GeoMergeP: Supporting an Ontological Approach to Geographic Information Integration *

Agustina Buccella, Laura Perez and Alejandra Cechich
GIISCO Research Group
Departamento de Ciencias de la Computación
Universidad Nacional del Comahue
Neuquen, Argentina
{abuccel,acechich,lperez}@uncoma.edu.ar

Abstract

Since 90's, several new approaches providing solutions to integrated or federated systems have been defined. In particular, with respect to geographic information systems, proposals of integration did not take long time to appear. However, applicability of many of those approaches is still unlikely. In this context, we introduce an ontology-based approach – the GeoMergeP system – aiming at improving the capabilities of the integration process. In this paper we propose a methodology composed of two main process, semantic enrichment (by adding information about the ISO 19100 standard) and merging. The last one performs some tasks automatically and guides the user in performing other tasks for which his/her intervention is required. Finally, a plugin of the ontology editor, Protégé, is presented showing how the method is implemented through a case study.

1. Introduction

In last years geographic information is taken more and more attention. The construction of new technologies as GPS (Global Positioning System) devices, the new needs of the market, and the offer of free software and tools to recover, work and store geographic information, have generated a new explosion in this area. Old aspects, analyzed several years ago, have emerged to be combined to new research topics. New visualization techniques, new devices to capture data, and new requirements to implement these systems are some of reasons that contribute to put the geographic systems into research again.

Within all these new requirements, the integration of geographic information is an area in which new issues have reappeared. Several works in the literature propose novel and useful mechanisms in which integration processes, architectures or methodologies are described [12, 22, 31, 34]. Particularly in these works, geographic information is represented by using ontologies [15]. Ontologies appear to provide semantics to the real world allowing us to define a set of knowledge terms, semantic interconnections, and rules of inference on a particular domain.

In [8], we have analyzed eleven proposals that consider geographic information as sources to be integrated. Conclusions of this work are focused on three main aspects, *formal representation of ontologies*, *representation of the geographic information*, and *the integration process* itself. The first one refers to the logical formalism used to represent the ontologies. The second one refers to mechanisms to model geographic information in order to add more expressiveness. And the third aspect, the *integration process*, refers to methods and tools proposed to improve the process of finding similarities. In general, with respect to the last aspect, we can find three main overlapped mechanisms for that: using a top-level ontology, similarity functions, and/or logical inferences. For instance, proposals as [3, 14, 18, 19, 32] use a top-level ontology as a mediator among the source ontologies; and in [3, 17, 32], the top-level ontology is built in order to take advantages of the use of logical inferences in the integration process.

Considering these last aspects, we can conclude that the way geographic ontologies are represented has a direct impact on the way integration processes are implemented. Taking into account the assumption tested by [33] in which *"the more explicit semantics is specified in ontologies, the feasibility of matching will be greater"*, the first task involved in an ontology matching technique should be enriching the semantics of concepts in order to guarantee the effectiveness of the integration process.

Thus, we propose the GeoMergeP system based on two

* This work is partially supported by the UNComa project 04/E072 (Identificación, Evaluación y Uso de Composiciones Software).

1522-4902/08 $25.00 © 2008 IEEE
DOI 10.1109/SCCC.2008.9

main process: *semantic enrichment* and *merging*. In the first one, ontologies are generated by using the family of the ISO 19100 standards[1]. In particular we apply the ISO 19109 (*Rules for Application Schema*) [2] and ISO 19107 (*Spatial Schema*) [1] standards in order to improve the understandability of data. In the second process, we propose a merging methodology focused on three main phases: *unit*, *integration* and *system*. This merging process is mainly based on our work in [7, 9].

This paper is organized as follows: next Section presents related work in the literature taking into account two areas, geographic ontology modeling and ontology integration. Section 3 briefly describes the architecture of our GeoMergeP system together with the processes needed for the construction of its main components. Then, we describe the supporting tool created as a plugin of the Protégé-OWL editor[2]. Section 5 illustrates the two main processes in the GeoMergeP system by applying the plugin on a case study. Future work and conclusions are discussed afterwards.

2. Related Work

Our proposal combines two different but related sets of work. Firstly, works proposing new ways to model and enrich geographic systems must be taken into account. For example, proposals extending common data models such as Entity-Relationship diagrams [29] and object-oriented ones [6] have been presented in order to add geographic features to the data models. In addition, in [33], semantic enrichment techniques are applied to improve the integration process among conventional sources. Authors provide a formal language of first-order Quantified Modal Logic (QML) to build source ontologies as inputs of an ontology matching methodology.

With respect to works applying some of the ISO 19100 series of standards, the work presented in [4] defines the GeoUML approach through specialization of the ISO 19100 std. The work defines a set of integrity constraints (defined in OCL) on the spatial schema and topological relations focusing specially on the ISO 19107 std. Another related work presented in [21], uses the ISO 19109 standard as a base to create four quasi-ontologies written in GML [11]. The schema of these ontologies follows the structure of the ODGIS approach [13, 14]. An expert user is responsible for creating all these ontologies by using the information of all components the standard provide within its 4-level architecture (extracted from ISO 19109 std).

The work of Lemmens [26] proposes an applicative framework to support geo-processing on demand based on the semantics of distributed and heterogeneous geo-information and geo-services. In order to facilitate machine reasoning about geo-services (for integrability issues) a set of geo-information ontologies (written in OWL) is represented based on works by the Open Geospatial Consortium[3] (OGC) and the ISO 19109 and the ISO 19107 stds. The author proposes the use of different semi-automatic mapping methods based on reasoning over terminological axioms (TBox) and assertional axioms (ABox), and the use of similarity functions. In [27] authors propose a new formal Ontology for Transportation Systems (OTN) by translating and extending structures defined in the Geographic Data Files (GDF) standard. GDF is an ISO specification of how to store geographic information for transport systems. The resultant ontology is written in OWL, but reasoning is not applied. Although classes or features in the ontology can be represented by a geometry, relationships between these features are only non-geographic. The ISO 19107 std for representing the spatial schema is not considered in this work.

In the last work, Klien & Lutz [25] propose a method to add semantic annotations to geodata by extracting information on spatial relations. To do so, the method needs a geospatial ontology defining spatial concepts based on characteristic spatial relations and attributes. The language of the ontology is between DL and FOL. Authors implement spatial relations (e.g. *adjacent to*) as a sequence of GIS operations by defining algorithms based on the ISO 19109 std, the ISO 19107 std, and the Web Feature Service (OGC standard). In addition, Drexel University has developed a set of OWL ontologies based on ISO 19100 std[4]. However, some of them are incomplete.

Secondly, works proposing novel methodologies for data integration must be considered. We can find some surveys in the literature [12, 23, 24, 35] comparing and evaluating proposals focusing on data or information integration. The proposal of Euzenat & Shvaiko [12] is one of the more recent works describing and analyzing a wide set of ontology matching proposals. However, all these surveys are focused on conventional systems, and they do not analyze systems based on geographic information. In a previous work [8] we have described and analyzed several works integrating geographic information. As conclusions, we have found three main overlapped mechanisms to perform integration, *the use of top-level ontologies*, *logical inferences* and/or *matching functions*. Table 1 shows the more representative and referenced proposals classified by these three types.

One particularity of all these proposals is the use of ontologies to represent either top-level information or domain information or both of them. In the case of ODGIS [13, 14]

[1]created by the ISO Technical Committe 211 (ISO/TC211) - `http://www.isotc211.org/`

[2]`http://protege.stanford.edu/`

[3]`http://www.opengeospatial.org/`

[4]`http://loki.cae.drexel.edu/~wbs/ontology/list.htm`

Table 1. The three mechanisms for integration mapped to the proposals

	Top-level ontology	Logical Inferences	Matching Functions
BUSTER [34]		√	
Hakimpour et al. et al. [17, 18]	√	√	
Hess et al. [19]	√		√
MDSM [31]			√
ODGIS [13, 14]	√		
GeoNis [32]	√	√	
Aerts et al. [3]	√	√	
Our proposal	√	√	√
SIM-DL [22]			√
Quix et al. [30]	√		√

several ontologies are built (top-level, domain, and application ontologies) in order to provide more information about the domain and thus facilitate the integration process. But the activity of creating these ontologies is not an easy task and it demands a lot of effort. Other proposals as GeoNis [32], Aerts et al. [3] and Hakimpour et al. [17, 18] use a top-level ontology together with the advantages of a formal language (to make inferences) as tools to find more suitable mappings. The use of similarity functions, in proposals as Quix et al. [30], SIM-DL [22], MDSM [31] and Hess et al. [19], involves a set of functions that analyze the concepts and properties syntactically and semantically. In MDSM functions comparing similar structures are applied. In particular the use of these types of functions is useful when the ontologies are not complete (that is, there is absent information about the domain) and/or as starting point of an integration process when a top-level ontology is not involved. Proposals performing some manual step within the integration process require the assistance of an expert user to do so. For example, BUSTER [34] needs of an expert user although it uses inferences during the query process.

As we can observe in the Table 1, our proposal applies the three mechanisms to integrate ontologies. On one hand, source ontologies are enriched in their descriptions by applying the same formal rules and containing the same semantic structure (improving thus integrability). Then, logic capabilities and matching functions are combined in order to find more suitable mappings. The use of these three options makes our approach take advantage of using the standard in geographic information, the logic of data and the semantic information from ontologies.

3. The GeoMergeP System

The main goal of the GeoMergeP system is to provide a fully and user-transparent integration of the sources. In this way, we propose a layered-based architecture (Figure 1) consisting of four layers, *source information*, *ontologies*, *federation* and *presentation*. In the first layer source information is represented by local and autonomous geographic information systems. Formal ontologies are in the next layer representing information extracted from each source. In the integration layer two main components are specified in order to build the whole system: *logic* and *analysis* components. As a result, mapping rules and a global ontology are built involving concepts included in the formal ontologies. In order to query the system, potential users browse this global ontology through the presentation layer.

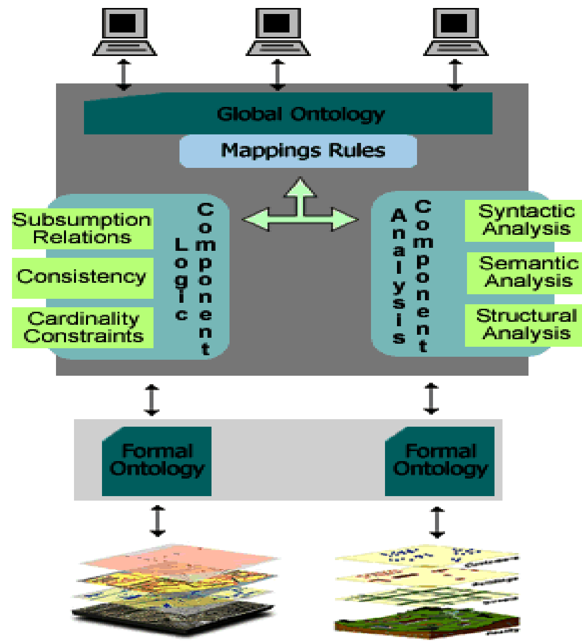

Figure 1. Architecture of the GeoMergeP System

In the next subsections the two main processes to build our system are described. The first process, *semantic enrichment*, defines the steps to create top-level and domain ontologies by applying the ISO 19100 standard for geographic information. Following, the *merging process* describes our merging methodology by using the Logic and Analysis Components showed in Figure 1. This process applies both logic capabilities and matching functions to find more suitable mapping. Once this process is finished, the system is available to be used.

3.1. The Semantic Enrichment Process

The semantic enrichment process is fulfilled through the formalization of standard geographic information. We are particularly focused on the ISO 19109 and the ISO 19107 standards. The ISO 19109 std defines a semantic modeling methodology to develop standardized concepts for geographic information. It has been developed to provide conceptual modeling of features and their properties from a universe of discourse. In addition, the ISO 19107 std specifies conceptual schemas for describing the spatial characteristics of geographic features, and a set of spatial operations consistent with these schemas. In our work, domain ontologies are defined for conceptualizing a particular domain by classifying concepts of a top-level ontology. This top-level ontology is based on the ISO 19109 and the ISO 19107 stds.

Figure 2 shows the 4-layer architecture of the ISO 19109 std. along with the ontologies used in our methodology. As we can see, the ontologies are specifically based on some levels of the standard in order to add interoperability aspects.

A *top-level ontology* and a *domain ontology* are built based on the information provided by the models of the standard. Gray arrows in the Figure show how the information flows among the models. Thus, the domain ontology is built considering both the General Feature Model (GFM) and the Application Schema [2]. The GFM is a meta-model of feature types. It defines the structure for classifying features used then to build the application schema. In this way, the domain ontology is located between two levels of abstraction (application and meta level) because it will be based on the GFM and will have features and associations defined by the application schema. The point is that the information stored by ontologies is different from the information stored in the application schema. An ontology is defined by how a community sees an specific concept, and an application schema is determined by how an application sees the same concept. A classical example is the concept *Car*. The application schema will store information about its model, color, function, etc., only if they are important for the application. On the other hand, the ontology should store all information about it because these features are necessary for being a car.

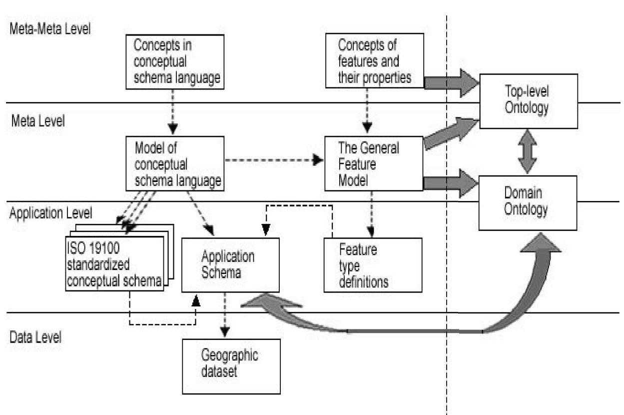

Figure 2. The 4-layer Architecture

Similarly, the top-level ontology is also in the middle of two layers, meta and meta-meta level layers. The information represented in this ontology will be based on both, the structure of the GFM, and the general features of the model being built.

In our work, as both ontologies – top-level and domain – have to be based on the standard before being created, the components of the ontologies are enriched in their descriptions through the metaclasses (from GFM) which they are instance of and the schemas on which they are based. The GFM acts as a top-level ontology providing an structure and semantics to classify main components of the spatial ontologies. Domain ontologies are then defined as subclasses of this top-level ontology.

Both ISO 19109 and ISO 19107 stds use UML (version 1.1) together with OCL (Object Constraint Language) constraints as the conceptual schema language (CSL) to model the system. Since the semantics of OCL is based on first-order logic, these constraints make undecidable any reasoning over. In order to take advantage of decidable theories we create the top-level ontology by modifying and translating the GFM to DL. The translation is based on the formalism proposed in [5] representing an UML class diagram with spatial information specified in the ISO 19109 and ISO 19107 stds. As the language used to represent the ontology is $\mathcal{ALCQI}$, the reasoning is EXPTIME-complete.

3.2. The Merging Process

The merging process involves the task of merging the geographic sources in order to create a global vocabulary by defining two main components (Figure 1). These components – logic and analysis – are applied on different parts of the merging process which is composed of three main phases: *unit*, *integration* and *system*.

In the *Unit Phase* each enriched ontology (during the last process) is analyzed separately in order to find possible inconsistencies or ambiguities. To do so, the logic component is applied to each system. This component uses a reasoner (such as Racer[5] [16] or FACT++[6] [20]) in order to take advantage of the logic of data. If an inconsistency is found, an expert user is responsible for solving it.

In the *Integration Phase* three processes are responsible for matching two normalized ontologies in order to create the global ontology. It contains the general concepts users will use to query the integrated system. In addition, a set of mappings are returned in order to represent the matching among the ontologies. *Merge*, *General Analysis*, and *Specialized Analysis* are the processes of this phase. To do the first process, both ontologies of each system are joined by using generalization/specialization relations. In this way, the ontologies are taken as they are returned from the unit

[5] http://www.sts.tu-harburg.de/~r.f.moeller/racer/

[6] http://owl.man.ac.uk/factplusplus/

phase. Following, the two ontologies belonging to two different systems are merged. The merge process is performed by matching the classes that are part of the standard (metaclasses). As both ontologies have the same superclasses, merging is an easy task.

Once the merge process is finished, the *General Analysis* starts. It applies two types of analysis: syntactic and semantic. Within the syntactic analysis three functions are used as follows [7].

- The *edit distance* function, which considers the number of changes that must be done to turn one string into the other, and weights the number of these changes with respect to the length of the shortest string.
- The *trigram* function, which is based on the number of different trigrams in two concepts or strings.
- And the *check constrains* function, which compares the constrains applied to the properties, for example, cardinality constraints. Only when both properties have the same restrictions, the function returns 1; otherwise it returns a percentage according to the number of restrictions that are the same.

The first two functions compare the names of the concepts in a different way. Thus, both functions return a different similarity result depending on the syntax of the compared names.

Then, in the semantic analysis, a thesaurus is used to extract synonym relationships between the concepts of the ontologies. These relationships are necessary because synonyms (in general) are not similar syntactically. In this case, WordNet[7] is used as the thesaurus. The *Specialized Analysis* performs a structural comparison by applying the similarity function described in [7, 31]. This function compares the number of properties that the classes have in common and analyzes them in a hierarchy (by calculating the depth of the most common superclass between the classes).

Finally, it is possible the processes executed before generate inconsistencies within this final ontology. Therefore, the *System Phase* re-normalizes the global ontology created in the last phase. Like in the unit phase, a logic process is applied, where the reasoning system is used once more to analyze possible subsumption relations and inconsistencies in the global ontology.

User participation is also needed in this phase. User here have two types of functions, committing the options the reasoner system detects and testing the global ontology.

[7]`http://wordnet.princeton.edu/`

4. Implementing a supporting tool

By experimenting the semantic enrichment process we create source ontologies and enriched ontologies using the Protégé-OWL editor. As $\mathcal{ALCQI}$ is a fragment of OWL-DL [28], our top-level ontology (from the GFM) can be translated to this language without losing expressiveness. In this way, we have built a plugin for Protégé implementing our merging process. In this section, we describe the design of the Protégé plugin called OWLSim.

During the design process, the Responsibility Driven Design (RDD) [36] model was used to keep our focus on the behavior of our software. This methodology helps us to identify the application's responsibilities and to divide them into collaborative objects.

Figure 3 shows our plugin's architecture. In the first place, the Transactions (*Control and business logic*) component includes the objects that are responsible for the control and business logic. Furthermore, this component mediates the interaction between Domain Model and Presentation components in order to avoid direct dependency between them. This component contains classes to coordinate the merging process described in the last section. It is responsible for running the logic process and applying the similarity functions in specific moments within the merging methodology. To do so, the component uses the Domain Model component to obtain information about the domain and top-level ontologies. The Presentation component is used when decisions must be taken by the user such as accepting or rejecting a proposed mapping.

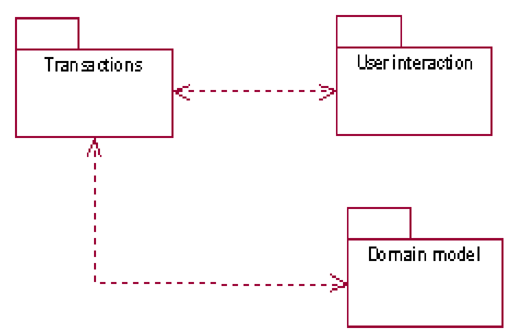

Figure 3. Plugin's architecture

Secondly, the Domain Model (Abstraction) component contains all those objects that represent the domain of both the methodology and the geographic information sources. Each information source (represented by a domain and a top-level ontology) is loaded on an object diagram to allow the methodology to obtain classes, properties, and restrictions. These objects are then used by the Transactions component to compare ontologies each other.

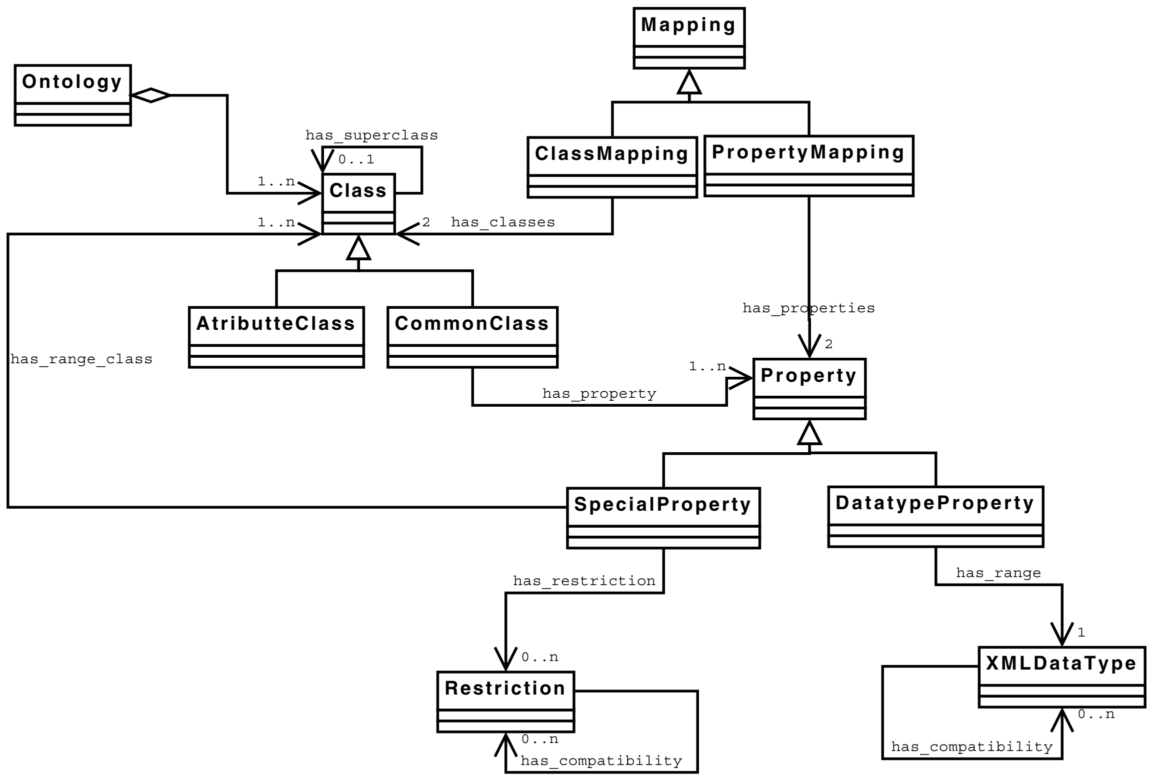

Figure 4. Part of the domain class diagram

Lastly, the User Interaction component (Presentation) is structured into objects that provide window, menu, and dialog functionality. They manage inputs and translate them into service requirements. This component is used by the Transactions component to allow users perform operations such as selecting ontologies, selecting specific concepts of the ontologies, accepting mappings, etc.

4.1. Designing classes

One of the most important activities in Object-Oriented design is to identify object classes. Thus, we move from the requirements and descriptions – the method specification explained in previous sections (behavior that the plug-in must accomplish), the definition of OWL ontologies and the Protégé model specification – to find and describe the most important classes.

4.1.1 Domain and application-specific objects

Domain objects represent concepts in a specific field of interest. In our domain, they are the ontologies and their elements, the mappings between them, and the similarity method.

Figure 4 shows part of the domain class diagram (within the Domain Model component) using UML notation – the part where the ontology and both their elements and mappings are modeled. There are classes that model the most important components of an OWL ontology, such as classes, properties and restrictions.

In addition, we also take into account the characterization of the ontology elements that the similarity method embodies (*Attribute_Class*, *Common_Class*, *Datatype_Property* and *Special_Property*). As shown in the diagram, both classes *Attribute_Class* and *Common_Class* are modeled as an specialization of the *Class* class. Because of their differences, the method gives different treatment to each of them. In addition, the *has_superclass* relation represents taxonomic relations in an ontology. Regarding the *Property* class an specialization of two different subclasses is modeled: *Datatype_Property* and *Special_Property* in accordance to differences among them. Both special properties and datatype properties ranges are different, so two distinct relations *has_classrange* and *has_XMLDatatyperange* are modeled to associate them to the range of *XMLDatatype* and *Class* classes. Further, the *has_compatibility* association shows that each data type might be compatible with other data types. Unlike *Common classes* that might have both types of properties, the *Attribute classes* have no properties; thus, the *has_property* association is between the *Common_Class* subclass and the *Property* superclass. Finally, properties restrictions are also modeled.

Following, mappings found by the method are contained in the *Mapping* class. The *Property_Mapping* and *Class_Mapping* classes are its subclasses. The former class involves classes using the *has_classes* and the latter class involves properties using the *has_properties*.

The *Similarity-Searcher* class represents an abstraction from the merging process. It is part of the Transactions component and it is subclassified into three subclasses: *DataTypePropertyAnalysis*, *SpecialPropertyAnalysis*, *ClassAnalysis*. Each of them represents the analysis method part over the elements identified in an ontology.

4.1.2 Similarity searching

As a result of being too complex to be implemented by a single class, main responsabilities of the *SimilaritySearcher* class are divided into sub-responsibilities reassigned to collaborating classes. Each class implements a quite different similarity search method depending on the elements of the ontologies, as Figure 5 shows. In the figure we can see that when ontology classes to be analyzed are *Common_Classes*, a datatype and special property analysis through the *DataType Property Analysis* and *Special Property Analysis* classes, will take place. Finally, the class analysis (through *Class Analysis* class) is carried out.

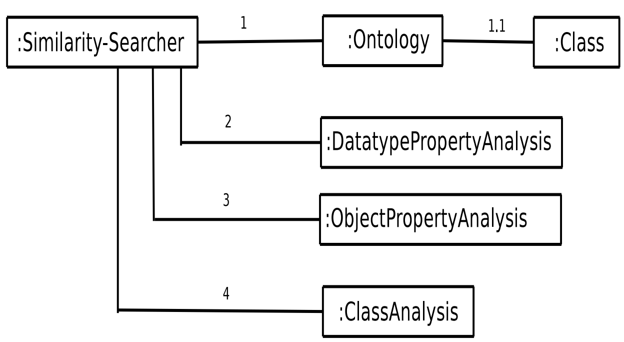

Figure 5. Similarity_Searcher collaborates with other objects to find similarities.

Each of these classes implements the different parts of our method. For instance, the DataType Property Analysis component implements the comparison between datatype properties applying all the functions (syntactic and semantic) described in Section 3.2.

5. A Case Study - Using the Plugin

In this section we present a real case study that allows us to show both the plugin interface and how the method works. Conceptual models used as starting points to build the ontologies came from XML files of the Italian Agency for Environmental Protection and Technical Services[8] (APAT). Data collected by APAT include climatic, hydrometric, cartographic and water pollution measures. In this case study we use two of these models [10]: *Storico* storing historical measures about rains and temperature, and *Temporeale* containing real-time measures about rains and temperature registered by stations.

The semantic enrichment task is the first activity within our integration methodology. This task enriches domain ontologies by standardizing and formalizing concepts within them. As result, all enriched ontologies will contain the same structure due to all components are subclassifying the same model. The formalized GFM acts as a top-level ontology classifying the elements of the ontology and making the integration easier.

Figure 6 shows an extract of the *Storico* enriched ontology in Protégé. In the figure we see the spatial attribute "centerPoint" represented as subclass of GF_SpatialAttributeType. It contains two constraints (on the right of the figure): the attribute "valueType" can only have GM_Point values (*iso19109:valueType only iso19107:GM_Point*) denoting that the datatype of "centerPoint" is GM_Point; and the association "carrierOfCharacteristics" can only have StationType values (*iso19109:carrierOfCharacteristics only StationType*) denoting that "centerPoint" is an attribute of the class StationType. In addition, other classes, attributes and associations are represented in the figure. For instance, the association "op_precipitation" is represented as a subclass of GF_AssociationType with a role "hasPrecipitation" as subclass of GF_AssociationRole.

Both ontologies, Storico and Temporeale, were translated and created manually. The resultant OWL sublanguage of them is OWL-DL.

Once the semantic enrichment process is finished for these two ontologies we must open the plugin tab widget in Protégé to choose the owl files containing them. If both owl files are loaded successfully, the mapping layout screen appears (see Figure 7). As we can see it is divided into two main panels. On the left side, there is the *select source classes form* that holds each of the selected ontologies class hierarchy, sucha that a pair of classes to be compared could be chosen. On the other side, the right one, there is the *show class and property mappings form* in which mappings between classes and properties found by the process are shown.

Once two classes are selected to be compared, the analysis takes place over those selected concepts by clicking the *"Map Selected"* button. As a result, for any mapping that is found, a confirmation is required from the user through a confirmation window as shown in Figure 7.

[8] `http://www.apat.gov.it/site/it-IT/`

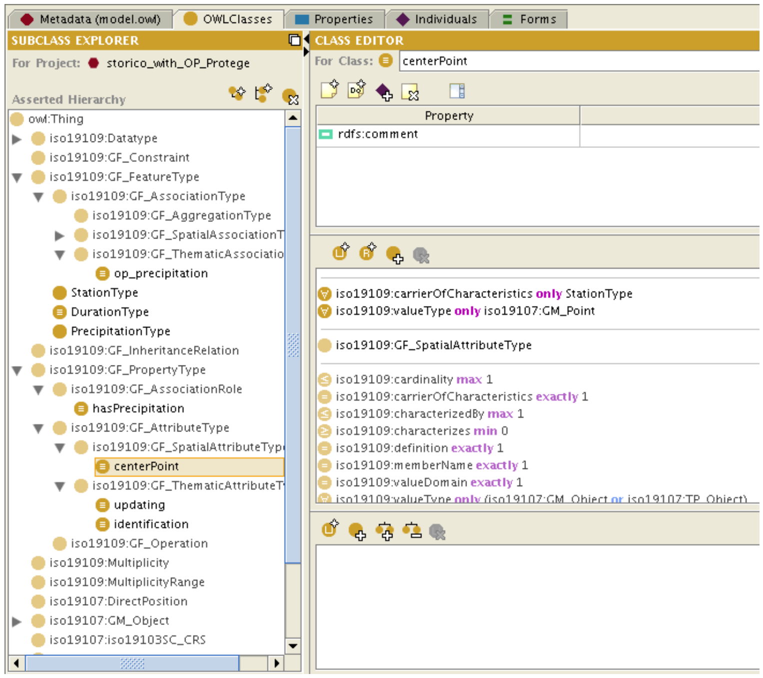

Figure 6. Storico enriched ontology in Protégé

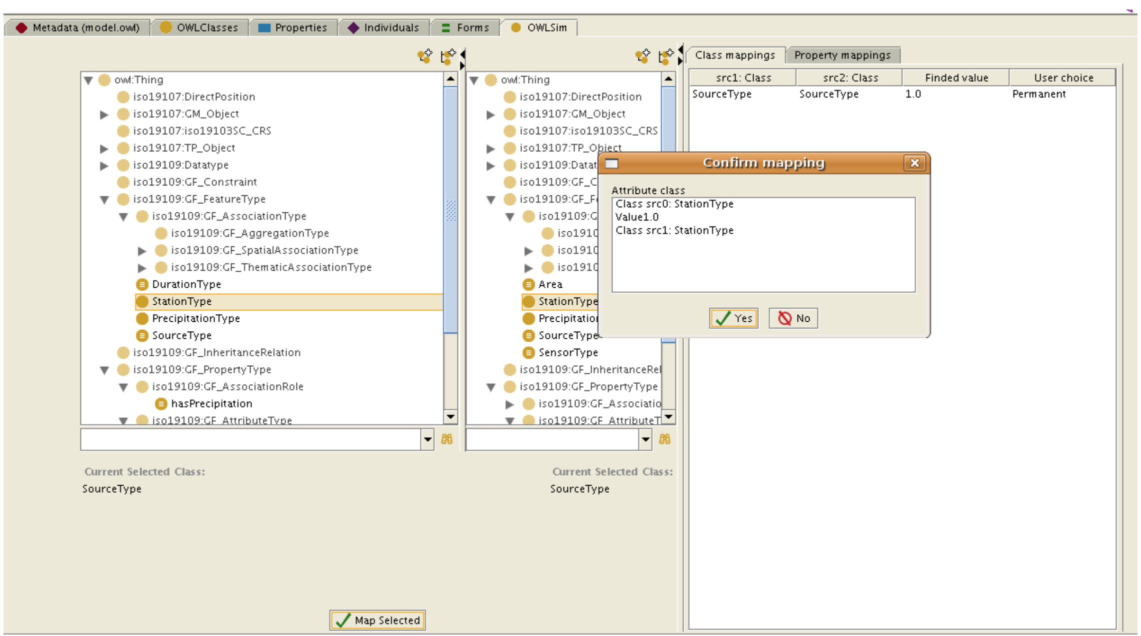

Figure 7. OWLSim plugin interface in which the merging process takes place

As an example of how the method works, when the classes StationType of both ontologies are compared the process triggers the comparison of all their attributes and associations. For instance, by using the association "linkBetween", the method compares all associations each subclass of GF_FeatureType has. In the case of attributes, the association "carrierOfCharacteristics" is retrieved from each GF_PropertyType. In this way, once all attributes and associations are analyzed and compared syntactic and semantically, the method starts the structural analysis calculating the number of them they have in common.

During the merging process the plugin analyzes the ontologies as graphs, taking into consideration both taxonomic and non taxonomic relationships among terms. As the ontology graph may contains cycles, a *cycle detection* technique is implemented in order to avoid visiting the same node twice.

Finally, all mappings found are then used to create the global ontology. A reasoning system is again applied in order to find inconsistencies or new subsume relations.

Preliminary experiments applying the GeoMergeP system on the APAT conceptual models have shown good results with respect to the reliability of mappings found and the performance of the process [9].

6. Lessons Learned

1. *Redundant information is highly minimized when normalized ontologies based on the ISO 19109 and 19107 stds are inputs of an integration process*. The quality and the way ontologies are built is crucial in every integration process. Redundant information can generate inconsistencies affecting the understandability of the concepts of ontologies and consequently the process of finding mappings. Our case study shows how the source ontologies are enriched by distributing classes, properties and restrictions in a standard top-level ontology. Thus, our merging process receives normalized ontologies structured into the same semantic concepts.

2. *The use of a top-level ontology as a common structure among all source ontologies avoids the problem of independence of top-level ontologies*. In [8] we have analyzed several proposals of geographic integration approaches in which source ontologies commit to the same top-level ontology to allow the reasoning system to start the integration process. Although this strategy is needed to take advantage of the benefits of logic inferences, the way top-level ontologies are proposed interfere with the independence of the system (as all communities must agree on the same structure). In this work, we propose an standard top-level ontology based on the GFM which can be shared by any community. In addition, as the ontologies can be merged in only one structure, an automatic implementation of the merging process is possible.

3. *Formal Representation of Ontologies*. Basic reasoning tasks such as class consistency, class subsumption and class equivalence, can be applied over our enriched ontologies by using reasoners. As we mentioned in last section, the reasoning is EXPTIME-complete.

7. Conclusion and Future Work

In this paper we have presented the GeoMergeP system for integrating geographic sources. In particular we have defined two processes to build the system – the semantic enrichment and the merging process. In the first one, we build a common top-level ontology by adding information about geographic standards. Thus, the second process can apply a combination of several matching techniques in order to find suitable mappings. An implementation of it as a Protégé plugin has been presented and a beta version is available at `http://giisco.uncoma.edu.ar-ResourcesLink`.

Currently, we are empirically validating our proposal in the context of the APAT Information Broker project [10].

References

[1] Geographic information. Spatial Schema. International standard 19107, ISO/IEC, 2003.

[2] Geographic information. Rules for Application Schema. Draft International Standard 19109, ISO/IEC, 2005.

[3] K. Aerts, K. Maesen, and A. Van Rompaey. A practical example of semantic interoperability of large-scale topographic databases using semantic web technologies. In *Proceedings of the AGILE'06: 9th Conference on Geographic Information Science*, pages 35–42, Visegrád, Hungary, 2006.

[4] A. Belussi, M. Negri, and G. Pelagatti. An iso tc 211 conformant approach to model spatial integrity constraints in the conceptual design of geographical databases. In *ER (Workshops)*, pages 100–109, 2006.

[5] D. Berardi, D. Calvanese, and G. D. Giacomo. Reasoning on uml class diagrams. *Artificial Intelligence*, 168(1):70–118, 2005.

[6] K. Borges, C. Davis, and A. Laender. Omt-g: An object-oriented data model for geographic applications. *Geoinformatica*, 5(3):221–260, 2001.

[7] A. Buccella and A. Cechich. Towards integration of geographic information systems. *Electronic Notes in Theoretical Computer Science*, 168:45–59, 2007.

[8] A. Buccella, A. Cechich, and P. Fillotrani. Ontology-driven geographic information integration: A survey of current approaches. *Computers & Geosciences Special Issue on Geoscience Knowledge Representation in Cyberinfrastructure*, 2008.

[9] A. Buccella, D. Gendarmi, F. Lanubile, G. Semeraro, A. Cechich, and A. Colagrossi. A layered ontology-based architecture for integrating geographic information. In Kacprzyk and Janusz, editors, *Studies in Computational Intelligence Series*. Springer. To appear, 2008.

[10] F. Calefato, A. Colagrossi, D. Gendarmi, F. Lanubile, and G. Semeraro. An information broker for integrating heterogeneous hydrologic data sources: A web services approach. In A. Xu, L. Chaudhry, and S. Guarino, editors, *Research and Practical Issues of Enterprise Information System, IFIP Series (Springer)*, volume 205, 2006.

[11] O. Consortium. Geography markup language. Available in http://www.opengis.net/gml/. Last access in august 2006, 2006.

[12] J. Euzenat and P. Shvaiko. *Ontology Matching*. Springer-Verlag, Heidelberg, Germany, 2007.

[13] F. Fonseca. *Ontology-driven Geographic Information Systems*. PhD Dissertation, University of Maine, Orono, ME, USA, 2001.

[14] F. Fonseca, C. Davis, and C. Câmara. Bridging ontologies and conceptual schema in geographical information integration. *Geoinformatica*, 7(4):307–321, 2003.

[15] T. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[16] V. Haarslev and R. Möller. Racer system description. In *Proceedings of the IJCAR '01: 1st International Joint Conference on Automated Reasoning*, Lecture Notes in Computer Science 2083, pages 701–706, London, UK, 2001.

[17] F. Hakimpour. *Using Ontologies to Resolve Semantic Heterogeneity for Integrating Spatial Database Schemata*. PhD Dissertation, Zurich University, Switzerland, 2003.

[18] F. Hakimpour and A. Geppert. Resolving semantic heterogeneity in schema integration. In *Proceedings of the FOIS '01: International Conference on Formal Ontology in Information Systems*, pages 297–308, Ogunquit, Maine, USA, 2001. ACM.

[19] G. N. Hess and C. Iochpe. Ontology-driven resolution of semantic heterogeneities in gdb conceptual schemas. In *Proceedings of the GEOINFO'04: VI Brazilian Symposium on GeoInformatics*, pages 247–263, Campos do Jordão, Brazil, 2004.

[20] I. Horrocks. The fact system. In *Proceedings of the TABLEAUX'98: Automated Reasoning with Analytic Tableaux and Related Methods*, Lecture Notes in Computer Science 1397, pages 307–312, Berlin, 1998. Springer-Verlag.

[21] S. Jang and T. J. Kim. Modeling an interoperable multimodal travel guide system using the iso 19100 series of international standards. In *Proceedings of the GIS '06: 14th annual ACM international symposium on Advances in geographic information systems*, pages 115–122, New York, NY, USA, 2006. ACM Press.

[22] K. Janowicz. Sim-dl: Towards a semantic similarity measurement theory for the description logic *cnr* in geographic information retrieval. In *OTM Workshops (2)*, pages 1681–1692, 2006.

[23] Y. Kalfoglou and M. Schorlemmer. Ontology mapping: the state of the art. *The Knowledge Engineering Review*, 18(1):1–31, 2003.

[24] M. Klein. Combining and relating ontologies: an analysis of problems and solutions. In *Proceedings of the IJCAI'01: 17th International Joint Conferences on Artificial Intelligence*, pages 53–62, Seattle, WA, USA, 2001.

[25] E. Klien and M. Lutz. The role of spatial relations in automating the semantic annotation of geodata. In A. Cohn and D. Mark, editors, *Proceedings of the COSIT'05: International Conference on Spatial Information Theory*, LNCS 3693, pages 133–148, Ellicottville, NY, USA, 2005.

[26] R. Lemmens. *Semantic interoperability of distributed geoservices*. PhD thesis, Delft, Nederlandse Commissie voor Geodesie (NCG). Netherlands Geodetic Commission NCG, 2006.

[27] B. Lorenz, H. J. Ohlbach, and L. Yang. Ontology of transportation networks. Technical Report A1-D4, REWERSE, IST-2004-506779, 2005.

[28] C. Lutz, D. Walther, and F. Wolter. Conservative extensions in expressive description logics. In *IJCAI*, pages 453–458, 2007.

[29] C. Parent, S. Spaccapietra, and E. Zimányi. Spatio-temporal conceptual models: data structures + space + time. In *Proceedings of the GIS '99: 7th ACM International Symposium on Advances in Geographic Information Systems*, pages 26–33, New York, NY, USA, 1999. ACM Press.

[30] C. Quix, L. Ragia, L. Cai, and T. Gan. Matching schemas for geographical information systems using semantic information. In *OTM Workshops (2)*, pages 1566–1575, 2006.

[31] M. Rodríguez and M. Egenhofer. Comparing geospatial entity classes: An asymmetric and context-dependent similarity measure. *International Journal of Geographical Information Science*, 18(3):229–256, 2004.

[32] L. Stoimenov, A. Stanimirovic, and S. Djordjevic-Kajan. Discovering mappings between ontologies in semantic integration process. In *Proceedings of the AGILE'06: 9th Conference on Geographic Information Science*, pages 213–219, Visegrád, Hungary, 2006.

[33] N. N. Tun. Semantic enrichment in ontologies for matching. In *Proceedings of the AOW '06: Second Australasian workshop on Advances in ontologies*, pages 91–100, Darlinghurst, Australia, Australia, 2006. Australian Computer Society, Inc.

[34] U. Visser. *Intelligent Information Integration for the Semantic Web*. Lecture Notes in Computer Science 3159. Springer, Heidelberg, Germany, 2004.

[35] H. Wache, T. Vögele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. Hübner. Ontology-based integration of information - a survey of existing approaches. In *Proceedings of the IJCAI'01: 17th International Joint Conferences on Artificial Intelligence*, pages 108–117, Seattle, WA, 2001.

[36] R. Wirfs-Brock and A. McKean. *Object Design: Roles, Responsibilities, and Collaborations*. Addison-Wesley 2003, 2003.