A CADASTRAL SPATIAL DATA STORAGE STRUCTURE 
BASED ON RELATIONAL DATABASE 

Luo De'an
Liao Lijiong

KEY WORDS GIS; spatial database; cadastral management

ABSTRACT This paper presents a cadastral spatial data storage structure based on relational database, the method and the procedure to realize it. The paper consists of three parts. In the first part, some existing problems in some developed cadastral management systems are discussed. These problems are the following four. 1) The security of cadastral spatial data is difficult to be assured. 2) It is difficult to verify cadastral data and the integrity of cadastral data is difficult to be kept. 3) To transmit and share cadastral data is difficult. 4) The efficiency of data access is low. In the second part, the feasibility of using relational database to store spatial data is analyzed and a new cadastral spatial data storage structure is presented. At the same time, the related table structures and field descriptions are given, and then the merits and demerits of this storage structure are analyzed in detail. In the last part, through a real example, the detailed methods to make the new storage structure a reality are given. Moreover, some involving key techniques of the new storage structure are discussed. These techniques are: 1) the application of database transaction, 2) the application of database trigger, 3) and the application of secure recovery of database.

1 Introduction

Cadastral maps and cadastral attributes are the core contents of cadastral information management. Some of the developed cadastral management systems employ GIS software (i.e. Mapinfo, Arcview or Maptitude etc.) to manage cadastral maps and use relational database (e.g. SQL Server, Sybase or Oracle etc.) to manage cadastral attributes, and some of them even directly to employ a GIS Platform to manage both of them. Some demerits exist in these developed cadastral management systems, which will be discussed in detail in the following passages.

1) Difficulty in the data security assuring

At present, most commercial GIS platforms keep spatial data and attributes stored separately. Spatial data are stored in a file or files according to a GIS software developer's definition and attributes are stored in a relational database, and the storage style has ever attributed to GIS application and its spread. But in a view of GIS development, the above spatial storage solution is only an integration and has an obvious demerit, which is the less security. In general, the security of file is managed by operation system with less security than database, so the security of cadastral spatial data stored in a file or files will not be a perfect solution. All the cadastral documents (both cadastral maps and cadastral attributes) are the legal credence of land users, and are very important to both land users and land managers. The security of cadastral documents is the most important in cadastral management system, so it should be thought much of in the process of designing cadastral stor-
2) It is difficult to change cadastral map objects and uneasy to keep their integrality

Cadastral change is the core of cadastral management. All daily cadastral work belongs to cadastral change domain after first land registration. Cadastral change consists of cadastral map change and its attribute change. Attribute change is easy to make reality while cadastral map change will be a difficulty in cadastral system design. Because of errors of maps input some fragments or splits will be produced in polygon combination or its split. Some measures (e.g. to join the fragments to the nearest polygon etc.) are taken to reduce these fragments and splits but these fragments and splits still cannot be eliminated entirely. Moreover, spatial data stored in a file or files is easy to be modified by wrong operations; if these modifications can not be corrected before commit operation they will become faults and be saved forever in GIS file, which will lead to loss the integrality of cadastral map. In short of the ability of security recovery this kind of faults can never be recovered in GIS spatial data management based on file.

3) Difficulty to in cadastral data transmission and sharing

Cadastral documents are important legal credence, and can help the government to do land statistics and tax land users. Data transmission and data sharing of cadastral documents are necessary in order to do land statistics and tax land users. Cadastral spatial data stored on file inevitably depends on the special GIS software. Different file formats exist in different GIS platforms. In order to keep their commercial interest, their interior file format will not be opened in general. In most case they provide interchange file format. So at this situation data sharing can be realized in two ways; employing the same GIS software or adding extra data transform to convert different file formats to a special file format.

4) Low efficiency of data access

GIS spatial data stored in file will bring about the following problems in its map object operations: (1) difficulties in locking map objects in multi-users environment, (2) low access and operation efficiency when insert or delete map objects, (3) and lower query efficiency than database.

Because of the above demerits of the storage structure based on the file or files, this paper presents a storage structure to employ relational database to store cadastral spatial data, and discusses its feasibility, data structure and the method of realization in detail.

2. Analysis of the feasibility to employ relational database to store cadastral spatial data

Data share and network management are two future directions of GIS development and GIS spatial data management based on file or files lacks both of them. Compared with the old storage structure the relational storage structure of spatial data has the following advantages: (1) perfect data security, (2) easiness to transmit and share data, (3) the strong ability of record lock and security recovery etc. These advantages provide cadastral data an efficient management and overcome the demerits in developed cadastral management systems.

To store spatial data in relational database is actually feasible. Spatialware and Oracle 81 are a spatial data storage solution based on relational database, and have the ability to store spatial data and attributes in database together. Although spatial data storage and attribute storage have differences on data organization and data structure, they are the same in nature. It is actually feasible to store spatial data in relational database if a legal data storage structure is introduced.

Of all the cadastral spatial data, the data about boundary nodes and property parcels is the most important. For the other cadastral spatial data, they are less important than the above two. So the storage structure and management style for the data about boundary nodes and property parcels should be different from the others’. For the data about boundary nodes and property parcels, because of their importance they should be stored in relational database. For the others’, because they are less important and usually treated as backgrounds, they may be stored in GIS file (i.e. Arcview’s .shp file).
or GIS files. However, you can store them in relational database if necessary.

3 The storage structure of cadastral spatial data based on relational database

Because of the solemnity and the legal function of cadastral management, cadastral survey must meet special procedure and criterion and the surveyed data must be real and present. Cadastral data can be classified as cadastral spatial data and cadastral documents. Cadastral spatial data includes boundary nodes, property parcels and other related map contents. In this paper the author employs relational database to store key cadastral spatial data and employs GIS file to store the other background map data. The storage structure of cadastral spatial data is illustrated in Fig. 1. (Here Sybase is employed as the database software and Mapinfo is employed as the GIS platform.)

In order to store in database the key cadastral map data, which here refers to the data of boundary nodes and property parcels, the object abstract work must be done and related data table structures must be designed. Here comes down to three tables, they are the coordinate table of boundary nodes, the basic table and its route table of property parcels, and their table structures are presented in the following passages.

![Fig. 1 The storage solution of cadastral spatial data]

1) The table structure to store the data of boundary nodes (jzd)

create table dbo.jzd (  
Uid identity, /* identity code */  
Jid char(20) not null, /* the number of boundary node */  
x real not null, /* x coordinate value */  
y real not null, /* y coordinate value */  
STYPE int not null, /* symbol style */  
...  
constraint jzd-1120034301 PRIMARY KEY CLUSTERED (Uid)
)

2) The table structure to store the basic data of property parcels

create table dbo.zd (  
Uid Myidentity identity, /* identity code */  
Zid char(20) not null, /* the number of boundary node */  
Ftype int not null, /* fill style */  
Ltype int not null, /* line boundary style */  
...  
constraint zd-480032021 PRIMARY KEY CLUSTERED (Uid)
)

3) The table structure to store the route data of the property parcel boundary

create table dbo.ZDindex (  
Uid identity, /* identity code */  
ZDuid int not null, /* the number of property parcel */  
JZDxh int not null, /* the serial number of boundary node */  
JZDuid int not null, /* the identity code of boundary node */  
JZDjid char(20) null, /* the number of boundary node */  
constraint ZDindex-1104069641 PRIMARY KEY CLUSTERED (Uid)
)

For the above three tables, the table to store the coordinate data of boundary nodes is employed to manage all the coordinate data of boundary nodes, and can be used to create or reconstruct the boundary node layer (reconstruct, refers to reconstruct the boundary node layer from related table when
destroyed). The table for storing the basic data of property parcels is employed to manage the essential information about property parcels. The table for storing route data of property parcel boundary is employed to store the route data of property parcel boundary. The property parcel layer can be created or reconstructed from the basic table of property parcels and its route table.

Even though the solution to store partial spatial data in relational database is employed, because the selected GIS platform (Mapinfo) stores spatial data in files, two temporary layers (i.e. the layer of boundary nodes and the layer of property parcels) must be employed to store related spatial data and interact with users. This solution has two obvious drawbacks: (1) Data redundancy is produced when the same spatial data are stored repeatedly by two temporary layers and relational database. (2) The system working efficiency is depressed because of additional transform work when operating the data stored in relational database. In fact, because spatial data is stored in relational database related software of spatial data processing may be developed to interact with the data directly.

The merits of the relational storage structure of spatial data are obvious regardless of its demerits. These merits are: (1) because spatial data is stored in database the security of the above two temporary layers is not so important, which can be recovered from database if damaged or destroyed; (2) the relational storage structure of spatial data is useful to long-distance data transmission and long-distance data access (e.g. transmit and access by PSTN or DDN etc.) and can provide the ability for data transmission (includes spatial data transmission) among province bureaus, town bureaus and county bureaus; (3) the above storage structure of spatial data is independent of GIS platform because of its open storage format. Any GIS platform can directly or indirectly interact with spatial data stored in relational database as long as they can provide functions to create point, line and polygon. The storage structure can be used for reference in the process corresponding with province and even national cadastral information management and related data transmission. The selection of GIS platform is not so important if a unique cadastral storage structure based on relational database is used by all kinds of land management bureaus. A large distributed database can be constructed if connecting distributed cadastral management systems, and any node can access the other one if accredited.

4 Implement of cadastral spatial data storage based on relational database

The two key techniques to employ relational database to store spatial data are input data from database and output data to database. Here a successful example can illuminate the basic methods to employ relational database to manage spatial data.

For the Penzhou cadastral information system, the author employed Mapinfo to manage partial spatial data and provide the ability of map interaction, and employed Sybase to manage the data of boundary nodes, the data of property parcels and all cadastral attributes. The system integration is realized by Visual Basic 6.0. All layers, which related to spatial data are stored in relational database, must have the "Uid" field as the same field in related Sybase tables. This method can keep Mapinfo map layers and the related spatial data stored in Sybase synchronous.

The access mode of database is multiform (e.g. ODBC, RDO and ADO etc.). Here the author employed ODBC to connect database and employed Mapinfo’s OLE function to make a reality of map interaction. In fact, the above two key techniques involve a series of map object operations (e.g. inserting, deleting, updating, keeping map object and its spatial storage data synchronous etc.). Trigger, transaction and secure recovery of relational database are useful and efficient on keeping the consistency and integrality of database.

1) Transaction

To connect the database and initialize the Mapinfo’s OLE object are two kinds of necessary work when initializing the system. Connections among VB, Sybase and Mapinfo will be constructed after the above two work, and then VB, Sybase and Mapinfo will be able to communicate with each
other. Here the author will only illustrate the system working principle with the operation of inserting a boundary node, and the rest map operations are just the same. (Here we assume the public variable of database connection as “cn” and the public object variable of Mapinfo OLE as “Mapinfo”.)

```
Dim ssql as string
cn.BeginTrans
ssql = “insert jzd values(’”
 cn.execute ssql
if InsertAPointIntoMapinfo(’” then
    /* The function can insert a point into the layer of boundary nodes, if succession return true, otherwise return false. */
    cn.CommitTrans
    Mapinfo. do “commit table < the map layer of boundary nodes>”
Else
    cn.RollbackTrans
    Mapinfo. do “rollback table < the map layer of boundary nodes>”
End if
```

2) Application of database trigger

Database trigger is useful to keep database consistency and can promote the system efficiency. Here two triggers are involved in. One of them will trigger to update the related records in “Zdindex” table whenever a record in “jzd” table needs to be updated, and the other will trigger to delete related records in “Zdindex” table whenever a record in “zd” table needs to be deleted.

- Trigger related to update a record in “jzd” table

```
CREATE TRIGGER dbo. Update-ZDindex-bh
ON dbo. jzd
FOR UPDATE AS
BEGIN
    declare @Jid char(20)
    declare @Uid int
    if @@rowcount = 0
        return
    select @Uid = Uid from inserted
    select @Jid=Jid from inserted
    update ZDindex set JZDjid = @Jid
    where JZDuid = @Uid
END
```

- Trigger related to delete a property parcel

```
CREATE TRIGGER dbo. DeleteRelatedZDindex
ON dbo. zd
FOR DELETE AS
BEGIN
    delete ZDindex where ZDuid in (select Uid from deleted)
END
```

3) Secure recovery

Because of all kinds of causes (e.g. medium failure etc.) any database is possibly damaged and crashed in process of work, at that time the function of database secure recovery will be employed to recover the damaged database. The ability of database secure recovery must be paid more attention to when designing database system. Generally, there are three methods to achieve the database recovery, which are disk mirror, database backup and online database log. In the process of database design, the ability of system recovery and the efficiency of system work must be planned carefully, and one of them will degrade another. For example, disk mirror can provide online database recovery, but this will lead to promote I/O access and depress the system efficiency at the same time. How to keep the ability of system recovery and the work efficiency of system on an appropriate level is a considerable problem when designing a database. In the process of designing Penzhou cadastral management system, the above three recovery styles are employed to keep data security and reliability because of the importance of cadastral data and the need of net office.

5 Conclusion

As the above presented, it is actually feasible to employ relational database to manage cadastral spatial data. It has obvious merits compared with conventional storage style based on file or files. These advantages involve data security, data transmission, data share, transaction and data recovery etc. The storage structure of partial spatial data based on relational database depresses the need of GIS platform dependency. The storage structure of purely relational database will be completely independent of
GIS platform. A satisfactory result had been gotten when the storage structure of partial spatial data based on relational database was employed to develop Penzhou cadastral management system.

References

1. Zhuang T H. The open C/S structure and heterogeneous database interoperation on Sybase. *Microcomputer Applications*, 1996(11):73~77 (in Chinese)
2. Tang M. The integration and implementation of intranet information system. *Computer Applications*, 1999(10):68~69 (in Chinese)
3. Wang W A, Wang X Y. Employ C/S database to store GIS map data and its attribute data. *Microcomputer Applications*, 1999(6):7~9 (in Chinese)
4. Zhu Y. An attempt to manage spatial data with RDBMS. *Remote Sensing Information*, 1993(3):11~14 (in Chinese)
5. Tao B Z. The technology handbook of sybase [M]. Beijing: Science Publishing House, 1997 (in Chinese)
6. Gong J Y. Some theories and technology about GIS in the contemporary era [M]. Wuhan: Publishing House of Wuhan technical University of Surveying and Mapping, 1999 (in Chinese)
7. Tom H. GIS database directions [A]. Geoinformatics’98 Conference Proceedings [C]. Beijing: China Association for GIS, 1998 (in Chinese)