

FIELDS AND OBJECTS ALGEBRAS FOR GIS OPERATIONS¹

GILBERTO CÂMARA, UBIRAJARA MOURA FREITAS,
Image Processing Division,
National Institute for Space Research (INPE)

MARCO ANTÔNIO CASANOVA,
Latin American Center for Solutions in Higher Education and Research,
IBM Brazil.

RESUMO

Este trabalho discute o problema da definição formal das operações em sistemas de informação geográfica. Os dados geográficos são divididos em duas classes: geo-objetos e geo-campos, o que reflete a natureza dual (discreto-continuo) das representações da realidade geográfica. Estudam-se as operações sobre campos, geo-objetos e as transformações entre geo-campos e geo-objetos. A análise deste trabalho serve de base para LEGAL, uma linguagem espacial utilizada no sistema SPRING, desenvolvido pelo INPE com apoio da IBM Brasil e da EMBRAPA.

ABSTRACT

This paper addresses the problem of formal definition of the operations on geographical information systems (GIS). Geographical data is divided in two main classes: geo-objects and geo-fields, which portray discrete and continuous representations of reality. We study the operations over geo-fields, geo-objects, and the transformations between geo-fields and geo-objects. This analysis has been used as the basis for LEGAL, a general spatial language, which is used in the SPRING GIS, developed by INPE, with support from IBM Brazil and EMBRAPA.

1. Introduction

This work discusses the nature of the operations performed on geographical information systems (GIS), based on a formal model of the various types of geographical data. The algebras proposed here are able to perform various classes of spatial analysis, including relatively complex ones.

Since the GIS industry has matured to a point where questions of data structure, algorithms and functionality are becoming standardised, data modelling is seen as playing a critical rôle in determining the usability and adequacy of a system [Good92]. This concern has led to a number of conceptual formulations for geographical data models, and to a growing interest in the formal definition of geographical operations.

This paper is part of the conceptual work behind the implementation of SPRING, a geographical information system which integrates the different classes and representations of geographical data. For a description of SPRING, see

¹ Revised version.

[CSFG96]. The algebra described herein is being used as the basis for defining LEGAL, a general purpose query and manipulation language used in the second version of SPRING.

2 Previous Work

Although the duality between *fields* and *objects* as representations of geographical reality on a GIS is well-established in the literature, there are very few attempts at providing a unified perspective of geographical operations.

Research on geographical algebra operators has been traditionally divided into two main branches: *manipulation function* on maps and *query and presentation* operations on objects. Tomlin [Tom90] presents a set of operations on map (most oriented towards the raster representation) called “map algebra”. Egenhofer [Ege90; Ege94] discusses the problems of designing a query and presentation language for geographical data (dealing mostly with the vector representation of geographical data).

In this work, we discuss the definition of algebras of *geographical objects*, *geographical fields*, and the transformations between *geographical fields* and *geographical objects*.

3 An Object-Oriented Model for Geographical Data

3.1 Model Definition Framework

In defining our object-oriented model, we shall follow the class-based framework of [Bee89]: an object is an instance of a class and is characterised by its *state*, or set of attribute values, and *behaviour*, or set of operations or methods that can be applied to the object. An object o can be constructed out of other objects o_1, \dots, o_n , in which case o is called *complex* and o_1, \dots, o_n are called the *components* of o . If an object is not complex, then it is called *simple*. Classes can be structured into hierarchies; the ancestors of a class C in the hierarchy are called the *superclasses* of C .

Our model enables modelling the real world as a collection of object-oriented classes, divided into *conventional classes* and *geographical classes* (or *geo-classes*). The geo-classes model geographical fields and objects, whereas the conventional classes correspond to classes whose instances are non-spatial objects.

The same real-world entity might be modelled as a part of geo-class or as part of a standard class, depending on the situation. For example, a line transformer in an electrical network, when stocked in a warehouse, may be modelled as an instance of a non-spatial class, with descriptive attributes such as weight and capacity. The same transformer, when installed in a network location, shall be considered as an instance of a geo-class. As traditional data modelling is extensively dealt with in data base literature, we shall from now on concentrate on the geographical classes.

We shall also consider that each geographical class of objects has both locational and conventional attributes. The locational attributes and associated properties are described below, and the *conventional* attributes are assumed to be derived from an universe U of *descriptive* attributes $\{A_1, \dots, A_n\}$, defined on domains $D(A_1), \dots, D(A_n)$.

3.2 Basic Model Hierarchy

Geographical Region

Definition 1 (Geographical Region).

A set of points R which is a subset of \mathfrak{R}^2 is called a geographical region.

Although this definition is independent of scale and projection considerations, it will be sufficient for our model.

Geo-Fields

A *geographical field* or *geo-field* represents a continuous geographical variable over some region of the Earth.

Definition 2 (Geographical Field).

Let R be a geographical region. A **geo-field** f is an object $[a_1, \dots, a_n, |]$, where $a_i \in D(A_i)$ and $| : R \rightarrow V$ defines a mapping between points in R and values on a domain V .

The geographical fields can be specialised. Depending on the range of the variable, we define the following subclasses of GEO-FIELD:

- THEMATICAL - an instance of this class, called a *thematical geo-field*, defines a mapping $\lambda: R \rightarrow V$ such that V is a finite denumerable set. The elements of V are called *geo-classes* and, intuitively, define the *themes* of a thematic map.
- NUMERICAL - an instance of this class, called a *digital terrain model* or simply a DTM, defines a mapping $\lambda: R \rightarrow V$ such that V is the set of real values.
- REMOTESENSINGDATA - a specialization of the NUMERICAL class, whose instances have a range V which is a set of discrete values obtained by quantization of the response of the earth's surface to incident radiation, obtained by an active or passive sensor. This class is particularly useful to integrate remote sensing images into a GIS.

Figure 1 shows an field where the R is the region of Manaus and the mapping λ associates to each element of R "its reflectance to the solar radiation on the LANDSAT TM sensor, spectral band 4".

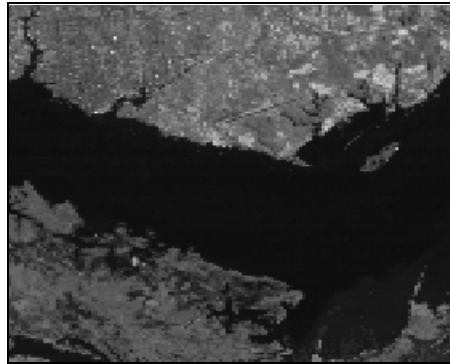


Figure 1 - Landsat image over Manaus

Geo-fields can be represented in a GIS in various formats. These representations reflect GIS *system design* decisions. We will not discuss the issue in detail here, but note that *digital terrain models* can be represented by regular grids or triangular grids, *thematic maps* can be represented by a topologically-structured set of vectors or by a symbolic array (raster representation), and *images* are usually represented by an array of values (raster representation).

The advantages and disadvantages of each storage option have been discussed extensively in the literature. Most studies have come to the conclusion that raster *and* vector (as well as regular *and* triangular grid) representations are useful alternatives, and a general GIS should provide both.

Geo-objects

Geo-objects represent individualizable entities of the geographic realm. They are phenomena that may have one or more *graphical representations*, which correspond to the geo-referenced set of co-ordinates that describe the object's location.

Definition 3 (Geo-object).

Given a set of geographical regions R_1, \dots, R_n , a **geo-object** go is an object $[a_1, \dots, a_n, geo_1, \dots, geo_n]$, composed by the values $a_i \in D(A_i)$ and by a set of geographical locations geo_i (where $geo_i \subseteq R_i$). We shall indicate the i -th attribute of go by $go.A_i$ and the i -th geographical location of go by $go.R_i$.

In other words, an object is a unique element that can be represented in one or more points in space, and which has various descriptive attributes. This definition allows for multiple geometrical representations to be assigned to the same geo-object.

Geo-objects are usually associated to a *vector* representation (points-lines-area). Figure 2 shows a geometrical representation of geo-object "Italy", shown in connection of the representations of other countries in Europe.

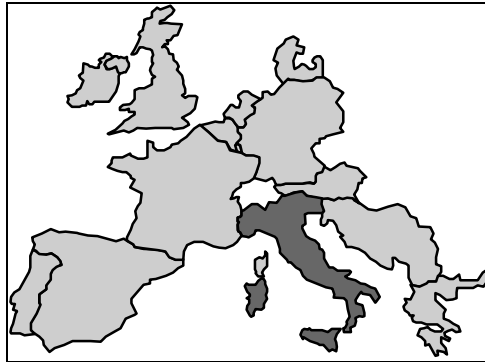


Figure 2 - Example of a geo-object

Geo-object Maps

In a GIS, each geographical object is associated to one or more geographical locations. Since most applications do not deal with isolated elements in space, it is convenient to store the graphical representation of geo-objects together with its neighbours. For example, the parcels of the same city borough are stored and analysed together.

These features lead us to introduce the concept of *geo-object maps*, which group together geo-objects for a given cartographic projection and geographical region.

Definition 4 (Geo-object Map).

Let R be a geographical region. A **geo-object map** mo is an object $[R, GO, geo]$ such that GO is a set of geo-objects and geo is a mapping $GO \rightarrow R$, which assigns, for each geo-object $go \in GO$, a location $geo(go)$ in R .

Therefore, the representations for *geo-objects* are maintained in instances of the class OBJECT-MAP. In practice, the mapping $geo(go)$ can be interpreted as the *link* between an description of a geographical object and its spatial location on a geographical region. This definition allows for multi-scale, multi-tile and multi-temporal representations to be associated to the same geo-object. This situations is typical of large geographical data bases, which include maps in different scales and projections, and over several UTM zones.

To illustrate the concept, consider figure 3, which illustrates a data base for rivers of the Brazilian Amazonia. Since the region covers a very large area, a geographical data base in the 1:250000 scale (on UTM projections) will consist of several non-overlapping tiles. We associate each UTM partition to an instance of the class OBJECT-MAP which includes a

mapping for all rivers which are included in the geographical area of the partition. Therefore, the Amazon river is seen by the data base as a single object, even though its representation may span several maps.

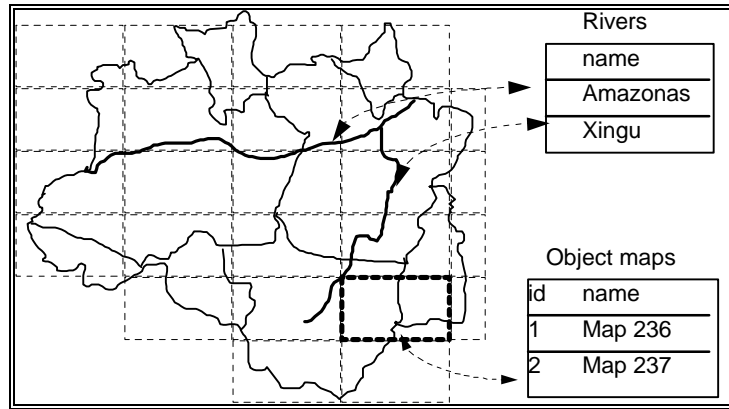


Figure 3 - Geo-Objects and Object Maps.

3.3 Operations on Geographical Data

There are three main types of geographical algebras, discussed in section 4:

- *Geo-objects algebra*: selection and query of geo-objects, based on descriptive and spatial properties.
- *Fields algebra*: manipulation of fields.
- *Combined operations*: generation of *geo-object maps* from fields, and generation of *fields* from *geo-objects*.

4 Algebra of Geo-Fields

We discuss the following types of operators for the algebra of geo-fields: *point*, *neighbourhood* and *zonal*.

Point Operators

A point operator produces a new geo-field, whose value in each point p depends only of the values in p in the input geo-fields. A point operations is specified as a mapping between the *ranges* of the input and output fields.

Definition 5. Point operations on fields.

Let R be a geographical region, V_1, V_2, \dots, V_{n+1} sets which define possible ranges for geo-fields, and F_i ($i=1, \dots, n+1$) be the class of all geo-fields which have R as a location and V_i as its range.

The *point operation* $\Pi: F_1 \times F_2 \rightarrow F_{n+1}$ induces a function π such that, for every geo-field $f_i \in F_i$ ($i=1, \dots, n$):

$$f_{n+1}(p) = \pi(f_1(p), \dots, f_n(p)), \quad p \in R.$$

where the spatial values of the output geo-field $f_{n+1} \in F_{n+1}$ are defined by the mapping $\lambda_{n+1}: M \rightarrow V_{n+1}$.

Point operators include transformation operators, mathematical functions, boolean operations, comparison operators and functions such as finding extremes and averages. The value of the output field at each location is a function only of the input values at the corresponding location. Depending on the ranges of the input and output fields, there are different possibilities for P :

- **unary operators**, such as *weighting* (a mapping from a thematical geo-field into a numerical geo-field), *slicing* (transformation from numerical into thematical) and *reclassification* (mapping a thematical geo-field into another).
- **boolean and comparison operators** can be applied to all types of geographical fields. When the resulting map is a thematic map, it is usually necessary to specify a set of conditions that have to be satisfied for each output class. An example: “calculate a soil aptitude map based on climate, soil, and slope maps, where the conditions are such that a soil is deemed “good for agriculture” if it rains more that 1000 m/year and the soil has a *ph* between 6.5 and 7.5, and the slope is less than 15%”.
- **mathematical operators**, such as arithmetic and trigonometric functions, can be applied to DTMs and (with restrictions on the output range) to IMAGES. An example would be: “calculate a soil loss equation, given by: $(\text{slope map})^{0.25} * (\text{soil ph})^2$ ”.

Figure 4 shows an example of the “weighting” operation (the conversion of a soils map into a weighted soils map). In this case, $V_1 = \{ \text{Le, Li, Ls, Aq} \}$, $V_2=[0.0,1.0]$ and ρ is the set of ordered pairs $\{(\text{Le} \rightarrow 0.60), (\text{Li} \rightarrow 0.20), (\text{Ls} \rightarrow 0.35), (\text{Aq} \rightarrow 0.10)\}$.

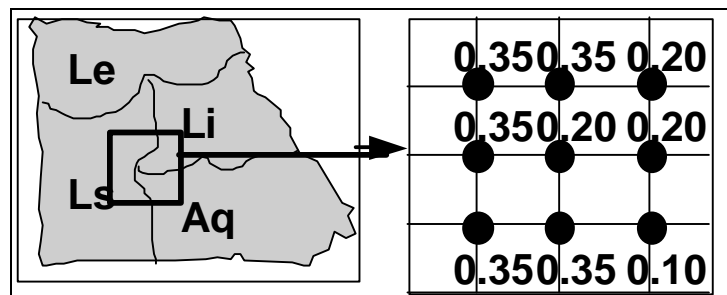


Figure 4 - Example of the “weighting” operation.

Neighbourhood operators

In this class of operators, the output field is computed based on the values of a continuously-varying surface in the neighbourhood of each location of the input field. To that end, we need to provide a definition for the neighbourhood in a geographical region.

Definition 6. Neighbourhood in a geographical region

Given a geographical region R , a set of $P \subseteq R$ is said to be *connected* iff, for any two points $p_1, p_2 \in P$ there is a line connecting these two points which is entirely contained in R . A *neighbourhood in R* is a mapping

$$N: R \rightarrow 2^R, \text{ such that } \forall p \in R, p \in N(p) \text{ and } N(p) \text{ é connected.}$$

Definition 7. Neighbourhood operations on geo-fields.

Let R be a geographical region and $F_0 \in F_1$ the sets of geo-fields which are defined over R and whose range is $V_i, i = 0, 1$. Let $N: R \rightarrow 2^R$ e $v: 2^{V_1} \rightarrow V_0$. The neighbourhood operation $\Psi: F_1 \rightarrow F_0$ induced by v is such that:

$$\forall f_1 \in F_1, \Psi(f_1) = f_0 \Leftrightarrow f_0(p) = v(\{\lambda_1(x) \mid x \in N(p)\}), \forall p \in R.$$

An example of this operation would be: “calculate the slope of an elevation map, based on the local derivatives at each

Zonal operations

This is a special class of neighbourhood operators, where one geo-field (usually a thematic map) is used as a spatial restriction on the operators on another geo-field (usually a DTM).

Definition 8. Zonal operations on geo-fields.

The zonal operation Z on a numerical geo-field f_1 , defined by $\lambda_1: R \rightarrow V_1$, (where V_1 is the set of reals), and a thematic geo-field f_2 , defined by $\lambda_2: R \rightarrow V_2$, (where V_2 is a discrete set $\{v_1, \dots, v_n\}$), and a local function v is such that:

$$Z(f_1) = f_{new} \mid \lambda_{new}(p) = v(\lambda_1(x), x \in L(p)) \text{ and the zonal region } L(p) \text{ satisfies}$$

$$\forall p \in R, \exists L(p) \subset R \wedge p \in L(p), \text{ such that } f_2(x) = v_1 \mid \forall x \in L(p).$$

An example of zonal operations would be: “Given an slope map and a soils map, find the average slope for each soil area on

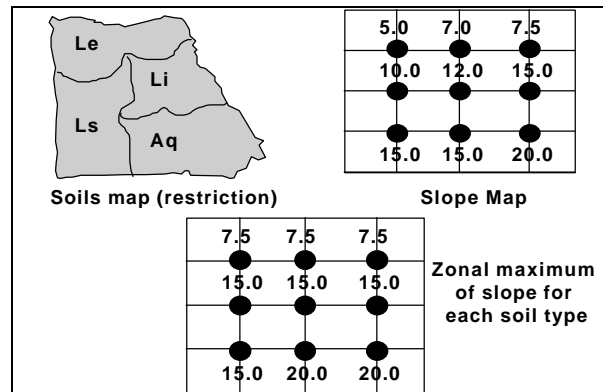


Figure 6 - Example of a zonal operation.

5 Geo-Objects Algebra

5.1 Spatial Relationships

In our model, we shall represent geo-objects as 2D geometries (points, lines and regions). As the operations of the geo-objects algebra may involve spatial restrictions, it is important to define spatial relationships, which may be divided in:

- *topological relationships*, such as “inside” and “adjacent to”, which are invariant to rotation, translation and scaling transformations. A formalization of this type of relationships has been proposed by Clementini et al. [CFO93], based on earlier work by Egenhofer [Ege90];
- *directional relationships*, such as “above” and “beside”. There are many informal proposals but little formal work for this class of operations;
- *metrical relationships*, derived from the distance operations.

In our work, we shall consider only topological and metrical relationships on \mathbb{R}^2 , based on the following definitions:

- An *area* A is a 2D set of points of dimension 2, whose interior A° is connected (with no holes) and which has a connected frontier δA .
- A *line* L is a set of connected points of dimension 1, whose frontier δL is the first and the last point or an empty set in the case of a circular line (an “island”), and its interior L° is the set of the other points.
- A *point* P is a set of dimension 0, whose interior P° is the point itself and whose frontier δP is empty.

To analyse the topological relationships on \mathbb{R}^2 , Egenhofer [Ege90] has proposed the use of the *4-intersection matrix*, which represents the relations between the interior and the frontiers of two point sets A and B:

$$\begin{bmatrix} \delta A \cap \delta B & \delta A \cap B^\circ \\ A^\circ \cap \delta B & A^\circ \cap B^\circ \end{bmatrix}$$

The 4-intersection matrix is not sufficient to uniquely identify all possible situations in the case of relationships between lines and areas and lines and lines. Therefore, Clementini et al. [CFO93] have proposed to consider the dimension of the intersection between the two sets and have found a minimal set of five relationships (*touch*, *in*, *cross*, *overlap* and *disjoint*) which are applicable to all cases. The formal definitions of these relationships is given below.

The *touch* relationship is applicable to area-area, line-area, line-line, point-area and point-line situations. A set of points S_1 *touches* another set S_2 when they have points in common, but their interiors do not:

$$S_1 \text{ touch } S_2 \Leftrightarrow (S_1 \cap S_2 \neq \emptyset) \wedge (S_1^\circ \cap S_2^\circ = \emptyset)$$

The *in* relationship is applicable to area-area, line-area, point-area and point-line situations. A set of points is *in* another when their intersection is the first set:

$$S_1 \text{ in } S_2 \Leftrightarrow S_1 \cap S_2 = S_1.$$

The *cross* relationship is applicable in the case of line-line and line-area situations. A line L *crosses* an area A when their interiors meet and the intersection of the two sets is not the line itself; two lines *cross* when their interiors have a non-empty intersection and this intersection is a set of points of dimension 0:

$$L \text{ cross } A \Leftrightarrow (L^\circ \cap A^\circ \neq \emptyset) \wedge ((L \cap A) \neq L).$$

$$L_1 \text{ cross } L_2 \Leftrightarrow (L_1^\circ \cap L_2^\circ \neq \emptyset) \wedge (\dim(L_1 \cap L_2) = 0).$$

The *overlap* relationship is applicable to area-area, line-line and point-point situations. Two point sets S_1 and S_2 *overlap* when their intersection is different from them, but forms a set of points of the same dimension:

$$S_1 \text{ overlap } S_2 \Leftrightarrow (S_1 \cap S_2 \neq \emptyset) \wedge (\dim(S_1 \cap S_2) = \dim(S_1) - 1)$$

These situations are illustrated in Figure 7. For a proof of these definitions, please refer to [Cam95].

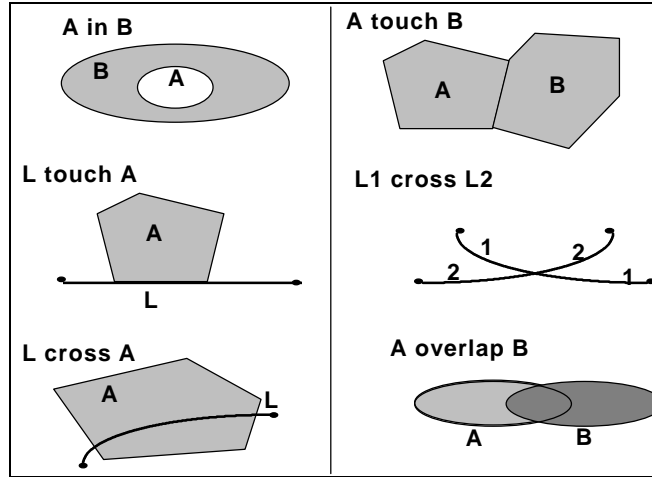


Figure 7 - Examples of topological relationships.

5.2 Operations

In order to define the spatial operations over geo-objects, we need to establish the notion of a *computable spatial predicate*.

Definition 9. Computable Spatial Predicate.

Let R be a geographical region, and GO a set of geo-objects which have representations in R , defined by an object map $om = [R, GO, geo]$.

A *computable spatial predicate* χ is a spatial restriction, defined by a topological relationship (*inside, touch, cross, overlap e disjoint*) or a metrical relationship, which can be computed over the representations $geo(go_i)$ of the geo-objects $go_i \in GO$.

Spatial selection

Definition 10. Spatial selection

Let R be a geographical region, GO a set of geo-objects and mo an object-map $mo = [R, GO, geo_1]$ which contains the spatial location of the geo-objects $go \in GO$ in R .

The *spatial selection* operation $j : GO \rightarrow GO$, given a spatial predicate ξ which relates the geo-objects $go \in GO$ to a geo-object go^* which is represented in mo by a mapping $geo_2(go^*)$:

$$\phi_{\chi}(GO) = \{ go \in GO \mid \xi(geo(go)) \}.$$

The output of such operation is a subset of the original set, composed of all geo-objects that satisfy the geometrical predicate, as the example illustrate:

- “select all regions of France which are adjacent to the Midi-Pyrenees regions (which contains the city of Toulouse)”.

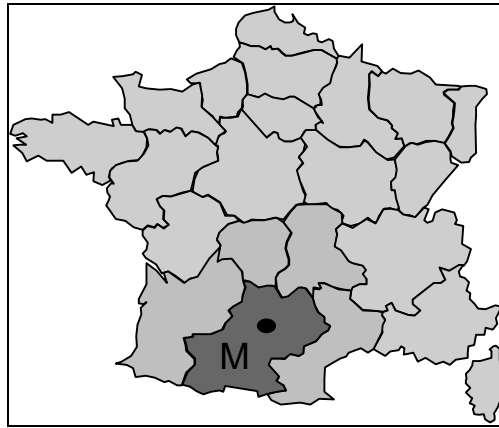


Figure 8 - Example of a spatial selection operation.

Spatial Join

Definition 11. Spatial Join

Let R be a geographical region, GO_1 and GO_2 two sets of geo-objects and mo_1 and mo_2 object-maps $mo_i = [R, GO_i, geo_i]$ which contain, respectively, the spatial location of the geo-objects $go_1 \in GO_1$ and $go_2 \in GO_2$ in R . Let ξ be a spatial predicate computable for every pair of geographical locations $((geo_1(go_1), geo_2(go_2)))$.

The spatial join operation $\theta: GO_1 \times GO_2 \rightarrow GO_1 \times GO_2$ is such that:

$$\theta_{\xi}(GO_1, GO_2) = \{ (go_1, go_2) \in (GO_1, GO_2) \mid \xi(geo_1(go_1), geo_2(go_2)) \}.$$

The spatial join is an operation where a comparison between two sets of geo-objects GO_1 and GO_2 takes place, based on a spatial predicate which is computed over the representation of these sets. The name “spatial join” is employed by analogy to the join operation in relational algebra. The result of the spatial join operation is a *set of object-pairs*, which satisfy the spatial restriction. Examples are:

- “Find all indian reservations located closer than 50 km to the main roads in Amazonia”.
- “Find all cities in the state of Ceara which are located close than 10 km from a water reservoir.”

In the first example, the answer is a set of pairs of geo-objects (reservation, road) and in the second a set of pairs (cities, reservoir).

6 Transformations between Geo-Fields and Geo-objects

Another set of operations for geographical data concerns the transformations that generate geo-fields from sets of geo-objects (and vice-versa). These transformation operations are of special importance, as they are the link between the two general classes of geographical data.

6.1 Generation of Geo-Objects from Geo-Fields

We shall consider one important instance of such operations, that of *spatial interpolation*.

Definition 12. Spatial Interpolation

Let R be a geographical region, V_1, V_2, \dots, V_n sets which define possible ranges for geo-fields, and $F_i (i=1, \dots, n)$ be the class of all geo-fields which have R as a location and V_i as its range. Let GO be a set of geo-objects and mo be an object-map $mo=[R, GO, geo]$ which assigns geographical locations in R to the geo-objects in GO .

The *spatial interpolation* operation $\otimes: F_1 \times \dots \times F_n \rightarrow GO$ is such that:

$$\forall f_1 \in F_1, \dots, f_n \in F_n,$$

$$\otimes(f_1, f_2, \dots, f_n) = GO \Leftrightarrow \forall go \in GO, go = [v_1, \dots, v_m, a_{m+1}, \dots, a_n, geo(go)], \text{ and}$$

$$geo(go) = \{ p \in R \mid f_1(p) = v_1 \wedge \dots \wedge f_n(p) = v_n \}.$$

This definition corresponds to the generation of an object map from the spatial intersection of a set of geo-fields. This situation occurs, for example, in *zoning* applications, when an overlay of thematic maps is performed to obtain homogeneous zones. When a cadastral map is created from an overlay of geo-fields, each resulting geo-object inherits all descriptive attributes from the original geo-fields. Consider the following example, as shown in figure 9: “Determine the homogeneous regions of Australia, as the intersection of the vegetation, geomorphology and soils maps”.

In the GIS literature, the spatial intersection operation is very often wrongly classified as “a special type of spatial join” [11]. Although there are similarities in graphical algorithms used to compute them, the *spatial intersection* operation is conceptually different from the *boolean operations* between geo-fields and from *spatial join* operations between geo-objects.

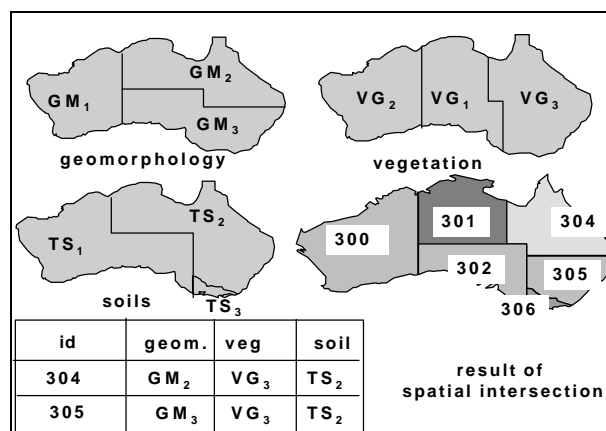


Figure 9 - Spatial Interpolation Operation

6.2 Generation of Geo-Fields from Geo-Objects

These operations take as input a set of geo-objects GO , represented in the geo-objects map mo and generate as output a field f , defined on a map M by a mapping $\lambda: M \rightarrow V$. We shall consider two operations, that of *distance maps* (buffer zones) and that of *attribute reclassification*.

Buffer zones

Definition 13. Buffer zones operation.

Let R be a geographical region, F a set of geo-fields defined over R whose range is \mathbb{R}^+ . Let GO be a set of geo-objects, and mo an object-map $mo = [R, GO, geo]$, which assigns geographical locations in R to the geo-objects in GO .

The buffer zones operation $\Delta: GO \rightarrow F$ induced by mo is such that, given a distance metric $dist$ computable in mo and an object $go \in GO$:

$$\Delta_{mo}(go) = f \Leftrightarrow \forall p \in R, f(p) = dist(p, geo(go)).$$

Figure 10 shows the example of a buffer zone operation.

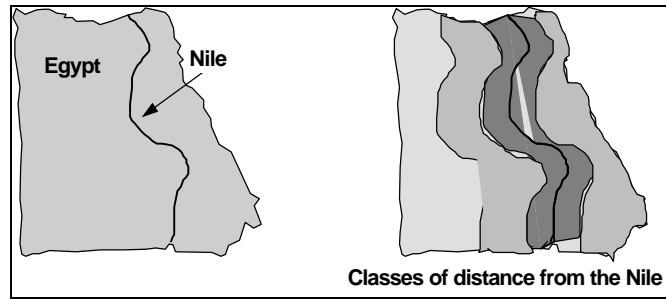


Figure 10 - Example of buffer zones operation.

Attribute reclassification

Definition 14. Attribute reclassification operation.

Let R be a geographical region, GO be a set of geo-objects whose descriptive attributes are contained in $D(A_1) \times \dots \times D(A_n)$, and mo an object-map $mo = [R, GO, geo]$, which assigns geographical locations in R to the geo-objects in GO .

Let F a set of geo-fields defined over R whose range is $D(A_i)$, where A_i is the i -th descriptive attribute of GO .

The attribute reclassification operation $\Omega: GO \rightarrow F$ induced by mo is such that:

$$\Omega_{mo}(GO) = f_0 \Leftrightarrow (\forall go \in GO, f_0(geo(go)) = go.A_i).$$

From the values of a specific descriptive attribute of a set of geo-objects, a new geo-field is created, whose mapping is defined by the spatial distribution of the chosen attribute. This is illustrated in Figure 11, which shows the operation:

- “For all countries in South America, generate a thematic map with the population growth of each country, divided in classes: { (from 0 to 2% per year), (from 2 to 3%), (more than 3%)}.”

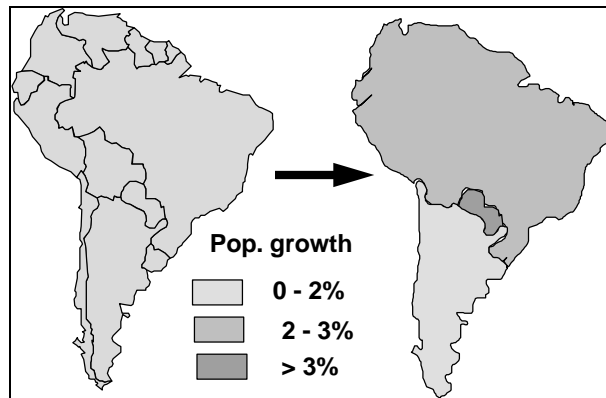


Figure 11 - Attribute reclassification operation.

7 The LEGAL language

The analysis of the algebras of GIS operations serves as a basis for the definition of a language for query and manipulation of spatial data, called LEGAL (in Portuguese, “Linguagem Espaço-Geográfica baseada em Álgebra” - Spatial Algebra Language).

The main features of LEGAL are:

- The operations of geo-objects algebra are implemented using extensions of the relational language SQL.
- The fields algebra and the combined field-object operations are implemented by statements which have the same semantic level as the SQL language.

LEGAL is *strongly typed*, and has the following basic types:

- THEMATIC, IMAGE, DTM, which are specialisations of geo-fields;
- OBJECT, for geo-objects;
- OBJECT MAPS, for geo-object maps;
- COLLECTIONS, for storing collections of geo-objects resulting from spatial join operations.

Further work by the authors ([Cam95] [CCF+96]) concentrates on the definition, implementation and use of LEGAL.

Acknowledgements

SPRING is the result of a team effort, which includes the following partners:

- INPE-DPI (National Institute for Space Research/Image Processing Division)
- EMBRAPA/CNPTIA (Brazil's Agricultural Research Agency - Center for Technological Research in Informatics for Agriculture).
- IBM Brasil - Latin American Center for Solutions for Higher Education and Research
- CC/SIVAM (Coordinating Commission for the System for Surveillance of the Amazon).

The project has received generous support from the Brazilian National Research Council (CNPq), through the programs RHAE and ProTem/CC ("GEOTEC project").

SPRING's development team includes:

At INPE: Carlos Felgueiras, Cláudio Clemente Barbosa, Eduardo Camargo, Fernando Mitsuo Ii, Fernando Yutaka Yamaguchi, Guaraci José Erthal, Eugenio Sper de Almeida, João Argemiro de Carvalho Paiva, João Pedro Cordeiro, João Ricardo Freitas Oliveira (1991-1993), José Cláudio Mura, José Zamith (1993-1994), Júlio Cesar Lima D'Alge, Laércio Namikawa, Lauro Hara, Leila Garcia, Leonardo Bins Marina Ribeiro, Marisa da Motta, Mary Minamoto, Misae Yamamoto, Silvia Shizue Leonardi, Sergio Rossim and Virginia Correa. Silvana Amaral, Eymar Lopes, Flavia Nascimento, Regina Bruno, Maycira Costa (1992-1994) and Lygia Mammanna (1992-1994) have assured user support, quality control and user documentation. Diógenes Alves, José Carlos Moreira, Eliana Kalil have conducted extensive applications of SPRING for land use and cover change surveys in Amazonia.

At IBM Brazil: Marco Casanova, Andrea Hemerly, Paulo Souza, Alexandre Plastino, Mauricio Mediano.

At EMBRAPA: Carlos Costa, João Camargo, Ivan Lucena, Moacir Pedroso.

At CC/SIVAM: Ana Paula Dutra de Aguiar, Claudia Tocantins.

References

- [Bee89] Formal Models for Object-Oriented Databases. In: PROC. 1ST. INTERNATIONAL CONFERENCE ON DEDUCTIVE AND OBJECT-ORIENTED DATABASES, pages 370-395, 1989.
- [Cam95] Câmara, G. "Models, Languages and Architectures for Geographical Information Systems". Ph.D. thesis, INPE, 1995.
- [CCF+96] Câmara, G.; Casanova, M.A.; Freitas, U.M.; Cordeiro, J.P.C; Hara, L. "A Presentation Language For Gis Cadastral Data". In: SECOND ACM WORKSHOP ON ADVANCES IN GEOGRAPHIC INFORMATION SYSTEMS, Proceedings, ACM, Washington, DC, 1996.
- [CFC94] Câmara, G.; Freitas, U.M.; Cordeiro, J.P.C. "Towards an Algebra of Geographical Fields". in: VII SIMPÓSIO BRASILEIRO DE COMPUTAÇÃO GRÁFICA E PROCESSAMENTO DE IMAGENS. Anais. Curitiba, 1994, pp. 205-212.
- [CSFG96] Câmara, G.; Souza, R.C.M.; Freitas, U. M.; Garrido, J. Spring: Integrating Remote Sensing And Gis By Object-Oriented Data Modelling. Accepted for publication in *Computers and Graphics*, vol.20, n.3, 1996.
- [CFO93] Clementini, E.; Di Felice, P.; Van Oosterom, P., 1993. "A Small Set of Formal Topological Relationships : THIRD INTERNATIONAL SYMPOSIUM ON SPATIAL DATA HANDLING, Proceedings, Singapore, 1993, pp. 277-295.

- [Ege89] Egenhofer, M., 1989. *Spatial Query Languages*, Ph.D. Thesis, Depart. of Surveying Engineering, University of Maine, 1989.
- [Ege94] Egenhofer, M., "Spatial SQL: A Query and Presentation Language", *IEEE Transactions on Knowledge and Data Engineering*, vol.6, pp. 86-95, 1994.
- [Goo92] Goodchild, M., "Geographical data modeling". *Computers & Geosciences*, vol. 18, no.4, pp. 401-408, 1992.
- [Güt94] Gütting, R.H., 1994. "An Introduction to Spatial Database Systems". *VLDB Journal*, vol. 3, no.4, October 1994.
- [Tom90] Tomlin, D. *Geographic information systems and Cartographic Modeling*. Prentice Hall, New York, 1990.