

A GENERAL DATA MODEL FOR INTEGRATING REMOTE SENSING AND GIS DATA

103

GILBERTO CÂMARA, UBIRAJARA MOURA DE FREITAS

RICARDO CARTAXO MODESTO DE SOUZA, JOÃO ARGEMIRO DE CARVALHO PAIVA

Image Processing Division (DPI)

National Institute for Space Research (INPE), Brazil

MARCO ANTONIO CASANOVA, ANDREA SILVA HEMERLY

Rio Scientific Centre IBM Brazil
er

ABSTRACT

This work discusses the design of an object-oriented data model for GIS, which caters for the diversity of data sources and formats, including both raster and vector data. The model combines the ideas of "layers" and "objects", and provides mechanisms for generalisation and specialisation of geographical data. The model is being used as a basis for the development of SPRING, a system which includes functions for image processing, geographical analysis and digital terrain modelling, integrated with an environmental data base.

1. INTRODUCTION

This work introduces a *general data model* for a GIS. The model, designed with object-oriented techniques, copes with the various types of environmental data. The more relevant contributions of the proposal are:

- integration of remote sensing images and digital terrain models with vector-based information (thematic maps, cadastral maps and networks).
- definition of a mapping between objects and their locations (similar to multimedia anchor), enabling more than one graphical representation to be associated with the same real-world entity.
- using the model for defining a high-level interface with semantic content.

This paper is an evolution of earlier work done at INPE (Erthal et al., 1988), which has been coupled with the design and implementation of systems. This proposal has been used as the basis for the

design and implementation of a GIS system (SPRING), which works on UNIX workstations, under the X window system.

2. THE IMPORTANCE OF DATA MODELS ON GIS DESIGN

A *data model* is a comprehensive set of conceptual tools used for structuring data in a GIS. Arguably, the *data model* is the most important single issue in the design of a GIS, since it describes how geographical reality is represented in the computer. No other decision limits so much the system's applicability and extension.

Data models used in many commercial GIS systems reflect directly the underlying geometries. The user refers to arc-node structures (in the case of vector-oriented systems) or to "grids" or "quad-trees" (for raster-based ones).

As stated by Goodchild (1992), 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. This concern has led to a number of conceptual formulations for geographical data models, and to a growing interest on object-oriented methods.

3. A GENERAL FRAMEWORK FOR GEOGRAPHICAL DATA

In order to introduce the model, it is necessary to distinguish between the various universes (levels) involved in the modelling process (Gomes and Velho, 1998):

- the *real-world* universe, which comprises the geographical reality entities that will be modelled in the computer. At this level, we find elements such as *parcels, terrain, rivers and telephone networks*.
- The *conceptual (mathematical)* universe, which includes a formal (mathematical) definition of the

further specialised, such as *grids, TINs, image* structures for raster and *arc-node* and *arc-node-polygon* structures.

- The *implementation* universe, where the data structures for operations on the geographical data are chosen, based on considerations such as performance and machine size. At this level, the actual coding takes place and we find data structures such as *R-trees* and *quad-trees*.

In this view, the *field-object* and the *raster-vector* dichotomies can be combined, since they are located at different abstraction levels. It also indicates that the user interface for a GIS should reflect, as much as possible, the conceptual (formal/mathematical) level, and hide, as much as possible, details from the representation and implementation levels. At the conceptual level, the user would deal with abstract concepts which are closer to his day-to-day reality, rather than having

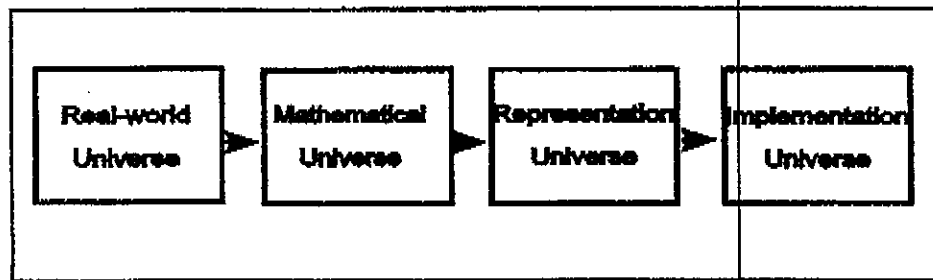


Figure 1 - Abstraction levels in Modelling

entities which are included in the model. At this level, we distinguish between *fields* and *objects*, and further specialise these notions into classes of geographical data.

- The *representation* universe, which defines how the various classes of geographical data are mapped to the different graphical representations. At this level, we distinguish between *raster* and *vector (topological)* representations, which may be

to understand intricacies of graphical representations.

3.1 The real-world universe

The model is not restricted to any particular GIS application, but aims at being a general framework for designing Geoprocessing systems, with application on fields such as Environmental studies, Agriculture, Geology and Facilities Management.

3.2 The conceptual (mathematical) universe

From a conceptual point of view, there are two large classes of objects in spatial databases: *fields* and *objects* (Goodchild, 1992b). On what follows, we define these concepts formally, and propose new abstractions (*geo-object maps*, *information layers* and *projects*). The model classes for the conceptual level are illustrated in figure 2.

Geo-objects

A *geo-object* is an abstraction of reality, and corresponds to an individual entity of the geographic realm. We denote the class of geo-objects by GEOOBJECT. The application designer may define specializations of geo-objects, that add attributes and methods.

Each geo-object has a unique identifier and attributes. A geo-object may have one or more *graphical representations*, which correspond to the geo-referenced set of coordinates that describe the object's location.

GIS applications usually do not store

and manipulate isolated graphical representations of geo-objects. A cadastral map of an urban area is such an example. Retrieving a line that describes a street and visualising it on the screen without its context hardly makes sense. By contrast, retrieving a street map and highlighting the line is the normal practice.

We therefore introduce the concept of *geo-objects maps*, which group together geo-objects for a given cartographic projection and geographical area, and are instances of the class GEOOBJECTSMAP. More formally, given the description of a region A (which includes a cartographic projection, a scale and a bounding rectangle), a *geo-objects map* has the following attributes:

- the DOMAIN of the map, whose value is a set of \mathcal{O} of geo-objects;
- the RANGE of the map, whose value is a set \mathcal{G} of graphical representations, wholly contained in A;
- a total function $f: \mathcal{O} \Rightarrow \mathcal{G}$ that assigns to each geo-object o in \mathcal{O} a unique graphical description $f(o)$ in

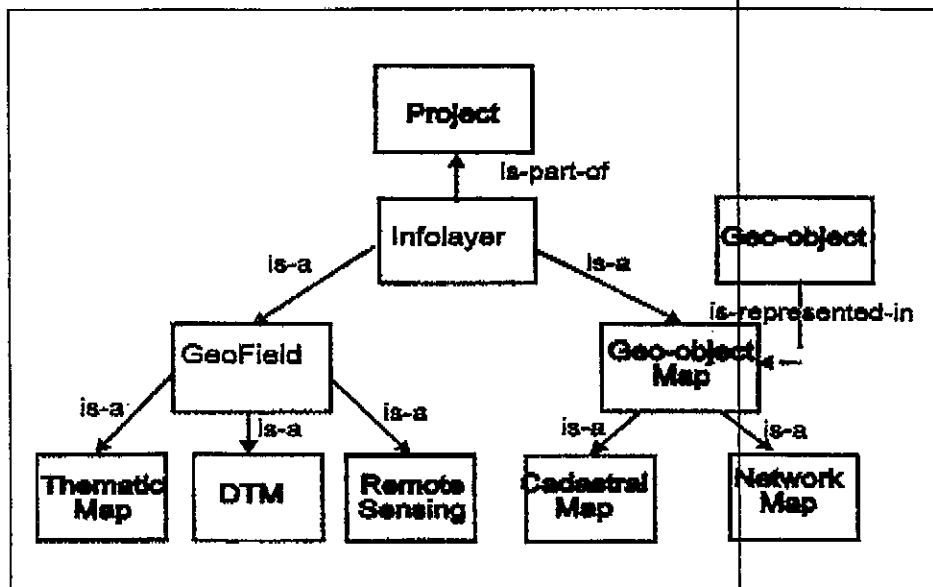


Figure 2 - The conceptual level

NO! -
Not at this
level.

K

⊆ (note that two objects may have the same description).

A geo-object can be a component of more than one map and thus have several representations, for the following reasons:

- The different representations may differ in terms of the geo-referencing scheme used, the precision adopted or the desired level of detail;
- they represent different historical versions or versions of the same data obtained from different sources;
- they partially represent the geo-object in different regions (we consider different map sheets as distinct);
- they represent the geo-object in maps associated with different themes.

The relation between a geo-object and a geo-object maps is an IS-REPRESENTED_IN one. More formally, given a set O of geo-objects and a set M of geo-object maps, we define the relationship IS-REPRESENTED_IN $\subseteq O \times M$ such that $(o, m) \in$ IS-REPRESENTED_IN iff o is in the domain of m (intuitively, iff m defines a locational representation for o).

graphical

A geographical database management system may implicitly or explicitly store the IS-REPRESENTED_IN relationship to enable the user to easily move from geo-objects to maps and vice-versa. This relationship may also be understood as a collection of *hypermedia links* connecting geo-objects to maps, with the graphical representation playing the rôle of *hypermedia anchors* within the maps.

graphical
of
maps?

Finally, we consider two specialisations of GEO-OBJECT MAP:

- CADASTRAL MAPS, whose instances, called *cadastral maps*, describe the location of *land information elements* (such as parcels and streets) and use the arc-node-polygon topology.

- NETWORK MAPS: these maps store the arc-node topology, location and other ancillary information that correspond to the linearly connected structures of the network. Ancillary information includes flow directions and dynamic segmentation properties.

Geo-fields

A *geographical field* or *geo-field* is a complex object that represents the spatial distribution a geographical variable over some region of the Earth's surface. We denote the class of geo-fields by GEO-FIELD. A geo-field has a unique identifier, and is characterised by:

- its DOMAIN, the description of a region A ;
- its RANGE, whose values define the set of values V that the geographical variable may take;
- a MAPPING $f: A \rightarrow V$ between points in A and values in V . If we include the so-called "~~dummy~~" or undefined value in V , then f will be a total function.

The variable can take values in the entire range of reals, or can be limited to descriptive choices. In the first case, geographers normally refer to *digital terrain models*; in the latter, to *thematic maps*. Satellite and aerial images are a special case of DTMs, since a continuous variation (that of reflectance to incident radiation) is usually quantized to a limited range.

Therefore, we define the following subclasses of GEO-FIELD:

- THEMATIC MAP - an instance of this class, called a *thematic map*, defines a mapping function $f: A \rightarrow V$ such that V is a finite set. The elements of V are called *geo-classes* and, intuitively, define the *themes* of the map.

- DIGITAL TERRAIN MODEL - an instance of this class, called a *digital terrain model* or simply a DTM, defines a mapping function $f: A \Rightarrow V$ such that V is the set of real values.
- REMOTE SENSING DATA - a particular case of a DTM, where data represents the reflectance of the earth's surface to incident radiation. At the mathematical level, the distinction between a DTM and a remote sensing data is purely one of convenience (and of tradition).

Information layers

Given that *geo-fields* and *geo-objects maps* have in common the property of geographical location, it is useful to define a *container* element, an abstract entity upon which GIS functions can be defined, which we call information layer (or infolayer for short).

We introduce the class INFOLAYER as a generalisation of the classes GEOOBJECTMAP and GEOFIELD. An instance of this class, called an *infolayer*, corresponds to a generalised notion of a *map*, and represents, for a given geographical area, the values of a *geo-field* or the geographical location of a set of *objects*.

Projects

We define a project as a complex object consisting of a collection of infolayers that have the same geo-referencing scheme (specific geographical area, scale and

projection). Projects frequently become the basic unit of work, in the sense that the user will initiate a session by selection a project and asking the GIS to retrieve infolayers that include data within the project.

A project can be seen as a *view* of the data base. Because of processing concerns, this vision may need to be materialised, that is, data is clipped to correspond to the geographical area of interest, and a version control procedure is initialised.

3.3 The representation universe

Graphical representations

The mapping from the mathematical universe to the representational universe reflects GIS system design decisions. It defines, for example, if digital terrain models are represented in the system by regular grids, triangular grids or both.

With that idea in mind, we have used a general concept, that of a GRAPHICAL REPRESENTATION class. Instances of this class, called *graphical representations*, define a mapping description for an instance of the INFOLAYER class. The GRAPHICAL REPRESENTATION class can be further specialised into the classes RASTER GRAPHICAL REPRESENTATION and VECTOR GRAPHICAL REPRESENTATION. The former class can be further specialised into the IMAGE, THEMATIC IMAGE and REGULAR GRID classes. The latter, into the ARC-NODE-POLYGON, TRIANGULAR GRID (TIN) and ARC-NODE classes. Figure 3 shows this hierarchy.

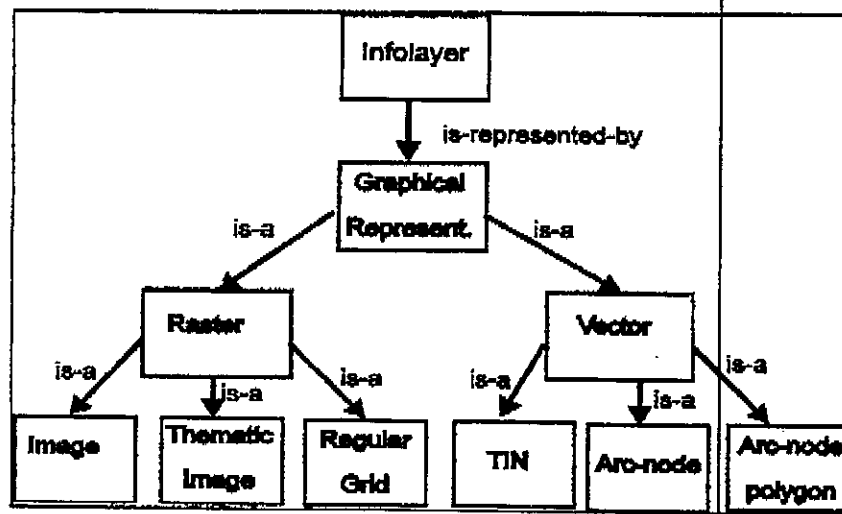


Figure 3 - The representation universe

More formally, given a set I of infolayers and a set R of graphical representations, we define the relationship $IS_REPRESENTED_BY \subseteq I \times R$ such that $(I, r) \in IS_REPRESENTED_BY$ iff I is in the domain of r (intuitively, iff r defines a graphical representation for I).

The set of possible pairs of (I, r) indicate the integrity constraints and design options for a GIS (the actual storage options for geographical data). For example, a thematic map can be stored either as a set of topologically-structured vectors or as a thematic image. 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 grid and TIN) representations are useful alternatives, and a general GIS should provide both.

Visual definitions- presentation control

Many GIS studies have pointed out the importance of allowing independent presentation control of the result of operations. The class `VISUALDEFINITION` has been devised solely to account for the graphical appearance of geographical data.

An instance of this class, called *visual definition* is associated to any class of the

model. The parameters controlled by a visual definition may vary according to each particular specialisation: in the case of DTMs, they included contour colours and style, and spacing between contour levels. For thematic maps, they would be shading colour, fill style and pattern and line width.

3.4 The implementation universe

The object-oriented approach has been proposed as a means of expressing the complex relationships between geographical data and of solving the problem of diversity of formats and structures (Egenhofer and Frank, 1989). We believe that object-oriented programming applies naturally to the implementation of a GIS environment. Each type of spatial data will be represented by a class, which may obey hierarchical relationships. Derived subclasses will inherit the behaviour of more general classes. The methods for each class implement manipulation, retrieval, transformation and visualisation procedures.

↓
los
↑

4. MODEL APPLICATION: USER INTERFACE

In our view, the proposed model enables the design of interface where the user operates at a higher level of abstraction. Interfaces designed using the object-oriented paradigm hide much of the so-called "GIS assembler", thus stepping up the learning curve, one of the greatest obstacles in fostering GIS technology.

In what follows, we discuss how the model would allow the definition of a geographical data base schema and simplifies function selection on GIS.

4.1 Definition of the Geographical Database Schema

The process of defining the conceptual schema of a geographical database consists of:

- specifying classes of geo-objects, possibly organised as a specialisation hierarchy;
- extending the specialisation hierarchy of infolayers introduced by the model;
- defining integrity constraints indicating, for each new class O defined as a specialisation of GEOOBJECT, which specialisations of GEOOBJECTMAP may contain descriptions of objects in O ;
- Defining integrity constraints for each specialisation of INFOLAYER (GEO-FIELD OR GEOOBJECTMAP classes), which specialisations of GRAPHICAL REPRESENTATION are associated with it.

Note that the last operation may not be possible if the system is not extensible (that is, if the mapping between the conceptual and the representation levels is fixed).

As an example, consider the definition of a geocoded database schema for a land use information system, which might contain:

- a FARM class, specialisation of the GEOOBJECT, which may be further specialised into classes such as CATTLE RANCH, AGRICULTURAL FARM and NON-PRODUCTIVE PROPERTY.
- a RURAL CADASTRE class, specialisation of CADASTRALMAP, which includes the graphical definition for the objects of the FARM class and its specialisations.
- a SOILMAP class, specialisation of the THEMATICMAP class, whose instances are geo-fields that store the different soils map for the study area.
- the classes ALTIMETRYMAP, SLOPEMAP and ASPECTMAP, specialisations of the DIGITALTERRAINMODEL class, whose instances are geo-fields that store altimetry, slope and aspect maps for the study areas.

4.2 Simplifying Function selection

Use of the model concepts has enabled the design of an user interface which allows manipulation of geographical data at an abstract level. When a user selects an *infolayer* which has been specialised into one of the classes of the database schema, only the operations available for the specific type of data are made available to him. This approach reduces to a large extent uncertainty in the choice of valid functions.

An infolayer can have more than one graphical representation, allowing raster and vector derivations of the same abstract entity. When a map analysis operation is requested, the GIS could know to which type of representation that operation is most conveniently applied and thus make the necessary conversions. The user would be relieved of the duty to deal directly with vectors or rasters, but would

refer to "combination of the Land Use map to the Slope map".

By contrast, many raster-based systems fail to distinguish between thematic and DTM operations and express operations in terms of "integer grids" and "floating-point grids". Actually, boolean operations on the so-called "integer grids" are thematic map functions, where arithmetic operations on "floating-point grids" are DTM functions.

5. MODEL APPLICATION: SYSTEM DESIGN

The model has been applied to the design and implementation of SPRING. The system (whose development language is C++) combines functionality for image processing, geographic analysis, digital terrain modelling with a spatial data base. SPRING is the result of the co-operation between INPE (the Brazilian National Space Institute) and the IBM Rio Scientific Centre, with help from EMBRAPA (the main agricultural research organisation in Brazil). Its first product version has been released in early 1994. For further details on SPRING, please refer to Câmara et al. (1992).

The user interface for SPRING reflects the model concepts, and real-life experience with users has shown a very favourable learning curve a high degree of user satisfaction with the usability of the system.

Table 1 shows the mapping between the conceptual and representational levels, as used in the SPRING system.

TABLE 1

MAPPING BETWEEN THE CONCEPTUAL AND REPRESENTATION LEVELS IN SPRING

Real-world data	General classes	Graphical Represent.

Soils Map	Thematic maps (fields)	Vectors w/topology (or) raster images
Parcels Map	Cadastral (objects)	Vectors w/topology arc-node-region
LANDSAT image	Remote Sensing (fields)	Raster images
Altimetry Geophys.	DTM (fields)	regular grids (or) TINs (or) samples (or) raster images (or) contour lines
Power network	Networks (objects)	Vectors w/arc-node topology

At the implementation level, the SPRING class library is a such general tool for building GIS systems. It contains methods for dealing with the various types of geographical data, organised according to the built-in data types.

REFERENCES

- CÂMARA, G., FREITAS, U., SOUZA, R., CASANOVA, M., and HEMERLY, A., 1992. "SPRING: Object-oriented Geoprocessing", in *Proc. International Symposium on Photogrammetry and Remote Sensing*, Washington, DC.
- EGENHOFER, M. and FRANK, A., 1989. Object-Oriented Modeling in GIS: Inheritance and Propagation". *Proc. Ninth International Symposium on Computer-Assisted Cartography*, Baltimore, MD, 1989, pp. 588-598.
- ERTHAL, G., CÂMARA, G. and ALVES, D.S., 1988. A General Data Model for Geographic Information Systems. *Proc. International Symposium on Photogrammetry and Remote Sensing*, Kyoto, Japan, 1988.

GOODCHILD, M. 1992. Geographical data modeling. *Computers & Geosciences*, 1992, Vol. 18, No.4, pp. 401-408.

GOMES, J.M.; VELHO, L. "Modeling in Computer Graphics". Short course, VI *Brazilian Symposium on Computer Graphics and Image Processing*, Recife (Brazil), 1993 (in Portuguese).