

A UML Profile for Conceptual Modeling in GIS Domain

Jugurta Lisboa-Filho¹, Gustavo Breder Sampaio¹,
Filipe Ribeiro Nalon¹ and Karla A. de V. Borges²

¹Departamento de Informática, Universidade Federal de Viçosa
36570-000 - Viçosa, MG, Brazil
jugurta@ufv.br, gustavobreder@gmail.com, frnalon@gmail.com

²Prodabel – Empresa de Informática e Informação do Município de Belo Horizonte
Av. Pres. Carlos Luz, 1275 – 31230-000 – Belo Horizonte – MG - Brazil
karla@pbh.gov.br

Abstract. After many years of research in the field of conceptual modeling of geographic databases, experts have produced different alternatives of conceptual models. However, still today, there is no consensus on which is the most suitable one for modeling applications of geographic data, which brings up a number of problems for field advancement. A UML Profile allows a structured and precise UML extension, being an excellent solution to standardize domain-specific modeling, as it uses the entire UML infrastructure. This article proposes an UML profile developed specifically for conceptual modeling of geographic databases called GeoProfile. This is not a definite proposal; we view this work as the first step towards the unification of the various existing models, aiming primarily at semantic interoperability.

Keywords: UML profile, GIS, Conceptual data model, Geographic database.

1 Introduction

One of the current concerns in software development is to better understand the domain of the problem, about which it is intended to create solutions that meet satisfactorily the real needs of users. To aid in this task, one of the techniques used is the conceptual modeling, which consists in to extract from the real world only those essential elements observed, leaving out implementation aspects.

The process of conceptual modeling allows a better understanding of the system being designed and is performed with the aid of specific modeling languages, which are languages whose syntax and semantics are focused toward the conceptual representation of a system [3]. The Unified Modeling Language (UML) has been widely used and accepted by the scientific community and industry, as a tool for design and specification of systems [18].

One area that has currently received much attention includes the geographic applications domain, given its wide range of usefulness to society and the scientific community and whose systems have particular characteristics that need to be taken into account in developing such applications.

Parent et al [20] emphasize that the conceptual modeling has several advantages for the design of geographic applications. It allows, for instance, users to express their knowledge on the application using concepts that are closer to them, without the need to use computational expressions.

For the past 20 years, several research groups have been studying the requirements for database conceptual modeling of Geographic Information Systems (GIS) [1]. Some conceptual models specific to this area were proposed. OMT-G [4], MADS [20], GeoOOA [14], UML-GeoFrame [15] and the Perceptory's model [2] are important among these models.

Despite the maturity of this research field, to date, there is no consensus among designers and users as to which model best meets the requirements for modeling a geographic database (geoDB). The lack of a standard model brings up serious problems in the development of the field, as for instance, communication difficulties among different projects. For example, considering CASE tools that support conceptual models specific to geoDB, data conceptual schemas cannot be migrated between different tools, as it happens with conventional database designs.

These problems would not exist if there was a standard for modeling such applications that incorporated the main features of the existing models. The creation of a UML profile is one option to standardize this type of models. UML profile is a feature that allows for a structured and precise extension of the UML elements so that it can fit into a specific domain [12].

This paper aimed to initiate the specification of a UML profile for the conceptual modeling of geoDB taking into account the requirements imposed on this application domain. Some models in the literature provided the basis for this task. The remaining of the paper is structured as follows. Section 2 presents the concept of UML profile. Section 3 describes the requirements of geoDB conceptual modeling, as well as the main current models, while Section 4 details the proposal to the GeoProfile and usage examples. Section 5 presents the final considerations and future work.

2 UML Profiles

Despite being a general purpose language, which can be used in different application domains, there are situations in which the UML elements are not able to express all the peculiarities of a given domain. Therefore, to prevent the UML became too complex, it was specified as an extensible language [10].

The OMG defines two ways of extending the UML. The first is based on the modification of the UML metamodel, thereby creating a new language, in which the syntax and semantics of the new elements are adapted to the intended domain. The second way is to adapt the UML to specific domains or platforms using the mechanism of profiles. In this second alternative, the elements of language are specialized, but respecting the UML metamodel and maintaining the original semantics of the elements unchanged [12].

In this first form of UML extension, the new language is created using MOF. In the second alternative, the language elements will be specialized by using the extension mechanisms provided by UML, which are:

- **Stereotypes.** A stereotype defines how an existing metaclass may be extended and enables the use of specific terminology for a domain or different platform in place of or in addition to the terminology used for the extended metaclass. Stereotypes can also change the appearance of the elements of the extended model using graphic icons;
- **Tagged values.** They are additional meta-attributes associated with a metaclass of the metamodel extended by a profile and add information to elements of the model;
- **Constraints.** These are restrictions associated with the corresponding elements of the metamodel. They can be written using natural language or OCL, which is also standardized by the OMG.

A UML profile is a set of extension mechanisms grouped in an UML package stereotyped as <<profile>>. As mentioned earlier, these mechanisms allow the extension of the syntax and semantics of the UML elements, but without violating the original semantics of UML and, therefore, consistent with MOF.

The idea of extending the UML for specific purposes is not new. UML 1.1 could already easily assign stereotypes and tagged values to model elements. However, the notion of profile was defined to provide a more structured and precise extension [18]. UML profile is already adopted as a standard modeling in some domains, such as CORBA architecture [19]. Other profiles are in the process of being adopted by the OMG or are being created by private organizations, software companies and research centers.

OMG [18] emphasized that there is no simple answer to the question of when to create a new metamodel or when to use the mechanism of profiles. Each alternative has its advantages and disadvantages, but the use of UML profiles provides a better cost-benefit ratio, by utilizing the entire structure of the UML tools and training materials. Fuentes and Vallecillo [12] mention that the benefits of using UML profiles undoubtedly exceed their limitations.

A UML Profile allows a structured and precise extension of UML constructors to customize UML for a particular domain. A well-specified UML Profile will have direct support of CASE tools. In other words, once the Profile is defined there is no need to implement new CASE tools. *Enterprise Architect* [9] and *Rational Software Modeler* [21] are examples of CASE tools with support for UML Profiles.

Hence, the development of a UML Profile has proven an excellent method to standardize modeling of specific domains, as it uses the language's popularity and tools compatible with UML 2.0, favoring standard acceptance and reducing time for training in new languages.

3 Conceptual Modeling of Geographic Database

The term Geographic Information Systems (GIS) is applied to systems that perform a computational analysis of geographic data. The main difference between GIS and a conventional information system is the ability of GIS to store both the descriptive attributes and the geometries of different types of spatial data [24].

GIS use has grown and continues to grow rapidly throughout the world due to advances in hardware and software and the increasingly easier access to these technologies. Worboys [24] points out that among the main components of a GIS is the storage component, which is called geographic database. Its function is to

structure and store data in order to enable carrying out the analysis with spatial data. Applications developed with GIS are highly complex and a major problem in developing these applications has been designing the geoDB [16].

The classical approach to project database is to divide the process into three stages: conceptual design, logical design and physical design [8]. In conceptual design, the conceptual database is drawn up on the basis of conceptual models that provide high-level abstraction builders to describe the requirements for application data.

One of the principles of conceptual modeling is that a conceptual schema should only contain the elements of the domain, discarding implementation aspects. The process of database conceptual modeling includes a description and definition of possible contents of data, as well as structures and rules that apply to them [15]. In the case of geoDB, the specific nature of geographic information led to the development of specific solutions for modeling spatial data.

Friis-Christensen et al [11] describe a survey of requirements for modeling spatial data. These requirements are classified into five groups, as follows:

- **Spatiotemporal properties.** Include the spatial requirements (coordinates in a reference system, representation of points, lines and polygons), time (need to record the existence time and the changes undergone by an object); need for representation of object attributes, and a unique identifier; and difference between fields (the real world is perceived as a set of space-varying attributes as a continuous function) and objects (the real world consists of entities with unique identity);
- **Roles.** A same geographic object can be defined in different ways depending on the universe of discourse. That is, the role of an object is dependent on the application. It should be possible the indication of roles based on the same type of object;
- **Associations.** Include topological relationships (e.g., overlap, touch), metric (involving distance and depending on the absolute position of objects in a reference system), semantic (e.g. “all lots must have access to roads”), and relationships to indicate that an object is composed of other objects;
- **Constraints.** It should be possible to attach constraints to objects (e.g., limiting the value of an attribute to a certain range) and associations (e.g., preventing a building from being located on a lake). Constraints are related to data quality, which is negatively affected when constraints are not met.
- **Data quality.** This information is important in order to know the source credibility and data accuracy. It should be compared with the application specifications to determine whether the data is accurate enough at that time.

Another list of requirements is shown in [17]. This study mentions eight groups of requirements, five of which are equivalent to those presented by Friis-Christensen et al [11]: possibility of modeling phenomena in the field and object view, spatial aspects, spatial relationships, temporal aspects, and quality aspects. The other requirements, not explicitly mentioned in the previous work, are: possibility of differentiating between geographical phenomena and objects without spatial reference; the need to organize the phenomena by theme; and the possibility of modeling phenomena with more than one spatial representation (multiple representations).

Friis-Christensen et al [11], compare some models with these requirements to show advantages and disadvantages of each model. One of the conclusions of this

study shows the importance of balancing ease-of-use of the model notation with its comprehensiveness. The posed challenge is to balance these two characteristics or improve them, and the development of a standard model provides the basis for data exchange.

The profile proposed in this paper is based on contributions from a number of models existing in the literature, as well as the concepts defined in Goodchild [13]. The models that have contributed most significantly to the GeoProfile development are cited below, but certainly other predecessor models also had their contribution.

The OMT-G (Object Modeling Technique for Geographic Applications) model [4] has a rich collection of conceptual constructors, the strong point of which is modeling spatial relationships, including spatial aggregation. The GeoOOA model [14] supports the abstraction of spatial classes, whole-part topological structures, network structures and temporal classes. MADS (Modeling of Application Data with Spatio-temporal Features) [20] approaches objects and relationships in its diagram, with structures very similar to the Entity-Relationship model. Its main feature is the orthogonality, in which spatial and temporal characteristics can be added either to objects or attributes or relationships. The Perceptory's model was the pioneer in the use of pictograms. These pictograms are grouped into the languages Spatial PVL and Temporal PVL (Plug-in for Visual Languages), which allow the addition of spatial-temporal characteristics not only to UML, but also to other visual modeling languages. The UML -GeoFrame model is based on a structured hierarchy of classes that make up the GeoFrame, providing the basic elements present in any geographic database [15]. The proposal of ISO-191xx Standard [6] differs from the models above mentioned for addressing more the logical level (records) than the conceptual level (abstractions).

Finally, Clementini et al [7] formally describe a small set of relationships capable of reproducing all the possible topological relationships that can occur between spatial elements with the representation of point, line or area. Although not proposing a model, this work has considerable importance in the scope of the GeoProfile design. Defining a minimum set of relationships, one eliminates the possible use of two relationships with different names, but having the same meaning. This set includes the following relationships: *touch*, *in*, *cross*, *overlap* and *disjoint*.

4 GeoProfile

GeoProfile is a UML profile built for the conceptual modeling of geographic databases. According to the proposed methods to guide the construction of a UML Profile (Section 2), two artifacts are generated during profile development: the domain metamodel and the profile itself. While the first is useful to understand the addressed problem, the second presents the extensions received by the UML metaclasses.

In order to check the validity of the GeoProfile specification, this profile has been implemented in RSM [21]. Mechanisms for creation of stereotypes were successfully tested, as well as automatic validation of schemas by checking OCL constraints.

Section 4.1 defines a metamodel for the geographical domain. Section 4.2 proposes a set of stereotypes for the proposed profile. Section 4.3 shows a way to specify additional integrity constraints. Section 4.4 shows the implementation of the GeoProfile in a CASE tool, and Section 4.5 presents examples of GeoProfile use.

4.1 Defining a metamodel for geographical domain

At the beginning of the metamodel specification, elements are identified in a conceptual schema, observing the requirements of this type of conceptual modeling.

The way each considered conceptual model in this proposal (GeoOOA, MADS, UML-GeoFrame, OMT-G and Perceptory's model) meets the found requirements was examined. The inclusion of the main mechanisms present in each of these models into the GeoProfile allows it to meet most requirements of a geoDB. Table 1 summarizes the results obtained in the comparative analysis between requirements and conceptual models, but also displays in its last column the models that most influenced GeoProfile construction in each requirement.

Among the discussed conceptual models, the UML-GeoFrame shows the closest organization to a metamodel. GeoFrame is defined in a class hierarchy representing the elements present in a geoDB. Thus, the metamodel development started from a GeoFrame adaptation (Figure 1).

A geoDB comprises a number of themes, which is characterized by the metaclass *Theme*. A theme can be formed by the aggregation of other themes or objects with or without spatial representation, characterized by the classes *GeoPhenomenon* and *ConventionalObj* respectively.

When one chooses to associate a spatial representation with objects of a class, it is possible that the phenomenon is perceived in the geographic field view (*GeoField*) or object view (*GeoObject*). Depending on the technique used in geographic information acquisition in the field, its representation be selected from six options as described in [13]: *AdjPolygons*, *Isolines*, *TIN*, *GridOfPoints*, *GridOfCells* or *IrregularPoints*. Representation of geographic objects can be of the types point, line, polygon or complex (the object geometry consists of other geometries).

To specify multiple representations, it is possible to use more than one stereotype in the same class of the conceptual schema, as in the Perceptory's model.

Table 1. Comparison between requirements and models presented, and major contributions to the GeoProfile.

Models X Requirements	GeoOOA	MADS	OMT-G	Perceptory	UML-GeoFrame	Contribution for GeoProfile
Geographical phenomena and conventional objects	Yes	Yes	Yes	Yes	Yes	Perceptory
Field visions and objects	Partial	Partial	Yes	No	Yes	OMT-G
Spatial aspects	Partial	Yes	Yes	Yes	Yes	OMT-G, UML-GeoFrame
Thematic aspects	No	No	Yes	Yes	Yes	UML-GeoFrame
Multiple representations	Partial	Yes	Yes	Yes	Yes	UML-GeoFrame
Spatial relationships	Partial	Yes	Yes	Partial	Partial	MADS, OMT-G
Temporal aspects	Partial	Yes	No	Yes	Partial	MADS, Perceptory

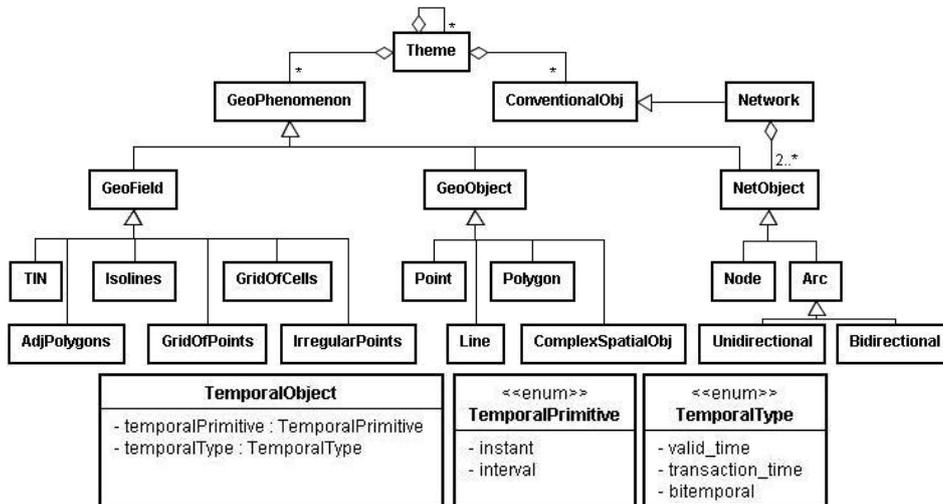


Fig. 1. Metamodel for the geographical domain

The requirements related to the roles and metadata are not considered in the GeoProfile proposal. Despite representing important information relating to spatial data, it is believed that they need not necessarily be demonstrated during the conceptual modeling of a geoDB.

Topological and composition are the main types of spatial relationships to be represented in a conceptual schema. There was no need to add new constructors to the GeoProfile to characterize composition, as the UML can indicate whether an association is a composition or aggregation. However, it was necessary to add new constructors to model topological relationships, including the capacity to represent networks.

With basis on GeoOOA and OMT-G models, which provide more detailed solutions for network representation, [23] proposed an extension of GeoFrame to address the requirement. This extension was incorporated into the metamodel.

The classes in charge of storing alphanumeric data and information on which elements participate in the network are represented by the metaclass *Network*. Since this metaclass does not have spatial information, it was defined as a *ConventionalObj* specialization. The networks are formed by network objects (*NetObject*), which can be nodes (*Node*), unidirectional arcs (*Unidirectional*) or bidirectional arcs (*Bidirectional*).

The other types of topological relationships are directly defined in the creation of stereotypes and OCL constraints. This is because a large number of possible relationships between spatial objects of the type point, line and polygon would overburden the metamodel.

The MADS and Perceptory approaches stand out among temporal aspects. Although they do not consider transaction time, icons added at different positions of the class diagram can indicate that the object's existence time, its spatial evolution or the evolution of values of certain attributes in that class should be kept in the database. Despite being an interesting solution, it can visually overload the schema. Another solution adopted by GeoProfile is indicated only whether a class is considered temporary or not, as in the GeoOOA model. In this case, it is implied that

both the attributes and spatial data of an object can vary, and these changes must be maintained in the database.

In this way, the metaclass *TemporalObject* was added to the metamodel. This metaclass has two attributes that characterize temporal information. One of these attributes indicates the temporal type (validity time, transaction time or bitemporal time), whereas the other defines the used temporal primitive type (instant or interval). There are two enumerations (*TemporalType* and *TemporalPrimitive*) for the possible values these attributes can assume.

4.2 GeoProfile stereotypes

After creating the domain metamodel, the next step is to extend the UML metaclasses to create the profile itself. Figure 2 illustrates the stereotypes of GeoProfile, generated from the metamodel shown in Figure 1.

The UML allows the definition of graphic and textual («...») stereotypes. The authors believe that the choice of graphic stereotypes is a matter of personal taste (or customary within an organization) and there is no need of standardization. For example, two designers, one familiar with the MADS model and the other with the notation used in the Perceptory tool, might start to use GeoProfile, but keep the original graphical representation of the stereotypes of their preferred model. CASE tools that support profile may allow different graphical views of the same data schema, enhancing conceptual interoperability. Thus, initially, it was decided not to propose graphic stereotypes for GeoProfile, leaving the standardization to future decision.

It is worth noting that not all metaclasses of the domain metamodel have a corresponding stereotype, as it happens with *Theme* and *ConventionalObj*. Themes can be represented by packages. Classes of conventional objects are, however, modeled by UML classes without addition of stereotypes. Therefore, the UML constructors themselves can reproduce these two concepts.

Another important observation is that some stereotypes are abstract (*GeoObject*, *GeoField*, *NetObject* and *Arc*). During GeoProfile use, these stereotypes are not available to be used. They are, nevertheless, useful for organizing profile elements, allowing addition of constraints common to all the other stereotypes created as their specialization. For example, a constraint that is common to the stereotypes *UnidirectionalArc* and *BidirectionalArc* can be added to *Arc*.

Geographic phenomena, extending the metaclass *Class*, are defined in a similar hierarchy to that found in the domain metamodel. The stereotype *Network* directly extends the metaclass *Class*, since there is no stereotype defined for representation of conventional objects.

To deal with temporal aspects, the stereotype *TemporalObject* was added to GeoProfile, as well as two enumerations (*TemporalPrimitive* and *TemporalType*). In addition, designers are allowed to indicate that an association between two objects is only valid for one period and this history should be kept in the database. This is done by simply assigning the stereotype *Temporal*, which extends the metaclass *Association* to an association of the schema.

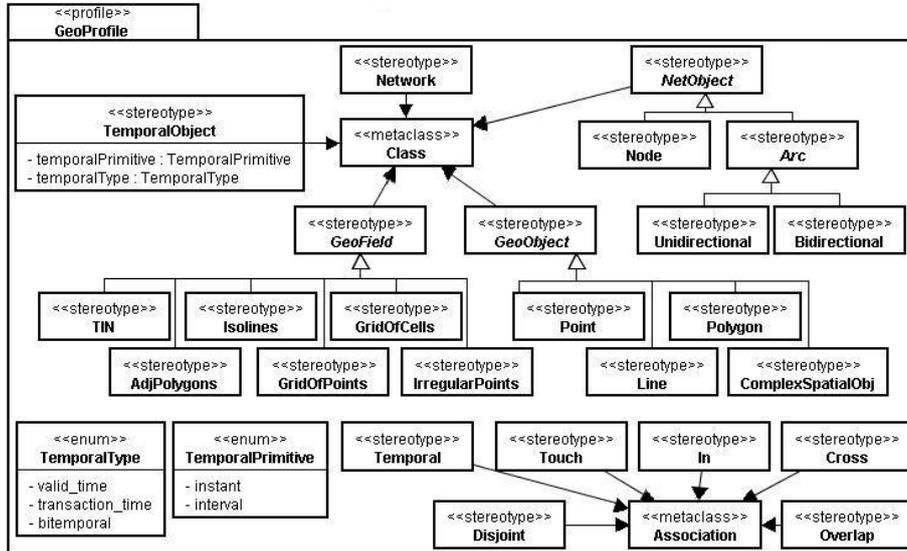


Fig. 2. GeoProfile Stereotypes

Finally, stereotypes were created to represent the topological relationships that were not considered during drawing up of metamodel. We chose to use the set of five relationships proposed by [5], as they are capable of representing any topological relationship between objects of type point, line or polygon. Thus, the stereotypes *Touch*, *In*, *Cross*, *Overlap* and *Disjoint*, all extending the metaclass *Association*, were added.

4.3 OCL constraints

The constraints included in the GeoProfile focus on the validation of the designer's conceptual schema. Consequently, they always have a stereotype of the GeoProfile as context, as well as being invariants.

Those constraints basically prevent the occurrence of three error types: addition of incompatible stereotypes with a same element, poor network construction and addition of impossible topological relationships between two elements (e.g. *Cross* relationship between two geographic objects with point representation). These three constraints groups were analyzed and a set of OCL expressions was specified. There is no limitation to the inclusion of the stereotype «TemporalObject» in classes of the schema or «Temporal» in their associations. Because of space limitation, this article describes only one of the OCL constraints as example.

The constraint (a) evaluates the use of incompatible stereotypes. Each class that receives a stereotype of geographic field (*context GeographicField*) must have all its applied stereotypes captured (*getAppliedStereotypes*). Stereotypes of the geographic object type are selected from the result using the *select* method, and the returned set must be empty (*isEmpty*), since a class cannot have object and field representation at the same time.

```

context GeographicField
inv: self.getAppliedStereotypes() -> select(s |
s.name = 'Point' or s.name = 'Line' or s.name =
'Polygon' or s.name = 'ComplexSpatialObj') ->
isEmpty()

```

4.4 Implementation of GeoProfile in a CASE tool

One of the greatest advantages in using a UML profile as a basis for modeling of a specific field is to use the entire UML infrastructure. Therefore, an implementation of this profile in the RSM [21] was carried out to verify the validity of the GeoProfile specification. Mechanisms for stereotype creation were successfully tested, as well as automatic validation schemas from the verification of OCL constraints. OCL constraints assist the designer in identifying basic errors.

RSM is produced by IBM® and supports UML 2.1. This work used the version 7.0.5. The tool interface can be changed according to user's preferences. Figure 3 illustrates the RSM interface with support for GeoProfile.

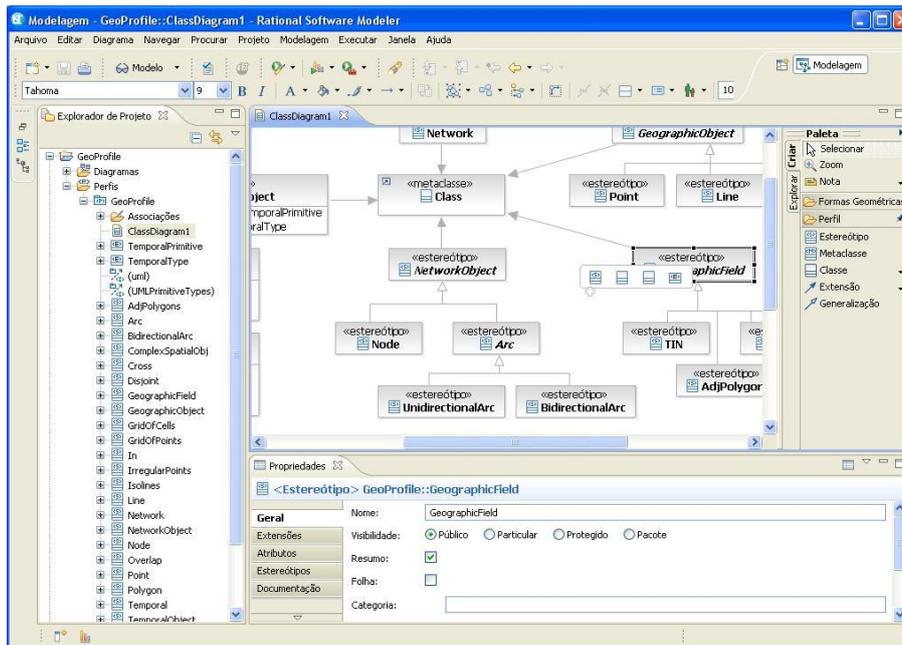


Fig. 3. RSM interface with GeoProfile

4.5 Usage examples of GeoProfile

Considering the results obtained after the establishment of the GeoProfile and its implementation in the CASE tool RSM, this section presents examples of conceptual modeling using GeoProfile. To allow a comparison between the GeoProfile and conceptual models that were the basis for its definition, each example also shows the corresponding conceptual schema in the other model.

Figure 4 illustrates part of the conceptual modeling of a system for pollution control in parcel (or plot) of land. The diagrams (a) and (b) display the model developed using the model GeoOOA and GeoProfile, respectively. The parcels have a polygonal representation. A non-geographic object providing information on the owners of each parcel must be stored. In addition, each parcel may contain several pollution controls, which are geographically represented by points. In the association between parcels and points of pollution control, the restriction that each control point must be contained in the area of the parcel with which it is associated is represented in the conceptual schema.

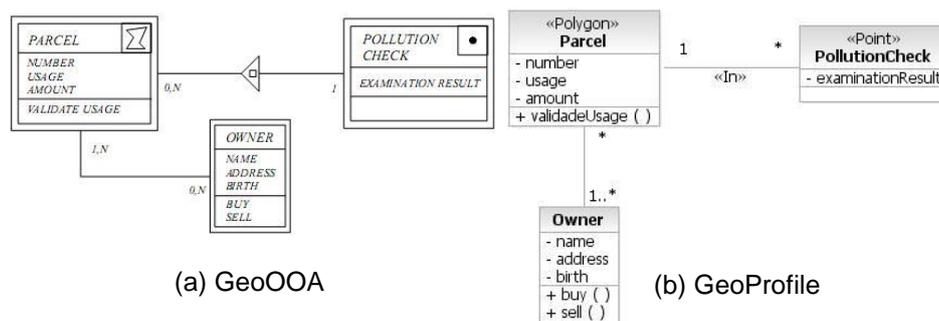


Fig. 4. Comparison between GeoOOA and GeoProfile (Source: (4-a) [14])

Another example of topological relationship involving parcels (or plots) of land is showed in Figure 5, which compares (a) the MADS model with (b) GeoProfile. In this schema each plot may contain several buildings, and both classes have polygonal representation. Furthermore, a restriction is imposed that the buildings belonging to a particular plot must have their geographical area within the area of the plot. In the case of the topological relationship «In», it may be important for a correct interpretation of the schema, to state which of the objects involved in a particular association must have its geometry contained in the geometry of the other object that participates in the association. In Figure 5-b, roles of the association were used for this purpose. Another option is to indicate the navigability of the association. Both solutions use resources of the UML specification.

Finally, the last example explores temporal aspects. In Figure 6-a, a class House is modeled using the Perceptory CASE tool [2]. The temporal pictogram located on the top right of the diagram of this class shows that the period in which the house exists in modeled reality (e.g., date of construction until the date of demolition) should be stored in the database. However, when the same pictogram is added to the side of the spatial representation pictogram or next to an attribute, it indicates that the historical of the object spatial evolution or the evolution of the values of an attribute, respectively, must be kept into the database. As discussed above, in GeoProfile, these

concepts are grouped into just one stereotype called «TemporalObject». In this case, it is implied that both the period of existence and the historical evolution of the attributes or geometry of an object must be kept in the database.

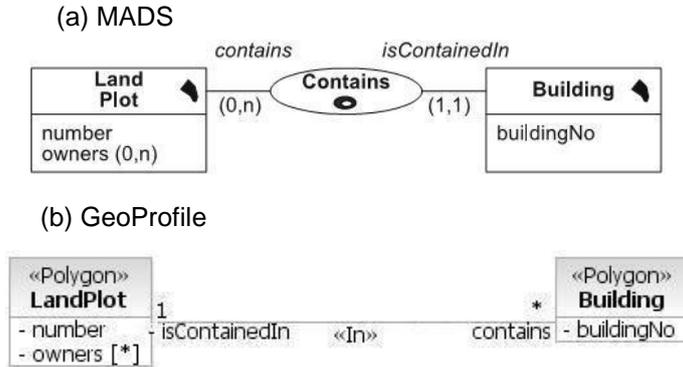


Fig. 5. Comparison between MADS and GeoProfile (Source: (5-a) adapted from [20])

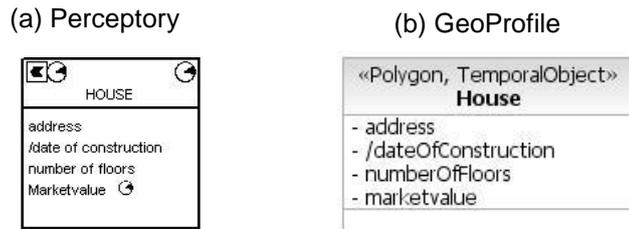


Fig. 6. Comparison between Perceptory and GeoProfile (Source: (6-a) [2])

5. Final Considerations

The idea of this paper is not to propose one more new conceptual model for GIS, but rather to propose a set of constructors, extracted from existing models toward a standard geographic profile for database modeling in GIS domain.

The existence of several alternative conceptual models of geographical databases prevents users and designers to migrate their projects from a CASE tool to another. Another major problem brought up by the lack of standardization is the difficulty in training designers, since although the models have been produced for the same purpose; each one has its differences and particularities. Users who are familiar with a model and its respective CASE tool (e.g. Perceptory [2] and ArgoCASEGEO) show strong resistance to accept a new one.

The use of a UML profile will solve these problems. Besides the wide UML acceptance by software developers, the availability of CASE tools with support for profiles rule out the need for implementing specific tools for a particular model.

A subject for future work is the logical-conceptual transformation of schemas produced with GeoProfile. The existence of logical standards, as defined by OGC and

the series ISO 191xx [6], will have a strong link with the level of conceptual modeling. Finally, the great challenge is to make authors of the existing conceptual models contribute to improve the GeoProfile. Moreover, to know the opinion of the users is important, because in many cases the database of a GIS application is designed by them. Thus, it is also important to measure the GeoProfile use's facility and its learning curve.

Acknowledgments. This project was partially funded by CAPES, FAPEMIG and CNPq / MCT / CT-Info.

References

1. Bédard, Y., Larrivée, S., Proulx, M., Nadeau, M.: Modeling geospatial databases with plug-ins for visual languages: a pragmatic approach and the impacts of 16 years of research and experimentations on Perceptory. In: CoMoGIS 2004. LNCS. vol. 3289, pp. 1148-1158. Springer (2004)
2. Bédard, Y.: Visual modeling of spatial databases: towards spatial PVL and UML. *Geomatica*, 53(2), 169--186 (1999)
3. Booch, G., Rumbaugh, J., Jacobson, I.: *The Unified Modeling Language user guide*. 2. ed. Addison-Wesley, Boston (2005)
4. Borges, K.A.V., Davis Jr., C.A., Laender, A.H.F.: OMT-G: an object-oriented data model for geographic applications. *GeoInformatica*, 5(3), 221--260 (2001)
5. Brodeur, J., Bédard, Y., Proulx, M.-J.: Modeling geospatial application database using UML-based repositories aligned with International Standards in geomatics. In: ACMGIS, Washington DC (2000)
6. Brodeur, J., Badard, B.: Modeling with ISO 191xx standard. In: Shekhar, S.; Xiong, H. (Eds.). *Encyclopedia of GIS*. Springer-Verlag, pp. 691--700 (2008)
7. Clementini, E., Di Felice, P., Oosterom, P.: A small set of formal topological relationships suitable for end-user interaction. In: *Int. Symp. on Advances in Spatial Databases*. Springer-Verlag, London. pp. 277-295 (1993)
8. Elmasri, R., Navathe, S.B.: *Fundamentals of database systems*. 4 ed. Addison-Wesley, Boston (2003)
9. Enterprise Architect. <http://www.sparxsystems.com/products/ea/>
10. Erikson, H., Penker, M., Lyons, B., Fado, D.: *UML 2 Toolkit*. OMG Press, Indianapolis (2004)
11. Friis-Christensen, A., Tryfona, N., Jensen, C.S.: Requirements and research issues in geographic data modeling. In: *ACM Int. Symp. on Advances in GIS*, Atlanta, pp. 2--8 (1993)
12. Fuentes, L., Vallecillo, A.: An introduction to UML profiles. *UPGRADE, The European Journal for the Informatics Professional*, 5(2), 6--13 (2004)
13. Goodchild, M. F., Yuan, M., Cova, T. J.: Towards a general theory of geographic representation in GIS. *Int. Journal of Geographic Information Science*, 21(3), 239--260 (2007)
14. Kösters, G., Pagel, B., Six, H.: GIS-Application development with GeoOOA. *Int. Journal of Geographical Information Science*, 11(4), 307--335 (1997)
15. Lisboa Filho, J., Iochpe, C.: Modeling with a UML Profile. In: Shekhar, S.; Xiong, H. (Eds.). *Encyclopedia of GIS*. Springer-Verlag, pp. 691--700 (2008)
16. Lisboa Filho, J. et. al.: A CASE tool for geographic database design supporting analysis patterns. In: CoMoGIS/ER 2004, LNCS vol. 3289, Springer (2004).
17. Lisboa Filho, J.; Iochpe, C.: A study about data conceptual models for geographic database design. *Informática Pública*. 1(2), 67-90 (1999) (In Portuguese)

- 18.Object Management Group. Unified Modeling Language: Infrastructure. V. 2.1.2 (2007)
- 19.Object Management Group. Profile Catalog. http://www.omg.org/technology/documents/profile_catalog.htm
- 20.Parent, C., Spaccapietra, S., Zimányi, E.: Modeling and multiple perceptions. In: Shekhar, S.; Xiong, H. (Eds.). Encyclopedia of GIS. Springer-Verlag, pp.682--690 (2008)
- 21.Rational Software Modeler. <http://www-01.ibm.com/software/awdtools/modeler/>
- 22.Selic, B.: A systematic approach to domain-specific language design using UML. In: 10th IEEE Int. Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC'07), pp. 2--9 (2007)
- 23.Stempliuć, S. M., Lisboa F., J., Andrade, M. V. A., Borges, K. V. A.: Extending the UML-GeoFrame data model for conceptual modeling of network applications. In: Int. Conf. on Enterprise Information Systems (ICEIS), Milão pp. 164--170 (2009)
- 24.Worboys, M., Duckhan, M.: GIS: A computing perspective. 2 ed. CRC Press, Boca Raton (2004)