DESIRABLE CHARACTERISTICS OF A SPATIAL DATA BASE MANAGEMENT SYSTEM

Stephen C. Guptill
U.S. Geological Survey
521 National Center
Reston, Virginia 22092

ABSTRACT

Just as rules have been developed to measure how well a data base management system conforms to the relational model, the desirable characteristics of a spatial data base management system can be specified. Spatial data base management systems must meet the requirements of conventional data base management systems as well as provide special facilities to handle spatial data. Characteristics such as the independent handling of feature, attribute, topology, and coordinate data and the support for alternative geometric representations are desired. This set of characteristics serves not only as criteria for evaluating existing systems, but also as input for future system design.

INTRODUCTION

The U.S. Geological Survey has collected, analyzed, and disseminated geologic, hydrologic, and cartographic information of the Nation for over 100 years. While most of this information has been provided in map and tabular form, an increasing amount of information is now being collected and stored in digital data sets that can be accessed and manipulated by computers. Although surveys, imagery, and maps form the basis of cartography, other spatial data can add useful information to the data base. Street addresses and geographic names are examples of other data used to refer to spatial entities. Items that are mappable, but not commonly shown on conventional maps (for example, geologic drill holes), and the information they convey need to be accommodated in a spatial data base model of geographic reality. Such data models typically consist of four major components: geometric descriptions of the spatial entities (objects), descriptive attributes for the objects, the topological relationships between objects, and feature relationships describing which objects comprise features.

Once spatial phenomena have been accurately and appropriately structured for a given model, the challenge remains for the user to extract and manipulate data and to display the resulting information in a manner relevant to a specific problem or application. Data may need to be extracted based on combinations of its geometry, topology, and attribute fields. Specifically, a spatial data base should be designed to meet the requirements of users who wish to define application-dependent sets of information, and spatial data base management systems need to provide the necessary functionality to meet these requirements.

Publication authorized by the Director, U.S. Geological Survey. Any use of trade names and trademarks in this publication is for descriptive purposes only and does not constitute endorsement by the U.S. Geological Survey.

THE DATA BASE ADVANTAGE

Most of the systems that have been developed to perform an automated cartographic or a geographic information system (GIS) type of processing have not utilized data base management system (DBMS) facilities. Viewed in the abstract, DBMS technology provides significant desirable characteristics for handling spatial data. These advantages include capabilities to store interrelated data without harmful or unnecessary redundancy to serve one or more applications in a optimal fashion; to make applications programs independent of the underlying data; and to provide a common and controlled approach to add, modify, and retrieve data from the data base (Martin, 1975). The capabilities offered by relational DBMS's to respond to ad hoc queries would seem to make them ideal for handling spatial data. However, only recently has DBMS software been used to handle spatial data (for example, Morehouse, 1985; Palimaka and others, 1986). Why has it taken over a decade to develop systems with some spatial data base management capability? Is the data base advantage no advantage at all when it comes to handling spatial data? The contention here is that the DBMS environment is fundamentally the preferred environment and that the special requirements for handling spatial data have delayed the development of such systems.

SPATIAL DATA PROCESSING ENVIRONMENTS

Systems that perform automated cartography or GIS types of functions need to handle large amounts of geometric data. This fact alone sets the spatial data processing environment apart from other DBMS environments. The high-level functions performed for automated cartography or GIS's are identical (Guptill, 1985). However, the operating environments in which those functions are applied are different for automated cartography and GIS, and thus impose different requirements on a spatial DBMS.

Automated cartography activities are now being performed on stand-alone workstations with independent computer processors, memory, and disk storage. Activities such as data collection, editing, symbolization, and display are performed on a data file that represents the spatial information for a given geographic area (for example, a 1:24,000-scale map sheet). The work is performed as an independent operation; data being processed is not accessed by others. Rapid system response time is a key to efficient production operations. Thus, efficient access to the spatial data becomes a key requirement for a spatial DBMS. However, inefficient implementations of spatial access mechanisms have caused some vendors to continue to forgo utilizing DBMS technology in favor of internal memory and file management operations. This solution does not help to satisfy the additional DBMS requirements imposed by GIS's.

GIS operations may take advantage of the same types of hardware as automated cartography systems, but a key difference in operations is the need for GIS's to support multiple-user access to a central spatial data base. Features such as security, integrity, synchronization, and recovery take on added importance in this environment. Support of multiple versions of a set of relations may also be required.

EXTENSIONS REQUIRED FOR SPATIAL DATA HANDLING

A major key to creating a spatial DBMS is support of spatial data types as an integral part of the DBMS. Extensions of the relational model to support points, lines, and polygons should overcome a number of the problems that exist in trying to use general-purpose data base systems today. Some researchers

(Waugh, 1986) have worked around this problem by storing coordinate data in various data types that allow for the bulk storage of up to 64,000 bytes (such as the LONG data type in ORACLE or the SEGMENTED STRING data type in VAX Rdb/VMS).

While these solutions allow for the integrated management of all the data, they fall short in the second major area, the extensions to the model to add a set of operators for these data types. One set of such operators has been defined (Claire and Guptill, 1982) and other sets are possible. Clearly this problem area is not limited to cartography and GIS, but is germain to various engineering applications such as CAD/CAM. Some of the required extended semantics for the relational model are described in detail by Stonebraker 1986.

The third major extension concerns the expression of spatial queries. Questions such as "find all the schools in Fairfax County that are further than 5 miles from a fire station" would be difficult to express in a query language such as SQL. Presumably the addition of spatial operators would eliminate some of these difficulties by adding such terms as "spatial intersection" to the SQL language. The handling of spatial queries is an area that would benefit from some type of natural language query parser. The addition of query capability using graphic displays and a mouse also would be of assistance to cartography and GIS users.

PERFORMANCE CONSIDERATIONS

The previous section has mentioned some of the basic extensions that could be made to relational DBMS's to allow them to handle spatial data. However, even if these extensions are implemented, it does not necessarily mean that the system will have an acceptable level of performance. Assuming that the spatial operators are implemented efficiently, there are two other areas that would appear to be key to system performance: query optimization and spatial search mechanisms.

The purpose of a query optimizer is to choose an efficient strategy for evaluating a given relational expression. The optimization process has been described in terms of four stages: (1) cast the query into some internal representation; (2) convert to canonical form; (3) choose candidate low-level procedures; and (4) generate query plans and select the cheapest (Date, 1986). Each of these stages will be complicated by the addition of spatial data types and spatial operators. For example, if spatial operators are treated as a series of restrictions on a given relation, an optimizing strategy that executes all restrictions before executing joins may be inappropriate. A more appropriate sequence may be relational restrictions, relational joins, and then spatial restrictions. Optimization routines also utilize information about the current state of the data base, including the existence of indexes. Acceptable performance for spatial data handling will probably hinge on an efficient spatial search and indexing technique.

A number of researchers have been investigating various one-dimensional access structures for use with spatial data. These include QUAD trees, K-D trees, grid files, Peano keys, R-trees, and K-D-B trees (see Kleiner and Brassel, 1986; Guttman and Stonebraker, 1983). Implementation and test results for the various methods are not generally available although some success has been reported using a variation of a Peano key index within GEOVIEW (Waugh, 1986). Rapid spatial retrieval also depends on an effective physical design of the data base. Issues such as buffer management and physical clustering of coordinate data on disk may be critical in engineering an efficient system.

SUMMARY AND CONCLUSIONS

The major desirable characteristics of a spatial data base management system have been presented. Spatial data base management systems should provide the functions of existing relational data base management systems, as well as special facilities to handle spatial data. The handling of spatial data will require the definition of spatial data types and operators, design of spatial indexing, query formulation and optimization strategies, and engineering of efficient access to mass storage devices.

REFERENCES

Claire, R.W., and Guptill, S.C.,1982, Spatial Operators for Selected Data Structures: Auto-Carto V, Arlington, Va, Proceedings, August 22-28, 1982, p. 189-200.

Date, C.J., 1986, An Introduction to Database Systems: Reading, Pa., Addison-Wesley Publishing Co., p. 333-359.

Guptill, S.C., 1985, Functional Components of a Spatial Data Processor: Auto-Carto 7, Washington, D.C., Proceedings, March 11-14, 1985, p. 229-236.

Guttman, Antonin, and Stonebraker, Michael, 1983, R-Trees: A Dynamic Index Structure for Spatial Searching: Memorandum No. UCB/ERL/ M83/64, College of Engineering, University of California, Berkeley, 28 p.

Kleiner, Andreas, and Brassel, Kurt, 1986, Hierarchial Grid Structures for Static Geographic Data Bases: Auto Carto London, September 14 19, 1986, Proceedings, p. 485-496.

Martin, James, 1975, Computer Data-Base Organization: Englewood Cliffs, N.J., Prentice-Hall, Inc., p. 17.

Morehouse, Scott, 1985, ARC/INFO: A Geo-Relational Model for Spatial Information: Auto-Carto 7, Washington, D.C., March 11-14, 1985, Proceedings p. 388-397.

Palimaka, John, Halustchak, Orest, and Walker, Ward, 1986, Integration of a Spatial and Relational Database within a Geographic Information System: Technical Papers, ACSM-ASPRS Annual Convention, Vol 3., p. 131-140.

Stonebraker, Michael, 1986 ed., The INGRES Papers: Anatomy of a Relational Database System: Reading. Pa., Addison-Wesley Publishing Co., p. 313-392.

Waugh, Thomas, 1986, The GEOVIEW Design: Dept. of Geography, University of Edinburgh, 34 p.