CONSISTENCY OF SPATIAL DATABASE QUERY RESULTS

MAINGUENAUD Michel

France Telecom - Institut National des Télécommunications
9 rue Charles FOURRIER
F91011 Evry - France
+ 33 1 60 76 47 82
+ 33 1 60 76 47 80 (fax)
Michel.Mainguenaud@int-evry.fr

ABSTRACT. *Numerous applications involve the use of a Geographical Information System (GIS). A working session consists of several spatial database orders. The scope of this paper is to provide a link between alphanumeric data and spatial representation of objects stored in a GIS database. The notions of Granule, Topology and Set_relationship are defined to guarantee a semantics to alphanumeric data parts associated to the results of spatial database queries.*

INTRODUCTION

In current research toward the design of more powerful tools for urban planning, remote sensing different groups are simultaneously concentrating their work on Geographical Information System (GIS). GIS needs are very well known (Smith et al 1987). Nevertheless, several problems are still open. The need for spatial query languages has been identified. Spatial databases contain spatial and nonspatial data that users query in any possible combination. Conventional database query languages (i.e., SQL) allow users to describe which objects to retrieve from a database and how to display them in alphanumeric form. Spatial data have additional properties that users must also be able to address in a query language. Three types of instructions are distinguished. The actual user query specifying the retrieval of data to be displayed. Additional queries, called display queries, are necessary to separate query results into more detailed sets, each to be displayed in an individual format. The actual display description specifies how to render the data (Egenhoffer 1991). This paper deals with the first component. GIS users are not supposed to be database specialists. A working session is a set of database orders. Defining a nonsense query is therefore possible since the semantics of database query languages is not always very clear. The scope of this paper is to provide a link between alphanumeric data and spatial representation of objects stored in a database. The notions of Granule, Topology and Set_relationship are defined to guarantee a semantics to alphanumeric data parts associated to the results of spatial database queries.

The first part presents a toy database to illustrate these notions. The second part presents these notions. The third part presents the database modelling. The fourth part presents the application of these notions to a toy set of operators. The last part presents the conclusion.

A TOY DATABASE

Several database models have been proposed to manage geographical data. Some of them are based on an extended relational approach with Abstract Data Types (ADT) or a Non First Normal Form philosophy (Bennis et al 1990, Schek and Waterfeld 1986, Scholl and Voisard 1989, Vijlbrief and Oosterom 1992). Some of them are based on an Object-oriented paradigm (Banejeree 1987, Kemp 1990, Orenstein 1990). Some of them are based on rules and facts or graphs (Angelaccio et al 1990, Cruz et al 1987, Jungert 1984). Some of them are based on an algebraic approach (Frank 1982, Güting 1988, Svensson and Huang 1991, Sacks-Davis et al 1987). Whatever the given name (attributes, properties, etc.), basic geographical information is defined by a pair (name, domain). The names represent the identifiers and the domains represent the authorized values for such data.

Defining a new data model is out of the scope of this paper. To simplify the presentation without loss of generality, a toy database is defined with a relational approach (Ullman 1988) extended with an ADT for the spatial representation (i.e., the attribute Spatial_representation). Figure 1 presents a toy database. This database is used along this paper to illustrate the notion of Granule, Topology and Set_relationship.

Field	(Name, Fence_nature , Spatial_representation)
Town	(Name, Population , Spatial_representation)
Polluted_area	(Name, Pollution_type , Spatial_representation)
River	(Name, General_pollution, Border_pollution, Water_colour, Maximum_depth , Spatial_representation)

Figure 1 - A toy database

Several sets of spatial operators have been proposed to query geographical databases (i.e., Güting 1988, Egenhoffer 1989 for thematic data, Boursier and Mainguenaud 1992 for thematic and network data). Extending a DBMS with spatial operators implies to define a core of operators to fulfill spatial analysis requirements. Providing such a work is very difficult and sometime application-dependent. To simplify the presentation without loss of generality, we retain only three operators: the straight line, the inclusion and the intersection. They represent three classes of operators.

The first class provides a fictive object as a result. A spatial operator delivers a fictive object when the result does not correspond to a spatial reality. The straight line operator creates a fictive object representing the shortest line between two objects. This operator illustrates such a class.

The second class provides an already existing object as a result. A spatial operator delivers an already existing object when the result corresponds to a basic object defined in the conceptual database model. The inclusion operator illustrates such a class.

The third class provides a new object as a result. A spatial operator delivers a new spatial object when the spatial representation of the result is defined as a subset of the spatial representation of objects involved in this operator. The intersection operator illustrates such a class.

GRANULE, TOPOLOGY AND SET-RELATIONSHIP

The semantics of spatial database operators is defined on a particular attribute: "Spatial_representation." The properties of alphanumeric data must be defined about this attribute. We do not consider in this part the physical coordinates but the logical topology of spatial representations. To simplify the presentation without loss of generality, we consider a core of alphanumeric attributes. The attributes defined as a function (i.e., the density of population defined as a quotient of two attributes or functions) are not considered for the time. The spatial representation of a geographical object is defined by a boundary and an interior (Egenhoffer 1989). Alphanumeric data may be classified into three orthogonal categories: the Granule, the Topology and the Set_relationship.

Granule

The Granule defines the validity on the whole or on a subset of the spatial representation (without taking into account the notion of interior or boundary). This property is named Integrality or Subset. Let us use the attribute Maximum_depth and the attribute Name to illustrate the notion of Granule. The attribute Maximum_depth is a property defined for a river but no spatial representation has been associated in the data model. No guarantee is provided that the value of this attribute is still relevant for a subset of the spatial representation. Therefore, this attribute is classified as Granule:Integrality. In reverse, the value of the attribute Name (i.e., a key attribute in the relational data model) is still valid for a subset of the spatial representation. Therefore, this attribute is classified as Granule:Subset. This notion is close to the notion of inheritance defined in (Barrera and Buchmann, 1981).

Topology

The Topology defines the validity on the boundary, the interior or the entirely spatial representation (i.e., Global stands for boundary and interior) for a given attribute. Let us use the attributes Maximum_depth, Border_pollution and General_pollution to illustrate the notion of Topology. The attribute Maximum_depth is a property defined for the interior of a river. Therefore, this attribute is classified as Topology:Interior. The attribute Border_pollution is a property defined for the boundaries of a river. Therefore, this attribute is classified as Topology:Boundary. The attribute General_pollution is valid for the entirely spatial representation. Therefore, this attribute is classified as Topology:Global.

Set_relationship

The Set_relationship defines the possible overlap of two instances having the same semantics. Let us use the Town and Polluted_area objects to illustrate the notion of Set_relationship. Two polluted areas may have a physical spatial overlap. There is no physical reason for them to define a non-overlapping space division. The Polluted_area "object" is therefore classified as Set_relationship:Overlap. In the opposite side, two towns have an administrative reason to define a non-overlapping space division. The Town "object" is therefore classified as Set_relationship:Non_overlap.

DATA MODELLING

A good database design implies to define several properties of alphanumeric data. The notion of key, integrity constraints, referential constraints are examples of semantic information introduced in a relational database schema.
The properties of Granule, Topology and Set_relationship are also introduced in a database schema. The attribute level is concerned with the notion of Granule and Topology. The relation level is concerned with the notion of Set_relationship. These definitions are therefore introduced in the Data Definition Language (DDL) of a spatial database.

Attribute level

To explain the methodology of classification, let us consider a basic decomposition (i.e., without the concept of inheritance) of a geographical "object." A geographical object is defined from a conceptual point of view as a boundary and an interior. These basic components can be decomposed into a set of segments and a set of cells. The terms segment and cell are used as a metaphor (i.e., there is no link with a physical storage data model). To each step of decomposition a classification with a pair is associated. Let --> be the database aggregation constructor and -->> be the database set constructor (Figure 2).

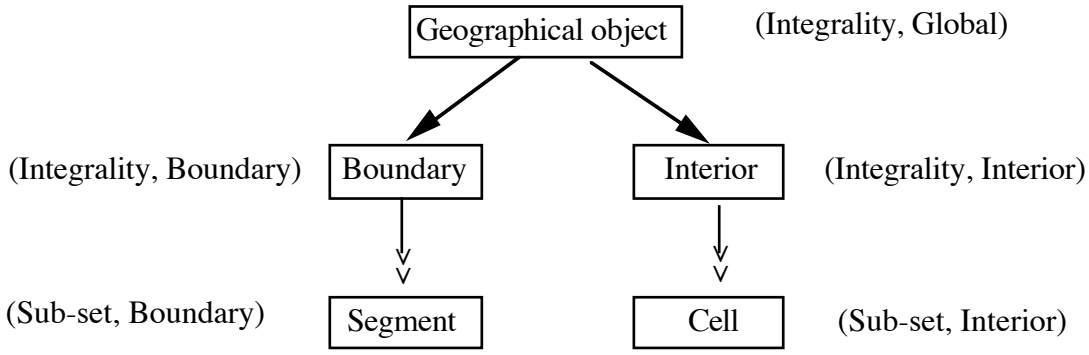

Figure 2 - Basic decomposition of a geographical object

The following rules are defined to classify an attribute A of a relation.

(Rule 1) Whenever the value of A is only relevant for the cell component:

The value associated to A is relevant for all the cells, the attribute A is classified (Subset, Interior). The attribute Water_colour of the relation River is an example.

The value associated to A is relevant for only part of the cells, is defined as a function of the cells or with an exogenous data, the attribute A is classified (Integrality, Interior). The attribute Maximum_depth of the relation River is an example. The maximum depth is known and stored in the database but the exact location is unknown.

(Rule 2) Whenever the value of the attribute A is only relevant for the segment component:

The value associated to A is relevant for all the segments, A is classified (Subset, Boundary). The attribute Fence_nature of the relation Field is an example.

The value associated to A is relevant for only part of the segments, is defined as a function of the segments or with an exogenous data, the attribute A is classified (Integrality, Boundary). The attribute Border_pollution of the relation River is an example. Due to the stream of a river, a general indicator of pollution was defined for a river (i.e., high, average, low). This indicator cannot be directly inferred from the database.

(Rule 3) Whenever the value of the attribute A is relevant for the cell component and for the segment component:

The value associated to A is relevant for all the cells and all the segments, the attribute A is classified (Subset, Global). The attribute Name of the relation River is an example.

The value associated to A is relevant for only part of the cells or for only part of the segments or is defined as a function of the cells and/or the segments or with an exogenous data, the attribute A is classified (Integrality, Global). The attribute General_pollution of the relation River is an example.

To synthesize the notions of Granule and Topology, Figure 3 presents a 2-D array with the attributes of relations River and Field.

Granule / Topology	Integrality	Subset
Global	General_pollution	Name
Boundary	Border_pollution	Fence_nature
Interior	Maximum_depth	Water_color

Figure 3 - The spatial properties of alphanumeric data

Relation level

The property of Set_relationship is important as soon as an aggregate function is involved in a spatial query. Let us use two queries (i.e., query Q_1 and query Q_2), as an example. Let query Q_1 be "Which are the national roads crossing towns with more than 100,000 inhabitants such as the total distance is less than 15 km?". Figure 4 presents a spatial configuration of a road crossing towns.

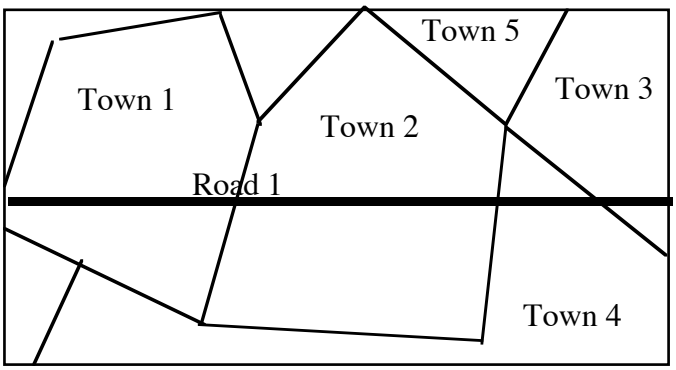

Figure 4 - Road crossing towns

Let query Q_2 be "Which are the national roads crossing polluted areas such as the total distance is less than 15 km?". Figure 5 presents a spatial configuration of a road crossing polluted areas.

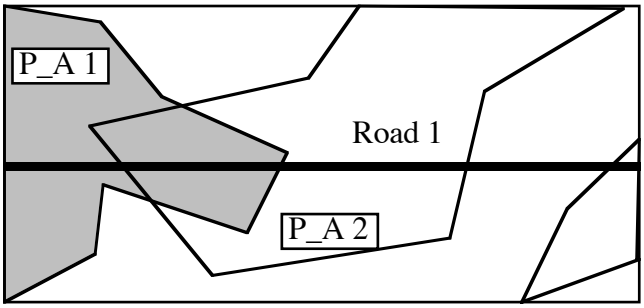

Figure 5 - Road crossing polluted areas

The semantics of query Q_1 and query Q_2 is the same (i.e., an intersection between two geographical objects). Defining a coherent extended SQL query language is out of the scope of this paper (i.e., the link between the Select clause and the Group by clause as soon as an operator appears in the Select clause, the link between an alphanumeric display and a graphical display as soon as the two kinds of information appear in the Select clause, etc.). Let us imagine an extended SQL language with the predicate of intersection (i.e., intersection) and the spatial operator of intersersection (i.e., Intersection).

The skeleton of an extended SQL statement for query Q_1 would be:

```
Select      Road.Name
From        Road, Town
Where       intersection (Road.Spatial_representation, Town.Spatial_representation)
            and Town.Population > 100 000
Group by    Road.name
Having      Sum ( length (
                  Intersection (Road.Spatial_representation, Town.Spatial_representation)
                )          ) < 15
```

Two basic requirements are defined for a query language. The first one is to provide the same skeleton of order for two queries having the same semantics. The second one is to free users from physical storage database model. Using the skeleton of query Q_1 to solve query Q_2, since the semantics is the same, would provide a wrong answer. The length of the intersection would be computed twice. To prevent such an error, the system should be able to detect this configuration and must be able to infer whether a polygon overlay is needed. The user is not concerned with the data model used to stored the alphanumeric data or spatial representation (i.e., the user has not to convert an integer into a float to be able to define a multiplication between two attributes in the Select clause). The spatial query is defined without any knowledge of the physical storage database model. The notion of Set_relationship allows the system to infer spatial operators linked to the physical data model (i.e., polygon overlay).

To synthesize the notion of Set-relationship notions, Figure 6 presents a 2-D array with the relations Polluted_area, Town and River.

Set-relationship	Polluted_area	Town	River
Overlap	X		
Non_overlap		X	X

Figure 6 - The Set-relationship

Database definitions

The properties of Granule, Topology and Set_relationship must be handled by the DBMS to guarantee the consistency of spatial query results. Therefore, the Data Definition Language (DDL) must be extended with these properties. Defining a new DDL is out of the scope of this paper. Let us use a SQL-like DDL. Figure 7 presents an example of such an extended DDL statement for the definition of the relation River.

```
Create Non_overlap table River
    (   Name : string, key, (Subset, Global),
        General_pollution : pollution_domain, (Integrality, Global),
        Border_pollution : pollution_domain, (Integrality, Boundary),
        Water_color : color_domain, (Subset, Interior),
        Maximum_depth : integer, (Integrality, Interior),
        Spatial_representation : spatial_domain
    );
```

Figure 7 - A DDL statement for the relation River

APPLICATION TO SPATIAL OPERATORS

A geographical "object" is an aggregation of spatial data and alphanumeric data. A GIS working session is composed of a set of database orders. These orders may involve several spatial operators. These operators work on a specific attribute (i.e., Spatial_representation). It is widely recognized that spatial operators must be closed on the spatial domain. This implies that the results of a spatial operator must be a geographical "object". This very pretty property allows to combine spatial operators (Güting 1988, Kirby and Pazner 1990, Mainguenaud 1993). The result of a spatial operator is therefore an aggregation of spatial data and alphanumeric data. Computational geometry and the specifications of a spatial operator provide the spatial data part of a result. The user or the application is in charge of providing the alphanumeric data part.

A high level query language for user interface must provide a very reduced number of operators to be accepted. These operators are defined as a combination of lower level operators. Several hidden database operations are performed to provide the end-user's result of a spatial query. An automatic generation of alphanumeric data part guarantees the consistency of high level spatial operator results since several spatial database orders may have been performed. Two levels of detail can be defined to build the alphanumeric data part. The first level, named semantic level, provides a general interpretation of a result (i.e., without taking into account the spatial representations). The second

level, named complete level, provides a more accurate interpretation of a result (i.e., taking into account the spatial representations). This paper deals with the semantic level since the complete level is highly dependent of the specifications of a spatial operator. To simplify the presentation, without loss of generality, let us consider the spatial operators as binary operators (i.e., defined between two objects O_1 and O_2). Let us name A_1 (resp. A_2) the set of attributes defined in the alphanumeric database model of object O_1 (resp. O_2).

The notions of Granule, Topology and Set_relationship allow an automatic generation of alphanumeric data part for the three classes of operators (i.e., providing a fictive object, providing an already existing object and providing a new object).

Operator providing a fictive object

The straight line operator illustrates the operators providing a fictive object as a result. The semantics of the straight line operator is to provide an object such as the spatial representation is the shortest line starting from the boundary of object O_1 to the boundary of object O_2. From a conceptual point of view, all the attributes of O_1, classified as Granule:Integrality, are not relevant since the result is built with a subset of O_1. All the attributes, classified as Topology:Interior, are not relevant since the result is built with a subset of the boundaries. All the attributes, classified as Topology:Global, are not relevant since the result is built with a subset of O_1. All the attributes, classified as Granule:Subset and Topology:Boundary, are relevant since the result is built from the boundary of O_1 to the boundary of O_2. Nevertheless, the classification in the result is now Granule:Integrality. Figure 8 presents the construction of the alphanumeric data part of the result from object O_1. The lines define the classification for the notion of Topology. The columns define the classification for the notion of Granule. The elements of the array define: first, the relevance of the attributes with such a classification; second the classification of this attribute in the data model associated to the result. The straight line operator is symmetric. A similar figure can be obtained using the object O_2. The data model of the result is built with the relevant attributes of O_1 and the relevant attributes of O_2.

Granule Topology	Integrality	Subset
Global	non relevant	non relevant
Boundary	non relevant	Granule:Integrality, Topology:Boundary
Interior	non relevant	non relevant

Figure 8 - Straight line operator: the alphanumeric data from O_1

Operator providing an already existing object

The inclusion operator illustrates the operators providing an already existing object as a result. This operator is non-symmetric. Let us suppose the inclusion operator, defined as $\subseteq (O_1, O_2)$, provides the included object (i.e., O_1) as a result. The semantics of the inclusion operator is to provide an object such as the spatial representation is identical to O_1. From a conceptual point of view, all the attributes of O_1 are still relevant. The attributes of O_2, classified as Granule:Integrality, are not relevant since the spatial representation of the result (i.e., O_1) is a subset of the spatial representation of O_2. The attributes of O_2, classified as Granule:Subset and Topology:Interior, are relevant for the result since the spatial representation of O_1 is a subset of the spatial representation of O_2. Figure 9 presents the construction of the alphanumeric data part of the result from object O_2. The semantics of this figure is similar to the semantics of Figure 8. The data model of the result is built with the attributes of O_1 and the relevant attributes of O_2.

Granule Topology	Integrality	Subset
Global	non relevant	non relevant
Boundary	non relevant	non relevant[1]
Interior	non relevant	Granule:Subset, Topology:Interior

Figure 9 - Inclusion operator: the alphanumeric data from O_2

A similar figure can be obtained if the result is defined as the inclusive object (i.e., O_2). All the attributes of O_2 are relevant. All the attributes of O_1 are relevant since the spatial representation of O_1 is included in the spatial representation of O_2. Nevertheless, the classification is now Granule:Integrality, Topology:Global. No information is provided on the exact location of the spatial representation of O_1 in the spatial representation of O_2. The semantics of Figure 10 is similar to the semantics of Figure 8. This figure presents the construction of the alphanumeric data part of the result from object O_1.

[1] This combination (Granule:Subset, Topology:Boundary) is an example such as the result of the complete level may be different from the semantic level. The spatial configuration may require the introduction of the attributes classified as (Granule:Subset, Topology:Boundary).

Granule Topology	Integrality	Subset
Global	Granule:Integrality, Topology:Global	Granule:Integrality, Topology:Global
Boundary	Granule:Integrality, Topology:Global	Granule:Integrality, Topology:Global
Interior	Granule:Integrality, Topology:Global	Granule:Integrality, Topology:Global

Figure 10 - Inclusion operator: the alphanumeric data from O_1

Operator providing a new object

The intersection operator illustrates the operators providing a new object as a result. The set union of A_1 and A_2 is not relevant to be the data model of the result. The semantics of the intersection operator is to provide a new object built from the common spatial parts of the objects involved in this operator. From a conceptual point of view, the attributes classified as Granule:Integrality cannot be relevant for the result. The attributes, classified as Granule:Subset, are relevant. Nevertheless, the attributes classified as Granule:Subset and Topology:Boundary cannot still be classified as Granule:Subset since the result is built from parts of the two objects. The classification is now Granule:Integrality. Figure 11 presents the construction of the alphanumeric data part of the result from object O_1. The semantics of Figure 11 is similar to the semantics of Figure 8. The intersection operator is symmetric. A similar figure can be obtained using O_2. The data model of the result is built with the relevant attributes of O_1 and the relevant attributes of O_2.

Granule Topology	Integrality	Subset
Global	non relevant	Granule:Subset, Topology:Global
Boundary	non relevant	Granule:Integrality, Topology:Boundary
Interior	non relevant	Granule:Subset, Topology:Interior

Figure 11 - Intersection operator: the alphanumeric data from O_1

CONCLUSION

The number of applications using a Geographical Information System (GIS) is considerable. Therefore, it is of prime importance to offer a powerful database modelling tool. Several database models have been proposed to capture the semantics of geographical data. This paper proposes three notions (i.e., Granule, Topology and Set_relationship) to be introduced into a database model to capture more semantics. These notions provide a link between alphanumeric data and spatial data. They guarantee a spatial consistency to alphanumeric data associated to the result of a spatial operator. These extensions imply news rules to construct the results of spatial queries (i.e., the relational projection operator in the context of an extended relational DBMS).

References

Angelaccio M., Catarci T., Santucci G., 1990. QBD* : A Graphical Query Language with Recursion, IEEE Transaction on Software Engineering, Vol 16, n°10, October 1990, 1150-1163

Banejeree J.,1987. Data Model Issues for Object-oriented Applications, ACM Transaction on Office Information System, Vol 5, January 1987, 3-26

Barrera R., Buchmann A., 1981. Schema Definition and Query Language for a Geographical Database System, Transactions on Computer Architecture: Pattern Analysis and Image Database Management, Vol 11, 250-256

Bennis K., David B., Quilio I., Viémont Y.,1990. GéoTropics: Database Support Alternatives for Cartographic Applications, 4th International Symposium on Spatial Data Handling, Zurich, Switzerland, July 1990

Boursier P., Mainguenaud M., 1992. Spatial Query Languages : Extended SQL vs. Visual Languages vs. Hypermaps, 5th International Symposium on Spatial Data Handling, Charleston, USA, August 1992

Cruz IF, Mendelzon AO, Wood PT, 1987. A Graphical Query Language Supporting Recursion, ACM SIGMOD Conference, San-Fransisco, USA, May 1987

Egenhofer M. J., 1989. A Formal Definition of Binary Topological Relationships, 3rd International Conference on Foundations of Data Organization and Algorithms, Paris, France, June 1989

Egenhofer, M. J., 1991. Extending SQL for Graphical Display. Cartography and Geographic Information Systems, Vol 18, n° 4, 1991, 230-245

Frank A., 1982. Mapquery : Database Query Language for Retrieval of Geometric Data and their Graphical Representation, Computer Graphics, Vol 16, n°3, July 1982, 199-207

Güting RH , 1988. Geo-Relational Algebra : A Model and Query Language for Geometric Database System, International Conference on Extending Data Base Technology, Venice, Italy, March 1988

Jungert E., 1984. Inference Rule in a Geographical Information System, IEEE Workshop on Language for Automation, New-Orleans, USA, November 1984

Kemp Z.,1990. An Object-Oriented Model for Spatial Data, 4th International Symposium on Spatial Data Handling, Zurich, Switzerland, July 1990

Kirby KC, Pazner M., 1990. Graphic Map Algebra, 4th International Symposium on Spatial Data Handling, Zurich, Switzerland, July 1990

Mainguenaud M., 1993. From the User Interface to the Database Management System : Application to a Geographical Information System, 5th International Conference on Human Computer Interaction, Orlando, USA, August 1993

Orenstein JA, 1990. An Object Oriented Approach to Spatial Data Processing, 4th International Symposium on Spatial Data Handling, Zurich, Switzerland, August 1990

Sacks-Davis R., Mc Donnell KJ, Ooi BC, 1987. GEOQL : A Query Language for Geographical Information Systems, Australian and New Zealand Association for the Advancement of Science Congress, Townsville, Australia, August 1987

Schek HJ, Waterfeld W., 1986. A Database Kernel for Geoscientific Applications, 2nd International Symposium on Spatial Data Handling, Seattle, USA, June, 1986

Scholl M., Voisard A., 1989. Thematic Map Modelling, 1st International Symposium on Large Spatial Databases, Santa-Barbara, USA, July, 1989, Lecture Notes in Computer Science, Springer-Verlag, n° 274

Smith TR, Menon S., Star JL, Ester JE, 1987. Requirements and Principles for the Implementation and Construction of Large Scale GIS, International Journal of Geographical Information Systems, Vol 1, n°1, 13-31

Svensson P., Huang Z., 1991. Geo-SAL: A query Language for Spatial Data Analysis, 2nd Large Spatial Databases Conference, Zurich, Switzerland, August 1991, Lecture Notes in Computer Science, Springer-Verlag, n°525

Vijlbrief T., Van Oosterom P., 1992. The Geo++ System : An Extensible GIS, 5th International Symposium on Spatial Data Handling, Charleston, USA, August 1992

Ullman JD, 1988. Principles of Database and Knowledge-base Systems, Computer Science Press, Maryland