Spatial Operations

Anda VELICANU
Economic Informatics Department, Academy of Economic Studies
Bucharest, ROMANIA
anda.velicanu@ie.ase.ro

This paper contains a brief description of the most important operations that can be performed on spatial data such as spatial queries, create, update, insert, delete operations, conversions, operations on the map or analysis on grid cells. Each operation has a graphical example and some of them have code examples in Oracle and PostgreSQL.
Keywords: *Spatial operations, Querying operations, Spatial data.*

1 Introduction

The values of the objects' spatial attributes represent the spatial data [1].

Spatial data can be divided in point data and regional data. The point data is a point which is completely characterized by its location in a multidimensional space. It may come directly from measurements or by transforming in order to be more easily stored and retrieved [2]. The representation of spatial data in Oracle is done according to the ANSI standard.

Spatial operations are functions that form important components of a spatial data model. They allow the storage of input data, their analysis and obtaining new data as output information. There are special spatial operations that depend as implementation of the operators made available by the database management system, in which the data is stored. No matter who the producing company is, there are certain operations that are common to any spatial products and that are based on queries. There are also regular operations that can be found in other models' data, such as create, insert, update, delete.

Intuitively, spatial operations represent different aspects from the same real world operation. Therefore, spatial operations have space-invariant properties based on which they can be describe [3].

The user views a system as an extended relational database management system. In addition to the usual attributes in relations, the user can define spatial attributes in a homogeneous way. The system supports four types of spatial entities: points, line segments, polygons, and regions. In addition to the standard SQL commands, spatial operations are augmented in order to process and query spatial data. Spatial attribute data is stored in suitable spatial data structures [4].

2 Querying operations

Query processes are generally made to select specific data in order to measure the information from data and to perform some calculations. The queries performed on spatial data can be: Local; Area; Neighborhood (the most difficult because it requires the evaluation of proximity – eg. determining the road which passes closest to a specified region).

There are two important categories of special spatial operations, which satisfy the retrieval requests and are very popular: operations that determine the spatial relations and spatial analysis operations.

Consider a and b, two spatial objects of the same type or not (points, lines, polygons, circles, etc.), that are represented the same system (x0y axis system, latitudinally – longitudinally system, a proper representation system etc.). With these spatial operations we can:

1. Find the equality between two spatial objects a = b, which means that the two spatial objects have only common points in the system they are represented.

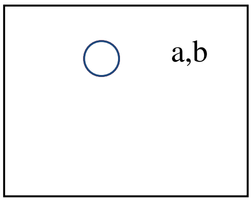

Fig. 1. Equality between two spatial objects

2. Find the disjunction between two spatial objects $a \cap b = \varnothing$, which means that the two spatial objects have no points in common.

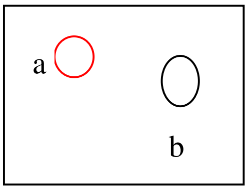

Fig. 2. Disjunction between two spatial objects

3. Calculate the minimum distance between two spatial objects Dist(a,b) = No, where No is calculated as the minimum distance between the polygons which minimum frame a and b objects

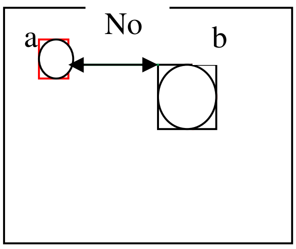

Fig. 3. The minimum distance between two objects

4. Obtain the union of geometries (spatial objects) aUb = c, where c is a spatial object that contains only once all points of a and b objects.

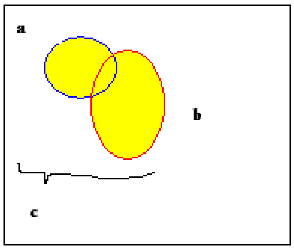

Fig. 4. The union of two spatial objects

5. Obtain the intersection of two or more geometries $a \cap b =$ c, where c is an intersection geometry, which all the common points of a and b geometries.

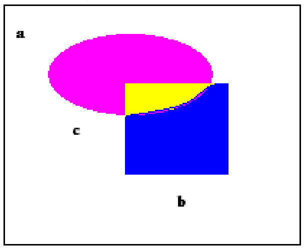

Fig. 5. The intersection of two spatial objects

6. Determine the closest spatial object Close(a) = b, if Dist(a,b) is minimum, no matter where the b object is situated in the same system as a object.

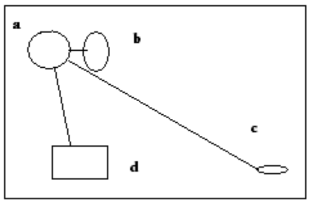

Fig. 6. Determining the closest spatial object

7. Calculate one object's area A(a) = No, where No is the area of a object.

8. Calculate the perimeter of an object P (a) = No, where No is the sum of all sides of the object

9. Calculate the centriod of an object C (a) = p, where p is the central point of object a

10. Calculate the closest geometric area CloseArea (a) = A (b), where Close (a) = b

11. Determine the buffer zone around an object

Buffer (a, No) = {b, c, d...}, where b, c, d ... is the set of points in the vicinity of the a object, at a maximum distance equal to No

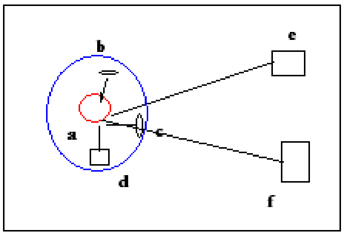

Fig. 7. Determining the buffer area around an object

12. Determine the way two objects relate RELATE(a,b) = true, if objects a and b have common points on the edges or inside them.

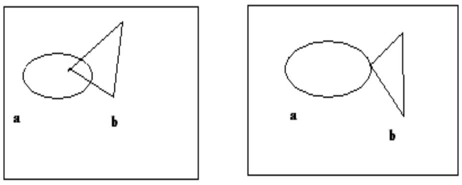

Fig. 8. Example relations between spatial objects

Determine the inclusion of a spatial object in another spatial object $a \subset b$ = true, if all a's points are contained in object b.

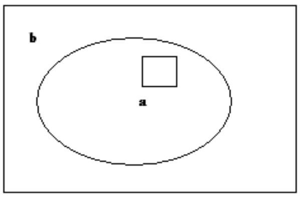

Fig. 9. Inclusion of a spatial object in another spatial object

13. Determine the touch of two objects Touch(a,b) = true, if $a \cap b = \varnothing$ and Dist(a,b) = 0

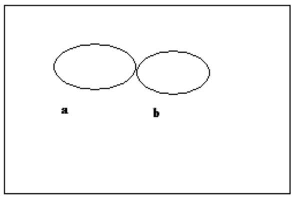

Fig. 10. The touch of two spatial objects

In order to exemplify I can briefly explain the features of spatial operations implemented in Oracle and PostgreSQL. The spatial component of Oracle is called Oracle Spatial and allows the implementation of following special operators:
- SDO_RELATE operator, which evaluates the topologic criterion (determines if two spatial objects interact);
- SDO_WITHIN_DISTANCE operator, which determines whether two spatial objects are from each other at a distance;
- SDO_INTERSECTION operator, which determines the topological intersection of two spatial elements;

An example of implementing this operator is given below, adapted from the book [5].

```
SELECT
SDO_GEOM.SDO_INTERSECTION(eg1.geom,eg2.geom, 0.005) as intersection FROM deposits eg1, deposits eg2 WHERE eg1.ID='1' AND eg2.ID='2';
```

- SDO_AREA operator, which calculates the area of a geometric figure;

An example of implementing this operator is given below, adapted from the book [5].

```
SELECT eg.name deposit, SDO_GEOM.SDO_AREA(eg.geom, 0.005) area FROM deposits eg;
```

- SDO_MAX_MBR_ORDINATE operator, which determines the maximum value for coordinates (x or y);
- SDO_LENGTH operator, that calculates the perimeter of geometric figures;
- SDO_DIFFERENCE operator, which determines the geometry resulting from the difference of two spatial objects;

- SDO_CENTROID operator, which determines the center of a polygon;

The spatial component of PostreSQL is called PostGIS and allows the implementation of following special operators:
- ST_INTERSECTION operator, which returns a geometry that is the common for two spatial elements;
- ST_AREA operator, that returns the calculated area of a geometric figure;
- ST_LENGTH operator, that calculates the perimeter of geometric figures;
- ST_DIFFERENCE operator, which returns a geometry representing a part of a spatial object that does not intersect another object;
- ST_UNION operator, which returns a geometry that represents the common part of two spatial objects;
- ST_GEOMETRYTYPE operator, that returns the geometric type of an object.

3 Create operations

Such operations generate new geometric objects. An example would be generating a buffer, which describes polygons at a specific distance from points, lines or areas.

4 Insert operations

This type of operations allows inserting data in the system.

5 Update operations

These operations change the characteristics of spatial objects.

6 Operations on the map

Such functions are used most often to operations that change a map's scale. Thinning the coordinates of a line means to reduce the number of coordinates that define a line. Similarly it can be applied to a polygon. Edge matching is bonding the maps and arranging them in a single unitary map.

7 Conversions

Conversions refer to rasterizing (transforming vector data into raster data) and vectorizing processes (transforming raster data into vector data).

8 Analysis of grid cells

This type of operation applies only to raster data. Handling grid cell consists of examining one cell or a combination of cells. Basic operations are: operations on a cell, matching operations on two cells and studying the neighbors of a cell. There can be performed different functions: trigonometric, exponential, statistical.

9 Delete operations

Involve the removal of objects from the system.

Aknowledgement

This article is a result of the project „Doctoral Program and PhD Students in the education research and innovation triangle". This project is co funded by European Social Fund through The Sectorial Operational Program for Human Resources Development 2007-2013, coordinated by The Bucharest Academy of Economic Studies.

References

[1] Velicanu A., Olaru S., "Optimizing spatial databases", *Revista de Informatică Economică*, Vol. 14, Nr. 2 / 2010, pag. 61-71, ISSN 1453-1305.

[2] Velicanu M., Lungu I., Muntean M., Ionescu S., "Sisteme de baze de date – Teorie şi practică", Editura Petrion, Bucureşti, 2003, pg. no. 339, ISBN 973-9470-70-X.

[3] Karimipour F., Delava M. Frank A., "An Algebraic Approach to Extend Spatial Operations to Moving Objects", *World Applied Sciences Journal 6 (10):* 1377-1383, 2009, ISSN 1818-4952.

[4] Aref W., Samet H., "Extending a DBMS with Spatial Operations", *Advances in Spatial Databases{2nd Symposium, SSD'91*, vol. 525 of Springer-Verlag Lecture Notes in Computer Science, August 1991, pg. 299-318, Zurich, Switzerland, ISBN 3-540-60153-8.

[5] Velicanu M., Lungu I., Botha I., Bâra A., Velicanu A., Rednic E., "Sisteme de baze de date evoluate", Editura ASE, 2009, nr. pg. 430, ISBN 978-606-505-217-8.