies are O_{geo} and O_{bio} . So, we need to start to look for the “no-consolidate substrate” class in the O_{bio} ontology. As soon as this class is found on the biologist ontology, the system has look for class similarities on the definition of this class. The class “no-consolidate substrate” in the biologist ontology has no similar class in any other ontology. In other words, we can say that the “no-consolidate substrate” class is not related with any other class using the owl tag *sameAs*. So, due to the fact that the conceptual similarity was not found in the first level of the biologist graph, the comparative module will go down one more level, looking for if the “no-consolidate substrate” class has some relationships with others classes or if it has sub-classes.

The comparative module finally finds that the “no-consolidate substrate” class has sub-classes (relationship ISA) with the classes: “sand”, “gravel” and “mud”. So, we can say, for example, that sand is a no-consolidate substrate.

Now we have to search about similar classes with “sand”, “gravel” and “mud” on the geologist ontology. All these classes are defined on the geologist ontology too. So, with the conceptual similarity found, the next step is to generate the query clause to be submitted to the geographic database. The Query Generator Module will construct this following SQL query clause:

```
SELECT geom_areia AS geom_areia, geom_lama AS geom_lama,  
geom_cascalho AS geom_cascalho  
FROM areia, cascalho, lama  
AS foo USING UNIQUE oid USING SRID = -1
```

This query results will show on the map the localization of all sand, gravel and mud within the coral reef region, like we can see on Figure 7. On Figure 8 we present the detailed information about the geographic features involved in the query.

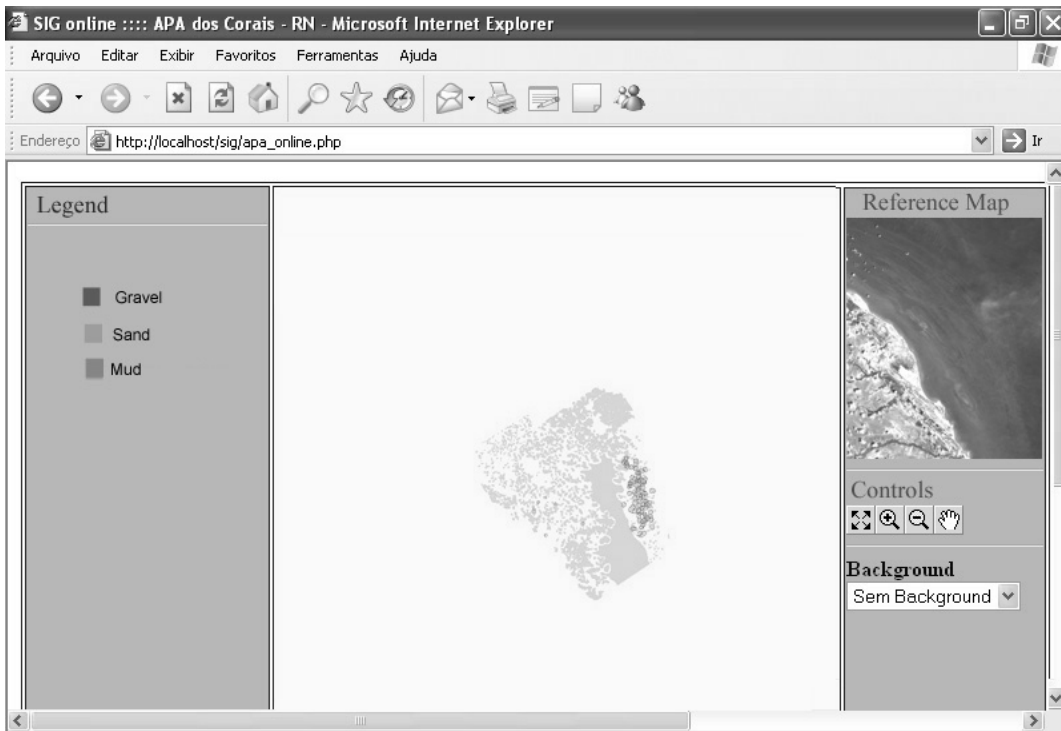


Figure 7. Biologist Query Results about “no-consolidate substrate”

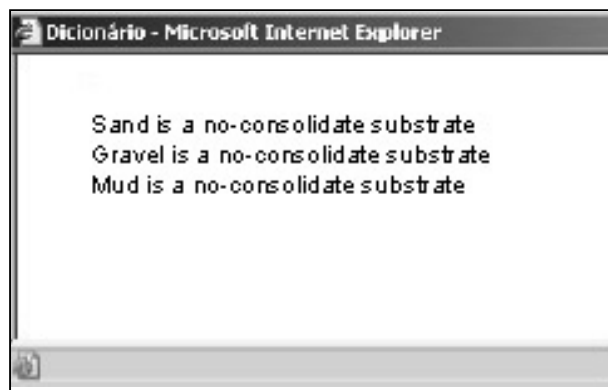


Figure 8. Pop-Up window to show detailed information about the geographic features.

These two complex query clauses generated by the system are some examples of how this system could help different type of users to search geographic databases. We have some updates to do in this system, but it is important so say that it is already implemented and that it works [Viegas 2006].

5. Conclusion

This work presented an ontology-based mechanism to access a geographic database. This mechanism allows that different users' communities, through geographic ontologies, access the same database, without know his internal structure.

It was developed a semantic layer that integrates the geographic information, through the use of ontologies and through the definitions about the semantic similarity between classes, which have the same concepts, but different nomenclatures.

In this work we developed ontologies for the coral reef domain, based on three points of view of different communities: the geologists community, the biologists and the tourists [Viegas 2006]. Each class of each ontology has a similarity with some class of other ontology, but the terms used to define them could be different. The proposal ontologies can be used as a navigation and query tool for the users, supplying the semantics information desired.

For future work we could define a ranking of similarity between the classes of the ontologies. This ranking could help the synonym classes mapping and decrease the time for search the geographic database, and also avoid the use of terms that could not be of interest of the specific activated application.

It is important to point out that although the domain chosen is a little restricted, the proposed architecture can be adapted to any spatial domain of multidisciplinary interest. And, in this case, the return of these new applications with the use of ontologies, using this architecture presented in this proposal, could be very satisfactory [Viegas 2006].

References

- [Breitman and Casanova 2005] Breitman, K. K. and Casanova, M. A. (2005). "Desenvolvimento de Ontologias para Engenharia de Software e Banco de Dados: Um tutorial Prático". Tutorial Apresentado no XIX Simpósio Brasileiro de Engenharia de Software. Uberlândia – MG.
- [Cardoso et. al 2005] Cardoso, R. C., Souza, F. F., Salgado, A. C. (2005) Using Ontologies to Prospect Offers on the Web. In: 7th International Conference on Enterprise Information Systems, 2005, Miami. ICEIS 2005 - Software Agent and Internet Computing, Lisboa - Portugal : INSTICC, 2005. v. 4. p. 200-207.
- [Dongilli et. al 2004] Paolo Dongilli, Enrico Franconi, and Sergio Tessaris (2004). Semantics driven support for query formulation. In Proceeding of the 2004 international workshop on description logics (dl'04), 2004.
- [Egenhofer and Herring 1991] M. J. Egenhofer and J. Herring. (1991). Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases. Technical Report, University of Maine, Orono.
- [Egenhofer et al. 2000] Egenhofer, M., Fonseca, F., Davis, C. and Borges, k. (2000) Ontologies and Knowledge Sharing in Urban Gis. CEUS - Computer, Environment and Urban Systems. 24(3): 232-251

- [Fonseca 2001] Fonseca, F. (2001) "Ontology-Driven Geographic Information Systems". Phd Thesis. University of Maine.
- [Franconi and Tessaris 2005] Enrico Franconi and Sergio Tessaris (2005). A unified logical framework for rules (and queries) with ontologies. In Rule Languages for Interoperability, 2005.
- [Guarino 1998] Guarino, N. (1998) "Formal Ontology and Information Systems". In: Formal Ontology and Information Systems (FOIS 98). Italy, 1998.
- [Jena 2004] JENA – A Semantic Web Framework for Java - Jena 2 Ontology API. (2004). <http://jena.sourceforge.net/ontology>.
- [Noy and Musen 1999] Noy, N. F., Musen, M. A.: SMART: Automated Support for Ontology Merging and Alignment. Workshop on Knowledge Acquisition, Modeling, and Management, Banff, Alberta, Canada, 1999.
- [OGC 1999] Open GIS Consortium, Simple Features Specification for SQL. Revision 1.1. May, 1999. <http://www.opengis.org/>.
- [OWL 2004] OWL WEB ONTOLOGY LANGUAGE GUIDE (2004). <http://www.w3.org/TR/owl-guide/>.
- [PostGis 2006] PostGis, 2006. PostGis Geographic Objects for PostgreSQL. Disponível em: <http://www.postgis.org>. Accessed: 07/08/2006
- [Protégé 2005] PROTÉGÉ ONTOLOGY EDITOR (2005). <<http://protege.stanford.edu/>>.
- [Sondheim et al. 1999] Sondheim, M., Gardels, K and Buehler, K. (1999) GIS Interoperability. in: P. Longley, M. Goodchild, D. Maguire and D. Rhind. Geographical Information Systems - Principles and Technical Issues. John Wiley & Sons, New York, pp. 347-358.
- [Surveys 1994] Surveys and Resource Mapping Branch – Ministry of Environment, Lands and Parks, British Columbia – Canadá (1994). Spatial Archive and Interchange Format: Formal Definition 3.1.
- [Viegas 2006] Viegas, R. F. "GeOntoQuery - Um Mecanismo de Busca em Bancos de Dados Geográficos Baseado em Ontologias". Dissertação de Mestrado. Programa de Pós-Graduação em Sistemas e Computação. Dimap – UFRN. Julho de 2006.