

IMPROVING ANALYSIS PATTERNS IN THE GEOGRAPHIC DOMAIN USING ONTOLOGICAL META-PROPERTIES

Evaldo Silva, Jugurta Oliveira and Gabriel Gonçalves
Department of Informatics, Federal University of Viçosa, 36570-000, Viçosa, Brazil

Keywords: Conceptual Modeling, Analysis Pattern, Ontology, Sharing of Knowledge.

Abstract: This article shows the improving of analysis patterns in the geographic domain through ontological meta-properties, where each pattern's class has its concept analyzed ontologically. This improvement also permits to restructure the class diagram of analysis patterns, increasing the reuse quality. Besides improving the analysis patterns, it is proposed one more topic in the template for analysis patterns documentation. This topic is based on the specification of the main classes defined during the process.

1 INTRODUCTION

Some researchers consider ontologies and patterns as mechanisms used during the phases of specification and construction of information systems (Guizzardi et al., 2002) (Devedzic, 1999) (Hamza, 2004). The both are also defined as mechanisms to reuse of knowledge. However, ontologies are created by domain specialists, where knowledge is reusable in a widely way. Analysis patterns are discovery and reused in application design level, by information systems designers.

According to Guarino (1995) "ontology is a specification of a conceptualization described by concepts and relationships that can exist in the real world. The conceptualization can be developed containing terminologies and vocabularies, establishing properties and allowing the knowledge to be reused, avoiding the rework or rediscover of equivalent terminologies."

The conceptual schema of an analysis pattern is modelled through collection of classes and associations, whose follows the notation of UML (Unified Modeling Language), it has some meaning in the context of an application, where the same structure can be considered valid to other applications. That causes the same conceptual schema to be reused to compose the modeling of other information systems (Fernandez, 1998).

There are researches that have been developed aiming the improvement of the reuse of analysis patterns through ontological approaches, increasing

the productivity and quality during the conceptual modeling of systems (Guizzardi et al., 2002) (Devedzic, 1999) (Hamza, 2004).

Guizzardi et al. (2002) propose an approach to derive frameworks from domain ontologies. They have shown how to implement an application of information systems using such approach in the developing process. According to the authors, domain ontologies are used to support the analysis activity, it is necessary its transformation in the view of class diagram, without losing the explicit representation of the knowledge.

Devedzic (1999) approaches that the concepts used in ontologies and software patterns can superpose each other, enriching the knowledge about the domain. He proposes the use of software patterns as a source of knowledge during the process of conception and developing of ontologies. In a similar way, Hamza (2004) proposes an ontological approach to improve the quality of analysis patterns reuse, where the knowledge taken from the patterns is used to the development of ontologies. The patterns are grouped and reused through ontologies, enlarging the knowledge about the domain.

Hamza (2004) also points some problems related to the reuse of analysis patterns, which can be resolved through an ontological approach: (a) the lack of formalism; (b) redundancy of patterns; (c) indefiniteness of the reuse domain.

In this way, this article shows the improvement of analysis patterns through ontological meta-properties in the geographic domain. The improvement permits restructure the class diagram

of analysis patterns, increasing the quality of reuse. It is also proposed the documentation of analysis patterns based in the specification of the main classes defined during the process.

To exemplify the process of analysis patterns improvement, it will be used the Parceling of Urban Land analysis pattern, proposed in Lisboa et al. (2002). As a technique to the application of ontological meta-properties, it will be used the VERONTO (ONTOlogical VERification) developed by Villela (2004). VERONTO technique is used to the verification and adequacy of class diagrams of UML (*Unified Modeling Language*), based on the philosophical notions of rigidity, dependency and identity, defined by Guarino and Welty (2000).

This article is organized as follows. Section 2 describes the VERONTO technique. Section 3 discusses the analysis patterns documentation using one more topic, which identifies the main classes of the domain. Section 4 shows the improvement of the Parceling of Urban Land analysis pattern. Section 5 presents some final considerations.

2 VERONTO TECHNIQUE

The VERONTO technique, proposed by Guarino and Welty (2000), uses meta-properties as rigidity, dependence and identity, in the validation of conceptual models specified through UML class diagrams. Ontological meta-properties are based on philosophical notions of essence, dependence and identity defined by Guarino and Welty.

When developed this technique, Villela et al. (2004) applied these meta-properties in elements of the class diagram, making it possible to apply taxonomic restrictions about the relationship between classes. Such restrictions are based on ontological analysis of meta-properties, applied in the elements of the class diagram.

2.1 Representation of Philosophical Notions through Ontological Meta-Properties

The notion of *essence* is represented by the meta-property rigidity. The meta-property of rigidity is about the knowledge of how classes can change in the course of time and other can't. A meta-property is rigid (+R) (~, anti-rigid) when an element of the domain that instantiates such property will continue to instantiate it during all its existence (Guarino and Welty 2000). For example, in a conceptual schema

which specifies an application of urban transport, an instance of the FAST TRAFFIC ROAD class can stop being a fast track road to become a local road, but it will always be a road within the context of the transportation system. Thus, it is possible to analyze that the FAST TRAFFIC ROAD class is anti-rigid (~R), as a fast traffic road will not be like that for all its existence. However, the ROAD class is rigid (+R), because in the domain of urban transport application, an instance of ROAD will be like that for all its existence.

The philosophical notion of *dependence* is about relations of dependency that can be intrinsic and extrinsic, represented by +D (-D, otherwise) (Guarino and Welty 2000).

According to Villela et al. (2004) dependency involves different relationships, such as the ones existing among people and their parents, being extrinsic and intrinsic. An intrinsic property is inherent to the individual, non dependent on other individuals, like having a heart or a fingerprint. Extrinsic properties are not inherent and they have a relational nature, like "being the mayor of the São Paulo City". For example, an instance of the DISTRICT SEAT class is externally dependent on the MUNICIPALITY class, as it can only be a district seat if there is a municipality in which it was created.

At last, *identity* is about the way we recognize individual entities, and it is based on the concept of Identity Condition (IC), proposed by Guarino and Welty (2000). A class that has an identity condition is represented by the symbol +O (-O, otherwise), only if it is rigid and executes an IC (+I) (-I, otherwise). A non-rigid class can execute an IC, if and only if this is inherited by a class that has a rigid meta-property, which subsume it. For example, the subclass FAST TRAFFIC ROAD, classified as non-rigid, can only execute its IC's, inheriting them from rigid meta-properties that superpose it, as a meta-property (+O) from the super class ROAD.

The meta-properties described above create some natural restrictions in the taxonomic structure of ontology (Guarino and Welty, 2000), supporting the analysis and adequacy of conceptual models. Be it two arbitrary classes (Φ and ψ). The notation Φ^M is used to show that a class Φ has the meta-property M , with the restrictions showed in Table 1.

Table 1: Taxonomic restrictions (Guarino, 2000).

Meta-Properties	Restrictions
Rigidity	ϕ^{-R} can't subsume ψ^{-R}
Identity	ϕ^{+I} can't subsume ψ^{-I}
Dependency	ϕ^{+D} can't subsume ψ^{-D}

Besides the meta-properties seen before, it is part of the VERONTO technique the addition of rules to the use of relationships, based on the rules proposed by Wand et al. (1999) that follow:

- a) Rules to optional relationships:
 - Optional associations (minimum cardinality 0) must be avoided.
 - Acquisition or loss of an association must be modeled as a change of class.
 - The capacity of instances of a class to participate in an association, without losing properties, must be modeled as a sub classification.
- b) Rules to aggregation
 - Each class component must be associated with a class composed through an aggregation.
 - The emergent properties of the composed class (“whole”) must be modeled as attributes and associations.

Practical implications of such rules, as well as some advantages are discussed by Villela (2004).

3 DOCUMENTING ANALYSIS PATTERNS WITH THE RIGID CLASSES

The analysis patterns description is important to register the definitions of the context in which the analysis patterns should be reused. Besides, it serves as a repository, allowing the sharing of the knowledge by designers of information systems.

The analysis patterns template proposed on this work follows the model proposed in Meszaros and Doble (1998), which defines the following topics that specify the catalog: Name, Problem, Context, Forces, Solution, Participants, and Related Patterns. Nevertheless, this specification does not follow a generic format. Some analysis patterns are documented and cataloging in a narrative form, but generally there are coincidences in the description of the analysis patterns in a topic structure. Though there is not a generic topic structure to use, the topics listed before have a basic format to the specification of analysis patterns.

Based on the template’s structure mentioned in Meszaros and Doble (1998), it will be specified one more topic, called Rigid Classes. The Rigid Classes topic is based upon the taxonomy concept “backbone” defined by (Guarino and Welty 2000). The “backbone taxonomy” is defined by rigid classes

found in the hierarchical structure of ontology. Those classes model the main concepts, embracing the entire domain and describing the basic structure which serves as a reference to the reuse of knowledge.

Thus, after the improvement of patterns through the VERONTO technique, it is also possible to identify the rigid classes that exist in the analysis patterns. The documentation based on the rigid classes aims at helping the designer in the reuse of patterns, identifying the main classes of the domain.

4 IMPROVING THE PARCELING OF URBAN LAND PATTERN

The patterns improvement process occurs through the ontological analysis, classifying each class based on the meta-properties described before.

To develop this section, it was chosen the Parceling of Urban Land analysis pattern (Figure 1), proposed in (Lisboa F. et al. 2002). The diagram follows the UML-GeoFrame model, a UML profile to model geographic database (Lisboa F. and Iochpe 2008). It was depicted using a CASE tool called ArgoCASEGEO (Lisboa F. et al., 2004).

The result of the ontological analysis about the Parceling of Urban Land pattern is the following.

According to Demographic Census from IBGE - Brazilian Institute of Geography and Statistics (IBGE 2008), a city is classified as an urban area of city (municipal seat) or village (district seat). Therefore, a village or district seat, for example, can be emancipated and become municipal seat. So, the CITY class is classified as: Anti-rigid (~R) – all city will not be like that for all its existence; it does not give identity (-O) – all city can exist in several urban regions as district or municipal seats; it executes identity (+I) – it executes identity inherited from seat; it is non-dependent (-D) – it does not depend on any other class.

The ADMINISTRATIVE DIVISION class models the subdivision of municipal territory. It is classified as: Rigid (+R) – every administrative division will be like that for all its existence; It gives identity (+O) – all instance of the ADMINISTRATIVE DIVISION class can be identified through a territorial division; it executes identity (+I) – if it gives identity it can also execute it; it is dependent (+D) – it is externally dependent on a municipality.

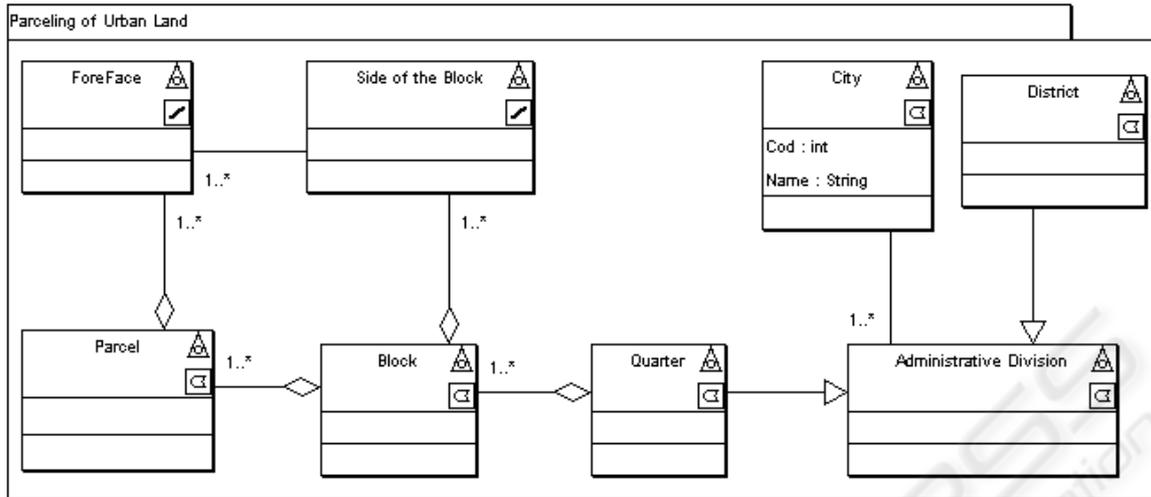


Figure 1: Parceling of Urban Land Analysis Pattern (Lisboa Filho et al. 2002).

The DISTRICT class is defined as an administrative unit of a municipality. It is classified as: Anti-Rigid (~R) – a district can emancipate itself, therefore, it will not be like that for all its existence; it does not give identity (-O) – the existence of the administrative division of a municipality identifies the districts; it executes Identity (+I) – it executes an identity inherited from the ADMINISTRATIVE DIVISION class; it is Dependent (+D) – it is externally dependent on a municipality.

The QUARTER class stores information about the intra-urban regions of a city or village, and through a municipal law, can become an administrative unit (IBGE 2008). It is classified as: Anti-Rigid (~R) – every instance of quarter it will not be like that for all its existence; it does not give identity (-O) – the quarter is identified from the existence of an intra-urban region; it executes identity (+I) - it executes identity inherited from the definition of an intra urban region it is non-dependent (-D) – it does not depend on any other class.

The BLOCK class stores information about urbanized areas of a city. A block can be created through a municipal law (IBGE, 2008). It is classified as: Rigid (+R) – every instance of the BLOCK class it will be like that for all its existence; it gives identity (+O) – every block can be identified within the urban area, through its spatial localization; it executes identity (+I) – if it gives a condition of identity, it also executes it; it is dependent (+D) – it is externally dependent on a municipality.

The PARCEL class stores information about a portion of a parceled land, with its front to public

way and meant for receiving a construction. It is classified as: Rigid (+R) – every instance of parcel it will be like that for all its existence; it gives identity (+O) – every parcel can be identified within its urban area, during its existence, through its spatial localization; it executes identity (+I) – if it gives a condition of identity, it also executes it; it is not dependent (-D) – it does not depend on any other class.

The FOREFACE class stores information about the alignment of a parcel or group of parcels facing the same road. It is classified as: Rigid (+R) – all foreface of parcel will be like that for all its existence; it gives identity (+O) – every foreface can be identified within the urban area, through its spatial localization; it executes identity (+I) – if it gives a condition of identity, it also executes it; it is dependent (+D) – it is externally dependent on the PARCEL class.

The SIDE OF THE BLOCK class stores information about the composition of the Foreface of the parcels. It is classified as: Rigid (+R) - every instance of the SIDE OF THE BLOCK class will be like that for all its existence; it gives identity (+O) – every side of the block can be identified within the urban area, through its spatial localization; it executes identity (+I) – if it gives a condition of identity, it also executes it; it is dependent (+D) – it is externally dependent on the BLOCK class.

4.1 Adequacy to the Restrictions Imposed by Meta-Properties

Following the technique VERONTO (Villela et al. 2004), the CITY class must be a subclass of a rigid

Table 2: New description of the Parceling of Urban Land analysis pattern.

Problem	How to structure a database of urban cadastre?
Context	The aim is to permit the manipulation of geographic information to support the public administrator. Those databases are used by many applications as distribution of health stations, taxes charging, and others. This kind of application needs information about the geographic regions within a municipality.
Forces	It is not financially easy to create a base with the aimed scale. The bigger the scale the bigger the cost to the creation and maintenance of data. A lot of kinds of regions are used depending on the size and organization of the city. The most common kinds include quarters, administrative divisions or urban zones.
Solution	Parceling_of Urban_Land.xmi (Figure 2)
Participants	The Municipality class has mandatory a seat, or urban area of city, and can have administrative units. An administrative unit can be a district or sub district. The quarter class is related to the Block class, through a multiplicity of one-to-many. But this association must be adapted to each specific situation, according to the municipal law.
Related Pattern	Circulation network
Rigid Classes	Municipality, Administrative Division, Municipal Seat, District Seat, Intra-urban Region, Parcel, Foreface, Block, Side of the Block

5 FINAL CONSIDERATIONS

The process of improvement of analysis patterns shown in this article permits to restructure the pattern's class diagram from an ontological analysis. The pattern catalog according to its domain and rigid classes, it is esteemed that it will be possible to reuse it more widely with a well-defined purpose.

With the improvement process and the analysis patterns catalog through ontological meta-properties, it is possible to integrate them with domain ontologies. Through this integration, it is possible to create mechanisms to recover analysis patterns through the existing knowledge in the ontology, increasing the quality and productivity of conceptual modeling of information systems.

ACKNOWLEDGEMENTS

This project was partially supported by the Brazilian National Research Council (CNPq), MCT/CTINFO and Fapemig.

REFERENCES

Devedzic, V. (1999) Ontologies: Borrowing from Software Patterns. Intelligence.
 Fernandez, E. B. (1998) Building systems using analysis patterns. Procs. 3rd Int. Soft. Architecture Workshop (ISAW3), Orlando, FL, November 1998. p. 37-40.
 Fowler, M. (1997) Analysis Patterns: Reusable Object Models. Menlo Park, CA: Addison Wesley Longman.

Guarino, N (1995) Formal Ontology, Conceptual Analysis and Knowledge and Representation. *Internacional Journal of Human and Computer Studies*, 43(5/6).
 Guarino, N., Welty, C. (2000) A formal ontology of properties. In R. Dieng, Ed., Proc. 12th Int. Conf. on Knowledge Engineering and Knowledge Management, Springer Verlag.
 Guizzardi, G., Falbo R.A., Pereira Filho, J.G. (2002) Using Objects and Patterns to Implement Domain Ontologies, *Journal of the Brazilian Computer Society*, vol. 1, no. 8.
 Hamza, H (2004) Improving analysis patterns reuse: an ontological approach. Proc. of Ontologies as Software Engineering Artifacts Workshop, OOPSLA'04.
 IBGE-Instituto Brasileiro de Geografia e Estatística (2008) Censo Demográfico. Available in: <http://www.ibge.gov.br>. Access in Jan, 29th 2008. (in Portuguese)
 Lisboa Filho, J., Iochpe, C., Borges, K. A. V. (2002) Analysis patterns for GIS data schema reuse on urban management applications. *CLEI Electronic Journal*, v.5, n.2.
 Lisboa Filho, J., Sodré, V.F., Daltio, J., Rodrigues, M.F., Vilela, V. (2004) A CASE tool for geographic database design supporting analysis patterns. Proc. Conceptual Modeling for GIS, ER2004. LNCS 3289.
 Lisboa Filho, J., Iochpe, C. (2008) Modeling with a UML profile. In: Shashi Shekhar and Hui Xiong. *Encyclopedia of GIS*. Germany: Springer-Verlag.
 Meszaros, G.; Doble, J. (1998) A pattern language for pattern writing.
 Villela, M. L. B., Oliveira, A. P., Braga, J. L. (2004) Modelagem ontológica no apoio à modelagem conceitual. XVIII Simpósio Brasileiro de Engenharia de Software. Brasília. (in Portuguese)
 Wand, Y, Storey, V. C., Weber, R. (1999) An ontological analysis of the relationship construct in modeling conceptual. *ACM Transactions on Database Systems*, 24(4): 494-528.