DATA MODELLING AND DEVELOPMENT OF A SPATIAL DATABASE FOR THE BRAZILIAN AMAZONIA

Gilberto Camara[1], Ubirajara Freitas[1], Ricardo C.M. Souza[1],
Marco A. Casanova[2], Andrea S. Hemerly[2]

[1]Image Processing Division
National Institute for Space Research
P.O. Box 515
12.201, S. José dos Campos, SP - Brasil

[2]Rio Scientific Center
IBM Brasil
P.O. Box 4624
20.001, Rio de Janeiro, RJ - Brasil

ABSTRACT

A conceptual framework for modelling environmental data, which caters for a diversity of data sources and formats, is first described. Then, the data model adopted in the design of an object-oriented image processing and geographical information system, called SPRING, and its associated spatial database management system, called EnvDATA, is introduced. The framework may in fact be viewed as an abstraction of the basic characteristics of the data model. Finally, a specific environmental database storing facts about the Brazilian Amazonia, implemented using the SPRING system, is briefly outlined.

KEY WORDS: Data Modelling, Geographic Information Systems, Geo-referenced Databases

1. INTRODUCTION

Environmental studies require that Earth be seen as a whole. The understanding of the interactions between the oceans, the land and the atmosphere has to be reflected in the design of the next generation of geographical information systems and environmental databases. Current trends include a staggering growth of data sources (including NASA's Earth Observing System) and on the complexity of environmental studies. The new software systems should be able to integrate, in a single data base, spatial information from diverse sources, including meteorological numerical weather and climate prediction, remote sensing imagery, geophysical and geochemical data and census data.

The design of systems that can cope with the great variety of environmental data requires data models that cater for the nature of environmental data and provide a framework for a single environment that integrates the various data formats. The need for conceptual advances on data modelling is a major research direction and has been so stated on the NCGIA research plan [11,12].

Such motivation led Brazil's Institute for Space Research (INPE), in collaboration with the IBM Rio Scientific Center, to design a system for environmental data visualisation and manipulation (SPRING) and a spatial database management system for large sets of environmental data (EnvDATA) [5].

In this paper we first introduce a conceptual framework for modelling georeferenced databases. Then, we describe the data model adopted in the current implementation of the SPRING/EnvDATA system. Finally, we outline a specific environmental database storing facts about the Brazilian Amazonia. The database, called AMAZONIA, is being implemented using the SPRING system.

The conceptual framework abstracts the basic characteristics of the SPRING/EnvDATA data model. But we believe that the framework is generally useful when designing geographic information systems since it clearly points out some interesting problems related to data modelling in this area.

The SPRING/EnvDATA data model is based on earlier work by the same group [2,6,10] and also serves as a basis for a meteorological visualization and manipulation system (METVIEW), being developed by INPE in cooperation with the European Centre for Medium Range Weather Forecasts [5]. An important goal for the model has been to enable the combination of an image processing system with vector-based GIS analysis methods. This can only be achieved by breaking the traditional vector-raster dichotomy [9]. Most efforts on data modelling for georeferenced databases focused on vector representations (points, arcs and regions) [1,4,8].

The need for integrating images and digital terrain models in SPRING/EnvDATA has led to a more general formulation. The concepts of the model are implemented as classes and methods in SPRING, thus deferring the choice of the most convenient data structures to later implementation stages.

2. A CONCEPTUAL FRAMEWORK FOR GEOREFERENCED DATABASES

A *georeferenced database* is basically a set of *geographic objects* or, abbreviatedly, *g-objects.* A g-object may be *complex*, that is, composed of an aggregation of other g-objects, and the g-objects of a database may be organized along a specialization hierarchy. For example, we may specialize the g-objects of a database into maps, digital terrain models (DTMs), satellite scenes and spectral bands, and consider the satellite scenes as complex objects composed of spectral bands.

The attributes of a g-object are classified as *intrinsic* and *extrinsic*, the latter being those that pertain to the objects the g-object represents. For example, the intrinsic attributes of a satellite scene may be defined as a timestamp, an orbit/point and a cloud coverage. As another example, a city map may have as intrinsic attribute a set of polygons (its geometry) and as extrinsic attributes the name and population of the city it represents.

Each g-object must always have among the intrinsic attributes one or more *geometries*, that are always georeferenced according to some specific scheme. This requirement is in fact what distinguishes georeferenced databases from ordinary databases. For example, the geometry of a spectral band may be a matrix of pixels. The geometry of a g-object may itself be complex, that is, composed of other geometries.

However, in this paper, we do not recognize the geometries as objects independent from the g-object they are associated with.

The definition of extrinsic attributes for a class of g-objects, although convenient, may create problems. Indeed, if a real world object is represented more than once, its attributes will be either redundantly represented in several g-objects, or arbitrarily associated with a single g-object.

To circumvent this problem, we first introduce *real world objects* or *r-objects* as a second major type of objects. Just as the g-objects, the r-objects of a database may be organized as a specialization hierarchy, they may be complex and they may have attributes.

We also introduce the *represents* relationships, or *r-relationships*, that relate g-objects and r-objects. For example, let C be a r-object that stands for a city, with its population, mayor, etc... as attributes. Let G be a g-object, that is, a map, satellite scene, etc... Then, we capture that G somehow depicts C by introducing a r-relationship between G and C. Thus, the attributes of the city will not be redundantly associated with its many representations, but stored just once in C.

Ideally, if an r-object O is represented by a g-object G, as explicitly defined by an r-relationship between O and G, it should be possible to decide if another g-object G′ is a representation of O or not since G and G′ are both georeferenced. In practice, this ideal is not attenable because of the computational cost of comparing G and G′ and the limitations pertaining to the geometries of G and G′ (errors, precision problems, scale differences, etc...). As a consequence, the set of r-relationships may be hybrid, in the sense that it is partly stored and partly computed, with the latter carefully defined to avoid the above pitfalls.

This concludes the general remarks about the modelling of georeferenced databases.

3. ORGANIZATION OF DATA IN THE SPRING SYSTEM

This section discusses some features of the SPRING system, including the overall data organization, the built-in classes of g-objects, the types of complex objects supported and how data is accessed.

Object Repository and Directory

The current version of the SPRING system supports the concepts of r- and g-object and a restricted form of the *represents* relationship, where each g-object can be a representation of at most one r-object. The system stores all r- or g-objects of all databases in an *object repository.*

The SPRING system also has a *directory*, whose entries essentially describe the classes of g-objects and r-objects defined for each database, along with their attributes, and a set of finite domains, defined by enumeration. The definition of object classes and finite domains is fundamental to the SPRING system since it induces a methodology to handle the data and permits the design of a user-friendly interface.

Built-in Classes of G-Objects

SPRING offers the following built-in classes of g-objects:

- thematic model: used for geographic regions defined by one or more polygons, where each region corresponds to a different theme. A thematic model is a complex object whose components are nodes, arcs, polygons and centroids. Thematic maps store topological information, which is important in spatial analysis. The geometries supported are either vector or raster.
- DTM (digital terrain model): describe the spatial distribution of a physical variable (such as temperature or wind fields). The geometries supported are vector, raster, rectangular grid or triangular grid.
- Network: linearly connected structures with arc-node topology, that store information about flows between different geographical locations. The geometry supported is vector.
- Image: store data generated by satellites (e.g., GOES, METEOSAT, TIROS and LANDSAT), by radar or by operations in other images. The geometry supported is vector.
- Observational model: locational entities, which indicate the existence of a punctual phenomenon in space. These data may represent measurements made in meteorological data collecting stations, historical climatological information, digital terrain samples and geophysical and geochemical measurements. The geometry supported is vector.

Complex g-objects

SPRING offers two types of complex g-objects, called *information layers* (or *infolayers*) and *projects*, informally defined as follows.

Each georeferenced database in SPRING is partitioned into projects. A project corresponds to a geographical area of interest containing a set of data to be analyzed. A project is an aggregation of information layers, where each infolayer represents a distinct type of information for a given geographical area on a specific time instance.

The infolayer is a powerful abstraction that enables the user to concentrate on the information and hides the different graphical representations. As an example, an infolayer that contains terrain elevation information may have the following graphical representations: vector (isolines), regular or triangular grid (resulting from interpolation or triangulation) and raster (a gray-level image). The information itself is the same, the changes are only in its presentation.

An infolayer has spatial, non-spatial, spectral and temporal attributes. Examples of infolayers are:

- A LANDSAT TM scene, acquired at a given date/time, defined as an aggregation of 7 spectral bands;
- A vector wind field, composed of horizontal and vertical components;
- A temperature 3D field, covering various layers of the atmosphere, for a given time instance;
- A thematic map, perhaps obtained by classification of a raster image.

An infolayer will be an aggregation of as many g-objects as it is possible to individualize. For example, an infolayer composed of a 7- band unclassified LANDSAT image contains 7 g-objects (the spectral bands). An infolayer consisting of a classified image that has been converted to the vector format contains as many g-objects as there are different classified areas.

In the current implementation, the same g-object cannot be a component of more than one infolayer. In addition, the geometries of all g-objects in the same infolayer must refer to same geographic area. These restrictions facilitate the problem of accessing g-objects, but they introduce the familiar problems created by fragmenting the space into disjoint regions [3].

Interface description

An important concern of the interface is how the user identifies the data. The internal complexities of the model should be hidden, and the user should be able to distinguish the nature of the database components.

Upon entering the SPRING system, each user starts a working session and typically selects a database and a project to work with, based upon the classes of objects defined. In general, he may invoke functions to:

- create a new database;
- select a working database;
- select a project;
- select data from a project;
- import data to the current project from another source;
- analyse/visualize g-objects;
- produce cartographic documents;
- save the current database;

The SPRING system also offers functions to import data from outside sources, such as that created by the SGI system, an older system also implemented at INPE.

4. THE AMAZONIA DATABASE

One of the major applications being implemented on top of the SPRING/EnvDATA system is a database, called AMAZONIA, that contains facts about the Brazilian Amazonia, focusing on deforestation data. Briefly, the final database will contain:

Infolayers: information layers associated with g-objects (6.000)

Images: recent images of the Amazon region - 3 bands only - 60m precision (700 instances, or 6.5 Ggytes)

Grid: topographic and climatological grids - 1:1.000.000 scale (400 instances)

Thematic: thematic g-objets distributed as follows:

- Total deforestation: 150.000
- Annual Deforestation: 750.000
- Vegetation: 10.000
- Draining: 30.000
- Roads: 1.500
- Geomorphology: 15.000
- Pedology: 15.000
- Political Division: 1.500

(a total of approximately 1.000.000 instances, or 1.1 Gbytes):

4. CONCLUSIONS

We briefly described in this paper a conceptual framework to organize data in georeferenced databases and the data model implemented in the SPRING/EnvDATA system. We also briefly outlined the profile of the AMAZONIA database, an important effort to capture and organize environmental data about a very large and important eco-system.

REFERENCES

[1] D.J.Abel et alii, EDSS: A Prototype for the Next Generation of Spatial Information Systems, Proc. AURISA 91, Wellington, New Zealand, Nov. 1991, pp. 66-77.

[2] D.S. Alves, Data Models for Geographic Information Systems, PhD. Thesis. University of Sao Paulo, 1990. (in Portuguese).

[3] Aronson, "The Geographic Database - Logically Continuous and Physically Discrete", in Proc. Ninth Interational Symposium on Computer-Assisted Cartography, Baltimore, MA, Apr. 1989, p.452-461.

[4] D. Benoît, "Modélisation, représentation et gestion d'information géographique", Ph.D Thesis, University of Paris 6, France, 1991.

[5] Camara,G.; Daabeck, J. "SPRING/WS: The conception of a new system for visualization of meterological and climatological data ", in Proc. 7th. International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hidrolology, New Orleans, LA, Jan. 1991.

[6] Camara, J.R. Oliveira, F.Y. Yamaguchi, F.R.D. Velasco, R.C.M. Souza, "Abstract Data Types in Computer Graphics and Image Processing", in Proc. II Brazilian Congress on Software Engineering, Canela, Brazil, October 1988 (in Portuguese).

[7] M. Egenhofer and A. Frank "Object-Oriented Modeling in GIS: Inheritance and Propagation", in Proc. Ninth Interational Symposium on Computer-Assisted Cartography, Baltimore, MD, 1989, pp. 588-598.

[8] M. Egenhofer,"Spatial Query Languages", Ph.D. Thesis, University of Maine, USA, 1989.

[9] M. Ehlers, G. Edwards, I. Berard, "Integration of Remote Sensing with Geographic Information Systems: A Necessary Evolution", Photogrammetry Engineering and Remote Sensing, vol. 55, no. 11, pp. 1619-1627, Nov. 1989.

[10] G. Erthal, G. Camara and D.S. Alves, "A General Data Model for Geographic Information Systems", in Proc. International Symposium on Photogammetry and Remote Sensing, Kyoto, Japan, 1988.

[11] National Center for Geographic Information and Analysis, "The Research Plan for the NCGIA", International Journal of Geographic Information Systems, vol. 3, no.2, pp. 117- 136, 1989.

[12] T.R. Smith, S. Menon, J.L. Star and J. Estes, "Requirements and principles for the implementation and construction of large-scale geographic information systems", International Journal of Geographic Information_ Systems_, vol.1, no. 1, January-March 1987, pp. 13-31.