DESIGN AND IMPLEMENTATION OF THE VALID TIME FOR SPATIO-TEMPORAL DATABASES

Jugurta Lisboa Filho, Gustavo Breder Sampaio, Evaldo de Oliveira da Silva and Alexandre Gazola
Universidade Federal de Viçosa, Departamento de Informática, 36570-000, Viçosa, MG, Brazil

Keywords: Geographic database, GIS, Spatio-Temporal Data Modeling.

Abstract: Three different types of time are identified in the literature on Temporal Database Management Systems: valid time, transaction time and existence time. This paper describes the design of the valid time for Spatio-Temporal Databases in Geographic Information Systems, based on the UML-GeoFrame conceptual data model. It is also presented two translation rules of the valid time from the conceptual to logical level, implemented for the TerraLib Spatial Components Library.

1 INTRODUCTION

A number of government and business organizations need to handle data that vary simultaneously with time and space, which is most of the time carried out using Geographical Information Systems (GIS). In some government agencies as well as the departments linked to mapping and monitoring of environmental risks, urban planning, residential and industrial zoning, of epidemic control, among others, the time factor is a fundamental element.

These agencies use GIS to execute operations of spatial analysis and make decisions based on results of these analyses, however, a great number of times the temporal aspect of the used information is not appropriately considered. This comes from the great difficulty GIS designers and users find when designing a spatio-temporal database and developing applications that allow the representation of phenomena that vary with time and space.

There are several conceptual data models for geographical database design, some with support for modeling temporal aspects. MADS (Parent. et al., 1999), Geo-ER (Tryfona. et al., 2003) and Perceptory (Bédard, 1999) are examples of these models and tools. These conceptual data models seek to consider all the possible variations of spatio-temporal data, making them highly complex models and of little practical use for most geographical database designers.

Three different types of time are identified in the literature on Temporal Database Management Systems: valid time - instant or interval of time in which a phenomenon is valid; transaction time - time in which the datum is inserted into or eliminated from a given database; and existence time - time in which the object really exists in the reality.

The conceptual data model UML-GeoFrame extends the UML class diagram to geographical database design (Lisboa Filho and Iochpe, 1999).

Rocha et al. (2001) proposed an extension to the UML-GeoFrame model for modeling temporal aspects that comprises these three types of time. However, as for the other models previously mentioned, the enormous possibility of combinations among the geographic phenomenon properties regarding temporal features, become the understanding, modeling and implementation of all these types very complex.

This article presents a simplified approach for modeling and implementation of spatio-temporal phenomena based on the valid time, using the UML-GeoFrame conceptual data model. It also presents the incorporation of these aspects into a CASE tool called ArgoCASEGEO. This tool is capable of transforming conceptual spatio-temporal data schemas into logical data schemas for the main data models of commercial GIS (eg.: Shape File Format, Oracle Spatial) and also for the data model of TerraLib spatial library (Vinhas and Ferreira, 2005).

Lisboa Filho J., Breder Sampaio G., de Oliveira da Silva E. and Gazola A. (2007).
DESIGN AND IMPLEMENTATION OF THE VALID TIME FOR SPATIO-TEMPORAL DATABASES.
In *Proceedings of the Ninth International Conference on Enterprise Information Systems - ISAS*, pages 569-573
DOI: 10.5220/0002386405690573

2 THE COMCEPTUAL DATA MODEL UML-GEOFRAME

The conceptual modeling of geographical database based on the Unified Modeling Language (UML) and on the GeoFrame framework (Lisboa Filho and Iochpe, 1999) generates an easy-to-understand database schema, improving communication among designers and/or users.

GeoFrame is a conceptual framework that provides a basic class diagram to assist the designer on the first steps of the conceptual data modeling of a new GIS application. The mutual use of UML class diagram and GeoFrame allows the solution of most requirements of GIS application modeling. A conceptual schema of geographical data, based on the UML-GeoFrame model includes, for instance, the modeling of spatial aspects of the geographical information, in other words, the type of spatial representation of each phenomenon, and the difference between conventional objects and geographical objects/fields. The specification of these elements is based on the stereotype concept, which is an UML extension mechanism (Booch,, Jacobson and Rumbaugh, 1998).

As example, Figure 1 illustrates an UML-GeoFrame data schema, where the UML package represents a theme relative to the real world that is being modeled and all of them concern the same specific geographical area (eg.: a street). The theme Street Mesch comprises four classes of an urban cadastre application. In the theme some classes are geographicals objects, wich are modeled with stereotype of object view (<>).

The spatial representation of classes of geographical phenomena perceived in object view can be of a point <>, line <>, polygon <> or complex <>, in this case when it is formed by more than one spatial object. The geographical fields can, in turn, be represented by one of the six types of representation of attributes varying in space, described by Goodchild (1992), which are: Irregularly sampled points <>, Adjacent polygons <>, Contour lines <>, Triangular irregular network, or TIN, <>, Cell grids <> and/or Regularly sampled points <>. The existence of multiple representations is modeled through the combination of two or more stereotypes in a same class. For example, a class Municipality can have two abstraction forms of its spatial component (a point or an area), which is specified by the pair of stereotypes <>.

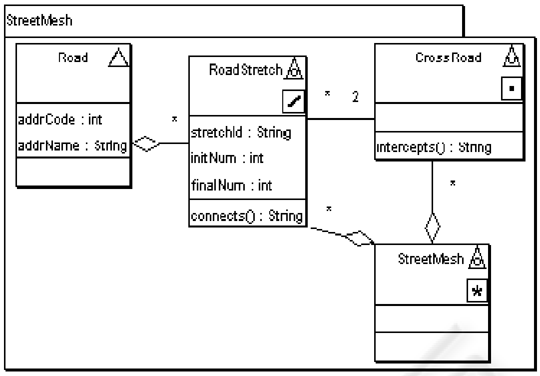

Figure 1: Example of UML-GeoFrame schema.

3 THE GEOFRAME-T PROPOSAL

GeoFrame-T is a GeoFrame temporal extension for modeling conventional, spatial, temporal and spatio-temporal objects (Rocha et. al., 2001) based on the TUML concept (Svinterikou, 1997). One of the main GeoFrame-T characteristics is to include the aspect time also for attributes and relationships.

The representation of temporal aspects in GeoFrame-T is carried out by the *TemporalObject* class, where there is an association of aggregation with the class *TimeObject*, which can have or not instances of the class *TemporalMetadata* associated with it. The *TemporalMetadata* class has attributes that describe the concepts of objects existing in the *TimeObject* class. These attributes qualify information on the reference systems based on time values such as the Greenwich meridian and local coordinate references. Other *TemporalMetadata* class attributes are granularity and calendar, which describe the time (month, year, day, second, etc.), and the type of calendar considered for the time. Rocha et al. (2001), defined the *TimeObject* class as a generalization of *Static* and *TemporalType* classes, which is specialized in *ValidTime*, *TransactionTime* and *Bitemporal*.

The *Static* class represents objects with no temporal variation associated. Through the class *TransactionTime,* it is possible conceptually to model objects that should have their historical information retained, whose time variation is linear, considering the time in which the information was entered into the database. The class *ValidTime* enables the temporal modeling of a spatial object, whose time is considered in accordance with the modeled reality, being likely to occur linear or ramified variation. Finally, the *Bitemporal* class can model spatial objects, considering both the recording and the valid time, inheriting thus characteristics of *TransactionTime* and *ValidTime* classes.

Following the same fundamentals of GeoFrame modeling, GeoFrame-T uses stereotypes to facilitate the modeling of objects and geographical phenomena. In the GeoFrame-T model, the *TemporalObject* class is considered as a time structure possible of being represented by the modeled classes. GeoFrame-T enables us to model conceptually the relationships between geographical phenomena and their temporal characteristics. These relationships are represented by the *Relationship* class. All the static and non-spatial relationships existent among the modeled classes are considered instances of the *Relationship* class.

4 MODELING THE VALID TIME

The descriptive, temporal and spatial dimensions are orthogonal. Temporal properties can be defined by the three categories of geo-spatial data: conventional objects, geographical phenomena perceived in field view and geographical phenomena perceived in object view. A model that considers the three types of time (valid, transaction and existence) for both classes and attributes and relationships provides the designer with great power of expression. However, the great number of possible combinations also makes the model difficult to learn and understand, eventually compromising the implementation of applications needed for handling spatio-temporal data.

Experience has showed that in most GIS applications only the valid time of the geographical phenomena has been considered, since this is the most significant piece of information in most queries involving temporal data. This article shows, in this way, how design aspects spatio-temporal based on the valid time, using UML-GeoFrame model, in which only the valid time is considered.

Valid time is the time instant or time interval when an object of the real-world is considered valid. For example, the strike against the World Trade Center took place on September 11, 2001, in turn, the Gulf War occurred in the period between August 1990 and February 1991. Hence, another important factor is the granularity of temporal information. The UML-GeoFrame model considers three types of time granularity: *Date, Time and Timestamp*. Specifying the granularity of a temporal attribute is the same as defining the domain of a descriptive attribute value (e.g. Char or Boolean). Finally, two types of temporal classes are discriminated: *Instant Class* and *Interval Class*.

If a class is modeled as being of *Instant* type (stereotype <◷>), it means that its objects are only valid at a particular point in time. In this case, the object does not evolve, since its validity is condensed into an instant. It is the case of a road accident representation, in which is essential to associate an instant of time with the object.

If the class is of the *Interval* type (stereotype <◑>), it means that its objects are valid in a period of time, i.e., between an initial and a final temporal value. These valid intervals do not necessarily have the same size. In addition, the object evolution is maintained, because its attributes can vary in the period corresponding to its valid interval.

If a class is temporal of *Interval* type, it indicates that every change in any object attribute will generate a new version of the object, and the old one will not be lost.

Each temporal class must have its granularity specified. The option *Date* is the default option and indicates that a value of the date type has to be stored. The option *Time* indicates the need for storage of an complete hour value. The option *Timestamp* indicates that a pair (date, hour) must be stored into the database. Each object (or object version) is associated with a piece of temporal information, characterized by a stereotype, along with the chosen granularity.

Besides allowing the modeling of temporal classes, the UML-GeoFrame model enables the representation of *temporal associations*, which is identified by the stereotype <<time>>. The validity of an association can be defined as the intersection of object's valid periods of classes involved in this association. This is because it is impossible for a relationship to exist in an instant of time in which a related object is not valid. Thus, considering *t* as this period of intersection, it follows that:

Temporal Association - its validity must be contained into the interval *t*, i.e., the association must be valid at the most for the period of time in which both objects coexist in time;

Non-temporal Association - its validity is same as interval t. In this case, the interpretation is the same given to conventional relationships, where the association is valid while the objects coexist in time. Some association examples involving classes of spatio-temporal phenomena can be presented, when two classes of geographical phenomena are perceived in object view, where both were modeled as temporal classes of the *Interval* type.

For example, an association between the classes Country and Epidemic, where the class Country could have information of name, population and

GDP (Gross Domestic Product). These attributes could vary in time, generating different versions of an object from the Country class. The Epidemic class describes epidemics that could occur in these countries, through the temporality. This temporality is important because of the need for observing its evolution in time, as well as possible periods of outbreaking and eradication. This relationship is defined as temporal (<<time>>), indicating that a same type of epidemics could occur in more than a period in the same country.

The association in Figure 2 illustrates the Fire and Building classes, the former being of non-geographical object type, but temporal of *Instant* type. The Building class describes the existent buildings and their use, such as museums, hospitals or schools.

The Building class is temporal of *Interval* type, which means that a building has an existence period characterized by the time interval relative to each type of use. The association (non-temporal) between these two temporal classes will make each Fire occurrence be associated with the most recent version of a Building object. Therefore, in case one wants to register the historical occurrence of a fire, it will not be possible to associate it with the correct version of the Building object.

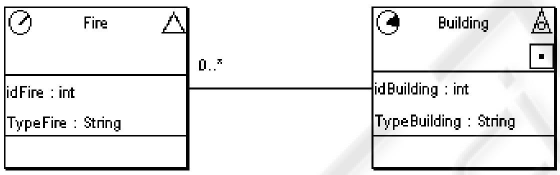

Figure 2: Non-temporal association with temporal classes.

5 TEMPORAL MODELING IN THE ARGOCASEGEO TOOL

ArgoCASEGEO is a CASE tool built to support modeling and design of geographical databases with the UML-GeoFrame model (Lisboa Filho et. al., 2004). To support the modeling of temporal classes, the modules Graphic Design, Data Dictionary and Automatic Generation were changed.

The ArgoCASEGEO Graphic Module was extended to support the temporal builders modeling, in other words, to enable the specification of the stereotypes *Instant* <◷> and *Interval* <◔>, as well as allowing the designer to specify a type of granularity for each temporal class.

A conceptual data schema drawn up using the ArgoCASEGEO tool is stored in a XMI file (*XML Metadata Interchange*), which is automatically created in the Data Dictionary Module. New tags were added to the XMI schema to store the definitions of temporal characteristics of a class.

The Automatic Generation Module (AGM) is responsible for the application of rules that transform a conceptual data schema into a logical-spatial data schema. As there is still no standard model adopted by the available GIS software, the ArgoCASEGEO tool has an AGM for some commercial systems and also an AGM for the Open Source Library - TerraLib (Vinhas, 2005).

Aiming at validating the temporal extension proposed to the UML-GeoFrame model, in this work the AGM-TerraLib was extended to enable the specification of logical data schema. Thus, starting from the XMI file, the AGM-TerraLib transforms an UML-GeoFrame conceptual schema into a TerraLib logical schema. In the following section the transformation rules implemented are described.

6 RULES FOR TEMPORAL ASPECTS TRANSFORMATION

The transformation rules are defined using the relational model as basis, following the TerraLib schema. Therefore, a primary key must be specified in each class in order to carry out the transformation correctly.

6.1 Rule 1: Transformation of Temporal Classes

The attributes defined for a class generate attributes in a corresponding relation. A temporal attribute with the domain defined in accordance with the specified granularity is added to temporal classes of *Instant* type.

For temporal classes of Interval type, having a primary key (PK) and a set of attributes {att}, a relation with PK is created. The foreign keys, if existing, are also added to this relation. In addition, a second relation is created to store the versions of its objects. To guarantee the integrity, a relationship 1..N between the first relation and the version relation is also created.

The primary key of the version relation is PK, plus a temporal attribute. Thus, the version relation will have the following schema:

```
RELATION_VERSION = (PK,begin,end,{att})
```

6.2 Rule 2: Transformation of Temporal Associations

Given a temporal association, regardless of its multiplicity, a new relation is created with the primary keys of the involved classes. This new relation is required for storing the validity period of the association. With this objective, two temporal attributes are created to indicate the beginning and the end of the association validity.

One wants to create an association between them in order that it is possible to store data on river sample collection for each collection point, with sampling being done at different instants.

The first alternative is to model a non-temporal association representing that each collection point has several samples.

Another alternative is to model a temporal association, indicating that each collection point has only one sample at a same instant in time. In this case the relations schemas shown in Figure 3 will be generated.

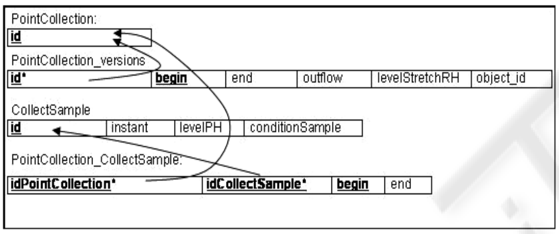

Figure 3: Schemas generated from a temporal association.

The designer can take the second option to express better the association. However, there is a question to be discussed; the attributes *begin* and *end* of the relation PointCollection_CollectionSample are redundant, since the only value that they can take is the valid instant of the sample collection. This restriction must be guaranteed by the application. It also must not allow that a point has more than one sample at a same instant, because this is incompatible with what was specified.

7 CONCLUSIONS

In this article we presented a proposal to simplify the conceptual modeling of temporal aspects in spatio-temporal database. The concepts were presented as an extension to the UML-GeoFrame model, however they can be adapted to any conceptual data model.

Modeling the temporal aspect of an application has not been a trivial task, probably due to the complexity of the existent temporal models in the literature. The completeness of the temporal aspects is actually necessary in a very restricted set of problems, for the great majority of temporal applications, modeling valid time of real-world objects is enough.

Some examples of modeling of classes and temporal relationships were presented and the Argo-CASEGEO tool support for modeling of spatio-temporal aspects was described.

At the present, only the Automatic Generation Module for TerraLib library is dealing with temporal aspects. An AGM for the Oracle-Spatial system is being implemented and the temporal extension will be drawn up for AGM-ShapeFiles.

ACKNOWLEDGEMENTS

This work was partially financed by the Foundation for Research Support of the Minas Gerais - Fapemig.

REFERENCES

Bédard, Y., 1999. Visual modelling of spatial databases towards spatial extensions and UML. *Geomatica*, v.53, n.2.

Booch, G., Jacobson, I. and Rumbaugh, J., 1998. *The Unified Modeling Language User Guide*. Addison-Wesley.

Goodchild, M. F., 1992. Geographical data modeling. *Computers & Geosciences.* 18(4).

Lisboa Filho, J. and Iochpe, C., 1999. Specifying analysis patterns for geographic databases on the basis of a conceptual framework. In *Proc. of 7th ACM Symposium on Advances in GIS*, Kansas City.

Lisboa Filho, J. Sodré, V. F., Daltio, J. Fidelis, M. R. and Vilela, V., 2004. A CASE tool for geographic database design supporting analysis patterns. In *Workshop on conceptual modeling for GIS, Shanghai. Springer-Verlag LNCS 3289.*

Parent, C, Spaccapietra, S. and Zimányi, E., 1999. Spatio-temporal conceptual models: data structures + space + time. In *Proc.7th ACM GIS*, Kansas City.

Rocha, L. V., Edelweiss, N. and Iochpe, C., 2001. GeoFrame-T: A temporal conceptual framework for data modeling. In *Proc.9th ACM GIS*, Atlanta.

Svinterikou, M. and Theodoulidis, B. 1997. *The temporal Unified Modeling Language (TUML)*. Dep. of Computation, UMIST, UK, (TR97-1).

Tryfona, N., Price, R. and Jensen, C. S., 2003. Spatiotemporal conceptual modeling. In Spatiotemporal Databases: the Chorochronos approach, Springer-Verlag.

Vinhas, L. and Ferreira, K. R., 2005. Descrição da TerraLib. In *Casanova, M. A. et al. Bancos de Dados Geográficos*, Curitiba: MundoGeo.