Spatio-temporal Data Model Based on Relational Database System

SHA Zongyao  BIAN Fuling

1 Introduction

Spatio-temporal information system improves existing spatial information by handling temporal information. Enormous researches have been made in spatial database.1-5, Because the traditional relational database has a strong basis both on theory and application, and if well extended, it can also be used to handle spatio-temporal data. A relational database system can be used to handle some specific issues in spatio-temporal database. Spatio-temporal data contains many useful patterns that may be mined. It is relatively easier to find those patterns in relational database by using data-mining techniques, most of which are focused on relational databases currently.

Before spatio-temporal information system becomes practical, numerous issues need to be investigated.6 A. U. Tansel has compared the storage, retrieval and update efficiency of spatio-temporal data storage methods organized in different ways.5 In static mode, snapshots keep all full states which lead to the storage of redundant information. To detect changes between snapshots, relatively expensive computational algorithms must be used, though this still does not explain the processes leading to the change. In differential mode, only the initial state is fully recorded. Changes are stored in one of two possible kinds of “delta files”, which record the difference from either the previous state or the initial one. In order to reinstate the previous states or the current one, a series of delta files must be applied to the initial state. The third one is sequential updating mode. This mode, which also keeps the current state of the map on record, only records changed records, and uses indexes to access previous information, eliminating data redundancy.

KEY WORDS  GIS; spatio-temporal data model; relational database; spatio-temporal analysis

ABSTRACT  In this paper, the entity-relation data model for integrating spatio-temporal data is designed. In the design, spatio-temporal data can be effectively stored and spatio-temporal analysis can be easily realized.
In this study, logical data design of GIS spatial database is focused in Section 2. A case study on spatio-temporal analysis in practical application is presented in Section 3.

2 Spatial data organization and storage

Spatial data, unlike the ordinary attribute data, need well-designed data model. In this study, the general idea of spatial data organization and storage is described as follows:

RDBMS manages basic data. The Client sends a request to database and the database Server retrieves the requested data and delivers them to the Client. The Client then makes a visual show through GUI (Graphic User Interface) commands that are also stored and managed by RDBMS.

2.1 Metafile

Metafile is a set of GUI commands, which can be created by a metafile device description table. When a metafile is created, its data can be redrawn by any application, so, by this method, data exchange under different structures becomes possible. When needed, the information can be accessed by standard SQL language.

2.2 GIS abstraction of units

In 2D model, the basic entity can be abstracted as point, line, and polygon. Point can be expressed with a pair of coordinate value \((x, y)\). Line is composed of a series of points. Polygon is different from line because the former has a round point series, i.e. the first point and the end point are the same. In this research, GIS abstraction of units is used to design data structure mode to explicitly store spatial objects and topological relations.

2.3 Logical data design

GIS data mode design can be complex. In order to represent the real world correctly and efficiently, an effective data structure and data model design is essential for a specific application. We take the advantage of relational database management system to express the real world, so the first step is to design a relational mode. With regard to GIS specialty, a relational database mode can be divided into the following sub-modes, which have a hierarchical structure level.

2.3.1 Relation database table (RDBT)

RDBT mode is \(RDBT(ID, region ID, layer ID, notification)\), where \(ID\) is the unique number standing for a given research region and \(layer\ region ID\) is a divided region number standing for a given region which is a small part of the research area; \(layer ID\) is the research layer in a certain region, and notification, which may include several fields, is some special information about the research ID region and layer. Through this method, spatial analysis becomes easier, especially for very large database.

2.3.2 Layer table (LT)

The essential fields for layer table include \(ID\), stamp \(ID\), and index \(ID\). This table can be defined as \(LT(ID, stamp ID, index ID)\), where \(ID\) is a key that connects with the corresponding RDBT; stamp \(ID\) is the unique number for linkage with other tables, and index \(ID\) is that for index. In fact, the above two tables present no data but some messages in the research area.

2.3.3 Stamping table (ST)

Stamping table gives any spatial object a unique number in any ID research area. This table contains stamp \(ID\), object \(ID\), and the valid time of the spatial object using fromtime and totime fields, thus this mode will be \(ST(stamp ID, object ID, fromtime, totime, beforestamp ID, beforeobject ID)\), where stamp \(ID\) is derived from LT table, object \(ID\) stands for a unique object; fromtime and totime stand for the object’s valid (world) time, which is explained as a fact of the object becoming true in the real world; beforestamp ID is the stamp ID that the object belongs to; and beforeobject ID is the object ID that the object comes from (if the object is not derived from any object, the value will be zero). The value of fromtime can be the real time when the spatial object starts to occur and totime can be the time when the object disappears. The “forever” is set to totime if the object never disappears. Although it also allows more dimensions for time (transaction time), if necessary, transaction time can be handled like valid time. Two spe-
cial fields, before stamp ID and before object ID are added to solve object relation in different time, for example, we can get to know the history of any object at any time. This function is very important in object sequential analysis. Now, the spatial object’s unique number is obtained, but the real data about an object are still not stored in this table.

2.3.4 Meta table (MT)

This table stores information about the object’s description with field object ID, object type, number of points, and other attributes, depending on various applications. So this can be written as MT (object ID, object type, number of points, other attributes), where object type is a point, line or polygon. Attributes about spatial objects are stored in this table.

2.3.5 Coordinate tables

According to object type, coordinate tables include point table, line table, and polygon table. All these tables include object ID, which is copied from MT. As for point table, x coordinate and y coordinate are included. Those points object ID may enter line table, depending on whether the point is a part of a line or not, and the same is for line to polygon. In all the tables, point table is the most frequently accessed and updated table that all object features stored in database have to refer to. Indexes are built to increase data accessing and updating speed. Here an R-tree\(^7\) is adopted. Minimum bounding rectangle and its center position coordinate of each spatial object are recorded. Each node of a tree of order \(M\) has at least \(M/2\) and at most \(M\) entries of the form \((R, P)\), where \(R\) is a rectangle that covers all the rectangles of the children of the node, and \(P\) is a pointer that either points to a child in a non-leaf node or to a spatial object in leaf node. Object spatial data are stored in coordinate tables.

It is very hard to design an absolutely satisfactory relational table. The difficulty lies in the constraint of relational databases because the numbers of lines that compose a polygon are different, and so is that of points. Although it is possible to use an object-oriented data model that excellently overcomes the shortcomings of relational database, the shortcomings that object-oriented data model lacks effective query language discount its application area. The difficulty facing relational databases mentioned above can be solved by integrating the various lengths of records into one record that can contain various items. The whole relation between those tables is shown in Fig. 1.

![Fig. 1 Hierarchical logical organization of database](image)

3 Case study

Dynamic land survey is very important in land management because it offers valuable information for government decision-making. In this experiment, this topic was studied. Visual Basic language is used to develop Client application, with the relational database management system SQL Server 7.0 as its server. According to object types, three layers, viz. point layer, line layer, and polygon layer, are abstracted to stand for three different spa-
tial entity types, each with a unique name. Three levels of system structure are used in this study. The structure of this three-level data model is shown in Fig. 2. Operations upon spatial data can be done at the end of the application, application service or data service with the intention of increasing efficiency of data disposal. The advantages of this structure have been discussed and have been adopted by many researchers, and in fact, this structure have been adopted by many corporations in their data exchange and data management.

3.1 Spatial retrieval

Using the standard query language, we can extract spatial data from relational databases and visualize them on screen. In fact, without knowing the physical location of the application service, the Client just sends its query to the application service and gets data from the data service through application service. The data then can be redrawn on Client. The structure can be seen in Fig. 2. If all spatial objects existing in 1997 have been retrieved then its query format can be expressed as follows:

1) selecting ID from RDBT where region ID = region1 and layer ID = layer1 InTo tempRDBT
2) selecting stamp ID from LT where ID = tempRDBT, ID InTo tempLT
3) selecting object ID from ST where stamp ID = tempLT, stamp ID InTo tempST and fromtime = 1997 or (fromtime < 1997 and (totime > 1997 or totime = 0)) InTo tempST
4) selecting objecte ID from MT where object type = polygon and object ID = tempST, object ID InTo tempMT

After the spatial objects have been retrieved, GUI commands or class objects are used to visualize these data on screen. This case can often be seen in a land use management information system that has an obvious character of managing spatio-temporal data.

3.2 Spatial data updating concerning temporal information

When a spatial object changes its position and/or shape, and when a new object is added to database, adding the new position on coordinate table(s) and attaching its valid time in Meta table are needed, which will avoid the storage of redundant data.

3.3 Spatio-temporal data analysis

Because any object in a database is attached with its valid time, temporal analysis can be easily realized (see Fig. 3). For example, if a river’s changes are to be displayed over the time from 1993 to 1997, the SQL can be used as follows:

1) selecting ID from RDBT where region ID = region1 and layer ID = layer2 InTo tempRDBT
2) selecting stamp ID from LT where ID = tempRDBT, ID InTo tempLT
3) selecting object ID from ST where stamp ID = tempLT, stamp ID InTo tempST and fromtime = 1997 or (fromtime < 1997 and (totime > 1997 or totime = 0)) InTo tempST
4) selecting objecte ID from MT where object type = line and object ID = tempST, object ID InTo tempMT

The above steps have got all line objects in table tempMT, then a specific river from tempMT needs to be extracted. Suppose specific object has one
attribute type ID stored in MT, and river has the field value as river, all rivers will be chosen using:

5) selecting object ID from tempMT where type ID = river InTo tempMTriver

Suppose every river has a unique value labeled as UNI, and River A’s UNI is A. River A changed its course every year, but we measured its course only in 1993, 1995 and 1997. In order to get the three years data, we use:

6) selecting object ID from tempMTriver where UNI = A

By using interpolation of 3D surface construction method[8], the river course in any year can be calculated and shown on screen.

![Diagram for spatio-temporal analysis in GIS](image)

**Fig. 3** Diagram for spatio-temporal analysis in GIS

### 3.4 Family chain analysis

Sometimes we need to find out whether a given spatial object has a “father” or even “grandfather”, and if it has, what are they? We call this “family chain analysis”. This function can be realized by using the two fields defined in ST table: beforestamp ID and beforeobject ID. For example if we want to look for the family chain of an object, the steps will be:

1. **Step 1**: determine which object you want to find in spatial database;
2. **Step 2**: make sure whether the beforeobject ID of the object is NULL, and if not do Step 3; else Step 5;
3. **Step 3**: from the father’s beforeobject ID, look for the father’s object ID, this is the father of the given object;
4. **Step 4**: go Step 2;
5. **Step 5**: draw out the chain of a given family;
6. **Step 6**: end

### 4 Conclusion

The method with a relational database have been discussed, and the advantages of this relational database have been reported. But within the community of GIS, a relational database is usually used to manage attributes of geographical objects, while the file method are used to manage graphical data. Shortcomings of this way are obvious. Accordingly, in this study, both attribute and graphical data are put under the management of a relational database. Some of its advantages are:

1) It takes advantages of relational database, such as data security, data availability, data efficiency, data standardization, data consistency, etc. thanks to strong power of relational algebra.

2) This method can store temporal data effectively, and can be used to analyze spatial objects’ moving tendency both for discrete or continuous changes by using extrapolation or interpolation[9].

3) It reduces data redundancy since only changed characters are to be stored in database.

4) Family chain analysis can be easily realized.

5) Classical data mining techniques such as generalization, classification, association can be easily used to find spatial regularities or rules in relational database.

Although relational database can be and actually has been used to manage spatial and temporal data in GIS in our study, problems also exist which have
to be further researched. Although topological relations are stored in the system in order to ease spatio-temporal analysis, operation efficiency may decrease.

References

1 Goenthe X. O. and Buchmann A. (1994) Research issues in spatial database. SIGMOD Record, 19(1):61-68
2 Goting R. (1994) An introduction to spatial database systems. VLDB Journal, 3(4):357-399
3 Medeiros C. B. and Pires F. (1994) Databases for GIS. SIGMOD Record, 23(2):107-115
4 Gong J Y, Zhu X Y, Zhu Q. et al. (2000) Design and implementation of object-oriented integrated spatial database management system. Journal of Wuhan Technical University of Surveying and Mapping, 25(4):289-293 (in Chinese)
5 Tansel A. U. et al. (1993) Temporal databases: Theory, Design and Implementation. New York, Benjamin Cummings, 2-7
6 Abraham T., Roddick J. F. (1999) Survey of spatio-temporal databases. GeoInformatics, 3(1):61-99
7 Ester, M. et al. (1995) Knowledge discovery in large spatial databases: focusing techniques for efficient class identification. In: Proc. 4th Int. Symp. On large spatial databases, Portland, Me.
8 Zhang W., Hunter G. J. (2000) Temporal interpolation of spatially dynamic objects. GeoInformatics, 4(4):403-418
9 Bian F. L. (1996) The Principles and Method of Geographical Information System. Beijing: Surveying and Mapping Publishing House (in Chinese)
10 Liu R. Y, Liu N., Su G. Z. (2000) Manipulation of spatial graphical data in RDBMS and its implementation in application of GIS. Acta Geodaetica et Cartographica Sinica, 29(4):329-333 (in Chinese)