IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4999

An Object-Oriented Approach for Modeling Discrete Spatio-Temporal Databases

Hatem F. Halaoui
Computer Science Department, Graduate Center/ City University of New York, 365 Fifth Ave New York, NY 10016-4309,
Email: hhalaoui@gc.cuny.edu

ABSTRACT

Spatio-Temporal databases became a must for many Geographical Information Systems where applications in these systems are not only interested in current values but also past, present and sometimes future values as well. This made the interest in finding efficient models for them higher and increased the work in the area. A lot of models were proposed for representing spatio-temporal databases as well as predicates and operations that can be applied on them. This paper offers an object-oriented approach based on "Composition" or "Has a" relation to model such databases where geometrical objects are attached to time to form a spatio-temporal object. The model is most suitable for objects or geometries in space that change discretely with time, however the idea can also be used to represent other spatio-temporal databases like moving object or combination of moving objects and geometries. The idea of the model is based on parts of ideas in two independent models. The first is a model offered by Tansel[2,3,4] for representing temporal elements and the second one is used in ROSE Algebra[9,10] that proposes data types, predicates and operations for spatial databases.

1. INTRODUCTION AND RELATED WORK

There is a big demand on Geodatabases or spatial databases to manage and manipulate geographical information. Examples of spatial information that need to be managed are land-ownership, place-locators, environmental and agricultural information of lands and territories, tracking moving airplanes or objects, road networks, phone cable networks, and hundreds of other examples. Also most of this information may change with time as well. For example, airplanes change their location with time, information about lands may change with time, road networks might always be updated with time and so on. Databases that change with time are called temporal databases and hence spatial databases changing with time are called spatio-temporal databases. Most of spatio-temporal databases fall under three main categories of applications.

(1) Applications dealing with moving objects.
(2) Applications dealing with objects that have some geometry and located in space.
(3) A third kind of spatial databases is a combination of the above two.

Many approaches and models were offered to represent spatial, temporal or spatio-temporal databases. In spatial databases, "ROSE Algebra or The Realm-Based Approach" [9,10] is one of the best representations for spatial databases where Realm based representations of geometries are used to represent abstract data. The approach was extended or updated later to " Dual grid approach" [5] to avoid some of the side effects of the first approach. Moreover, Gütting [8] present a nice introduction on spatial databases where he concentrates on "ROSE Algebra" [9,10] description of modeling and data typing. In temporal databases, Gadia [11] presents a nice N1NF homogeneous model where Tansel [3,4] offers a N1NF heterogeneous model. Also Snodgrass [12], Lorentzos and Johnson [13] propose a 1NF approach for temporal representation. Spatio-Temporal databases can be seen as a combination of spatial and temporal databases. Worboys [7] has one of the best models representing spatio-temporal databases where data types, predicates and operations for such databases are proposed.

This paper will offer an object-oriented approach for modeling spatial databases. Spatial data types from ROSE algebra will be used. Also, the idea of attaching time to objects from Tansel [2,3,4] approach of modeling temporal databases will be used as well.

The organization of the paper will be as follows. The second section will be introducing and defining spatial data types and their representation. In the third section, time will be added and spatio-temporal representations will be defined and used. The fourth section defines predicates and operations that could be needed for the proposed representation. Land-Ownership example is used to illustrate the introduced model in the fifth section. The last section will be demonstrating conclusions and some future ideas.

2. SPATIAL DATA TYPES AND THEIR REPRESENTATION

From the Realm based approach (ROSE Algebra) [9,10] consider the following abstract data types: Points, Lines, and Regions where:

Points: is a set of points
Lines: is a set of lines
Regions: is a set of regions.

Definition 2.1: Let GEO = {ID, Points, Lines, and Regions} to be defined as set of points, lines, and regions that define a geometry with ID as a key.

The relation between the objects is an object-oriented relation ("Composition") where a line composed of two points (composition relation) and a region is composed of many lines with some constraints. Also a GEO is composed of an id and a set of points, lines and regions. Figure 2.1 present the relation between the spatial object definitions.

Figure 2.1: Relations between spatial objects

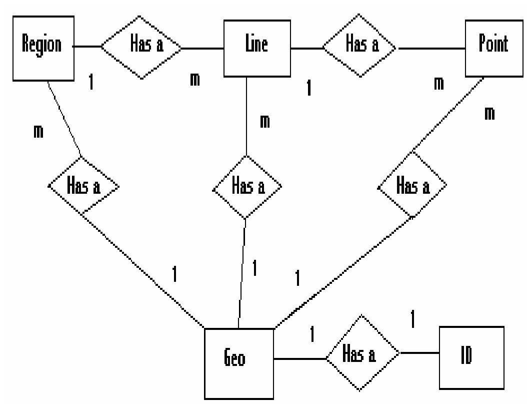

Figure 2.2: GEO G1 consisting of one point p1, 2 lines l1 and l2, and a region r1

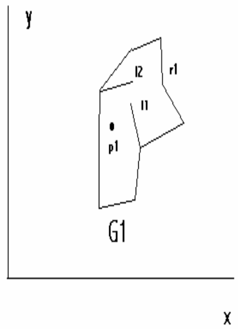

Figure 2.2 illustrates an example of a GEO that is identified by G1, p1 is a point in G1, l1 and l2 are lines in the G1 and finally r1 is a region in G1.

The advantage of using “ROSE Algebra” [9,10] spatial data typing is that all predicates and operations described there can be applied on any objects (instants) of the these types.

3. SPATIO-TEMPORAL DATA TYPES AND REPRESENTATION

In this section, the above definition of GEO is extended, by using the notation followed by Tansel [2,3,4] when he models temporal data. In his definition, Tansel [2,3,4] attached a time dimension to any object that has its value changing with time and hence this object becomes a temporal object. Similarly here, a time dimension will be added to a GEO object that will be referred to by its ID. The following definition will clarify the idea.

Definition 3.1: A temporal geometry “TGEO” is a pair <GID,T> that defines the geometry of an object at a specific time T where GID is the ID of a GEO object.

TGEO is not very complex. Only the ID of the geometry is attached to the time while the geometry itself is stored in a separate database or file. This advantage makes the model less complicated. When the GID is known (ID of the geometry) it is very easy to retrieve it from its databases and apply in any application. The way time is attached, which is similar to Tansel [2,3,4], makes the model flexible where also Tansel Algebra and operations [2,3,4] on temporal data can be used and applied.

Definition 3.2: “TGEO” is defined as a set of temporal geometries each of the form “TGEO”.

Figure 3.1 illustrates a graph representation of a <u>TGEO</u> ={<G1,2>, <G2,0>} where G1,G2 are the ID’s of one geometry at two times 2 and 0 respectively.

Figure 3.1: A set of Temporal Geometries <u>TGEO</u> that has 2 different shapes (G1 and G2) at 2 different times(2 and 0 respectively).

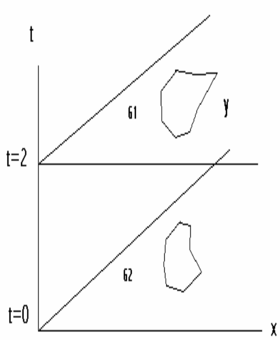

Table 3.1: LAND relation

LID	OWNER	GEOM
L1	Ann	{<G1,1>,<G2,2>}
L2	Joe	{<G3,1>,<G4,2>,<G5,3>}
L3	Mark	{<G6,1>,<G7,2>}

Table 4.1: Table of operations

Operation	Operand	Operand	Return Type
Spatial Projection (π^s)	TGEO		GEO_ID
Temporal Projection (π^t)	TGEO		T
Spatial Selection (σ_c^s)	TGEO		TGEO
Temporal Selection (σ_c^t)	TGEO		TGEO
Spatial Union ($\cup$)	GEO	GEO	GEO
Spatial Intersection (?)	GEO	GEO	GEO
If Equal (==)	GEO	GEO	BOOLEAN
IfSubset ($\subset$)	GEO	GEO	BOOLEAN
Boundary (β)	GEO		Region
Area (Δ)	GEO		REAL
Overlaps (Θ)	GEO	GEO	Boolean

Definition 3.3: An object in space can be defined as a Spatio-Temporal relation $R(A_0,..,A_n)$ such that there exist at least one Attribute A_i ={<u>TGEO</u>}.

Consider the LAND relation as an example of an object in space.

LAND(LID, OWNER, GEOM)

Where LID is the land ID, OWNER is the name of the owner and GEOM is a set of temporal geometries (TGEO). Table 3.1 is an example of the LAND relation table.

Note that the geometry database is a separate database where geometries (of GEO type) is saved and they are referred by their ID’s in the LAND relation. The geometries can be retrieved from their database or file when they need to be manipulated.

The relation will be used later in section 5 as a sample example where some predicates and operations will be applied.

4. PREDICATES AND OPERATIONS

Table 4.1 has spatial and temporal predicates and operations that can be applied on such model in addition to regular algebraic operations.

4.1 Predicates and Operations Brief Specifications

- Spatial Projection (π^s) detach the GEO object from time and return the GEO object while Temporal Projection (π^t) does the same thing but return the time.
- Spatial Selection (σ_c^s) selects a TGEO with a spatial condition while Temporal Selection (σ_c^t) with a time condition.
- Spatial Union (U) gets the union of two GEO’s by returning a GEO object that is the combination of the two GEO’s. Spatial Inter-section (•) on the other hand returns the common region (as a GEO object) of two GEO objects if exist and empty otherwise.
- If Equal (=) checks if two GEO’s objects are equal by comparing them together while IfSubset ($\subset$) checks if a GEO object is a part of another GEO object. Av checks if two GEO objects have any overlapping regions.
- Boundary (β) gets the boundary region of a GEO if exist and empty otherwise and Area (Δ) gets the area of the boundary region of a GEO object

4.2 Application of Some Operations: Spatial Union and Intersection

Consider the two GEO objects G1 and G2. Figure 4.1 show the two objects, figure 4.1 illustrates their union while figure 4.2 illustrates their intersection

Figure 4.1: G1 and G2

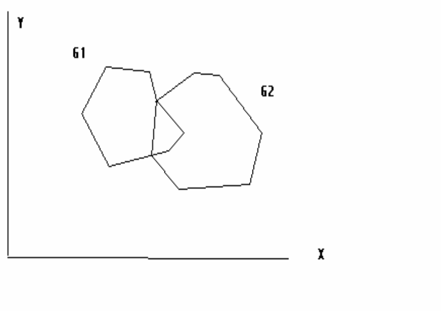

Figure 4.2: G3 =G1 U G2

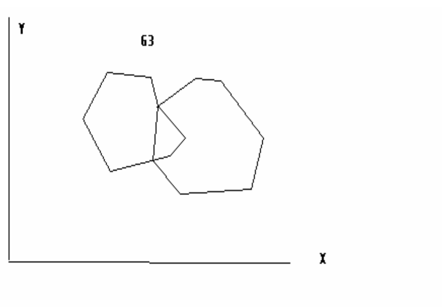

Figure 4.3: G3 =G1)” G2

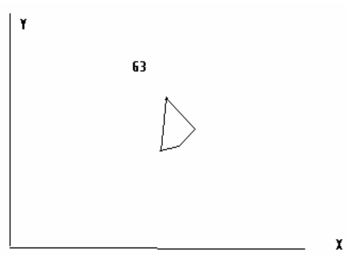

Table 5.1: LAND relation

LID	OWNER	GEOM
L1	Ann	{<G1,1>,<G2,2>}
L2	Joe	{<G3,1>,<G4,2>,<G5,3>}
L3	Mark	{<G6,1>,<G7,2>}

5. LAND OWNERSHIP EXAMPLE

In this section, the LAND relation defined previously will be used in some queries to illustrate an example of how such a model can be used. Again the LAND relation is defined as follows:

LAND (LID, OWNER, GEOM)

Also consider some data information about the LAND relation shown in table 5.1.

Consider the following query:

What the geometry of Ann’s land at time 2 ?

Step 1. Retrieve the tupple with owner =“Ann”
($\sigma_{OWNER=\text{”}ANN\text{”}}$(LAND)).
Step 2. Project on the GEOM column.
(π_{GEOM} ($\sigma_{OWNER=\text{”}ANN\text{”}}$(LAND))) Step 3. Get the ID of the geometry at time=2.
(π^t ($\sigma_{T=2}{}^t$ (π_{GEOM} ($\sigma_{OWNER=\text{”}ANN\text{”}}$(LAND))))).
Step 4. From the Geometries database, get the geometry itself that has the ID got in step 2.

Assuming that the geometries database is called GEOMETRY (GID,….), then resultant query could look like the following:

$\sigma_{GID=}\,\pi_{t(}\,\pi_{T=2t\,(}\;\sigma_{GEOM\,(}\pi_{OWNER=\text{”}ANN\text{”}(LAND)\,)\,))}$ (GEOMETRY)

6. CONCLUSIONS

This paper has discussed and object-oriented way of modeling of spatio-temporal databases by using object definitions (data types) used in “The Realm-Based Approach” [9,10] and attaching a time dimension to them as in Tansel [2,3,4]. The paper did not go briefly into the details of time which could be transaction time, real event time, or it could be a time interval. Such times could be added as another attribute similarly. There are some issues about this model.

- It is easy to understand
- The spatio-temporal representation is not complex since its ID only is attached to time.
- Spatio-Temporal data is not complex as well since the geometries are saved in a separate database.
- Operations in ROSE algebra as well in Tansel algebra could be used partially.
- The model easily be extended by adding dimensions.

The model is suitable for objects in space like lands, parcels, maps, and similar geometrical shapes. However the idea of the model could be used to express moving objects as well. A new type “POSITION” that is composed of a point could be added to determine the position of any object, which could be a spatial object.

REFERENCES

[1] A. U. Tansel and E. Tin. The Expressive Power of Temporal Relational Query Languages, IEEE Transactions on Knowledge and Data Engineering, VOL. 9 No. 3 (1997).

[2] A. U. Tansel. A Historical Query language. Elsevier Science Publishing Co. Inc. (1991).

[3] A. U. Tansel. Adding Time Dimension to Relational Model And Extending Relational Algebra, Information Systems Vol. 11, No. 4 pp. 343-355 (1986).

[4] A. U. Tansel. Temporal Relational Data Model. IEEE Transactions on Knowledge and Data Engineering, VOL. 9 No. 3 (1997).

[5] J. Antonio, C. Lema and R. H. Guting. Dual Grid: A New Approach for Robust Spatial Algebra Implementation. CHOROCHRONOS project (2000).

[6] M Erwing, R. H. Güting, M. Scneider, and M. Vazirgiannis. Abstract and Discrete Modeling of Spatio-Temporal Data Types. ACM (1998).

[7] M. Worboys.A Unified Model for Spatial and Temporal Information. Department of computer science, Keele University (1994).

[8] R. H. Güting. An Introduction to Spatial Databases. VLDB Journal Vol. 3, No 4 (1994).

[9] R.H. Güting and M. Schneider. Realm-Based Spatial Data Types: ROSE Algebra. VLDB Journal, 4(2):100143, 1995.

[10] R.H. Güting, T. de Rider, and M. Schneider. Implementation of the ROSE algebra: Efficient Algorithms for Realm-Based Spatial Data Types. In Proc. Of the 14th Intl. Symposium on large Databases, pages 216-239, Portland, Maine, August 1995.

[11] S. Gadia. A Seamless Generic Temporal Extension of SQL. Computer Science Department, Iowa State University (1992).

[12] R.T. Snodgrass. The Temporal query Language Tquel. ACM Trans. Database Systems, vol 12, no 2, pp. 247-298 (1987).

[13] N.A. Lorentoz and R.G. Johnson. Extending Relational Algebra to Manipulate Temporal Data. Information Systems, vol. 13, no. 3, pp. 289-296 (1988).

Related Content