

# **Towards a Geographic Semantic Database Model**

by

**Martin Feuchtwanger**

B.Sc. (Hons.), University of East London, England, 1982

M.Sc., University of Calgary, Alberta, 1985

Thesis Submitted in Partial Fulfillment of  
the Requirements for the Degree of  
**Doctor of Philosophy**  
in the Department of  
**Geography**

© Martin Feuchtwanger 1993

**Simon Fraser University**

July 1993

All rights reserved. This work may not be  
reproduced in whole or in part, by photocopy  
or other means, without permission of the author.

## APPROVAL

Name: Martin Feuchtwanger  
Degree: Doctor of Philosophy  
Title of Thesis: Towards A Geographic Semantic Database Model  
Examining Committee:  
Chair: M.C. Roberts,  
Professor

---

T.K. Poiker, Professor,  
Senior Supervisor.

---

A.C.B. Roberts, Associate Professor

---

Wo-Shun Luk, Professor  
School of Computing Science

---

Brian Klinkenberg, Assistant Professor,  
Department of Geography,  
University of British Columbia  
Internal Examiner

---

Michael Goodchild, Professor,  
Department of Geography,  
University of California, Santa Barbara  
External Examiner

Date Approved: July 12, 1993

## PARTIAL COPYRIGHT LICENSE

I hereby grant to Simon Fraser University the right to lend my thesis, project or extended essay (the title of which is shown below) to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users. I further agree that permission for multiple copying of this work for scholarly purposes may be granted by me or the Dean of Graduate Studies. It is understood that copying or publication of this work for financial gain shall not be allowed without my written permission.

### Title of Thesis/Project/Extended Essay

Towards A Geographic Semantic Database Model

---

---

---

---

### Author:

\_\_\_\_\_  
(signature)

Martin Feuchtwanger

(name)

12 July 1993  
(date)

# Abstract

The management of geographic data is a great problem in contemporary cartography. To date, little theory has been developed to assist such a task. This thesis proposes a geographic semantic database model — a concept for the design, construction, and use of geographic databases. The work involved the synthesis of both general semantic database concepts and specific geographic information concepts.

A logical database model incorporates notions of the structural and behavioral aspects of stored information. Structurally, a database contains entities, relations, and attributes. Behaviorally, a database has queries and transactions. Database models are evolving from syntactic to semantic forms, representing greater ability to directly and easily model reality.

Any things of interest in geographic data processing can be called phenomena. A phenomenon exhibits three primary characteristics: topical, spatial, and temporal. That is, it has some identification and position, and exists at some time. Information on phenomena thus exists within three characterization domains. It also exists within three abstraction domains: generalization, realization, and construction. That is, geographic data have some accuracy and resolution, some form between reality and concept, and a level of meaning or applicability. The characterization and abstraction domains are the particularly geographic ways for logically partitioning a collection of data.

The proposed geographic database model contains entities, such as features, profiles, layers, and composites, which represent geographic phenomena. The entities are characterized by topical, spatial, temporal, and scale attributes, and by semantic, topologic, and abstraction relations to other entities. They can be retrieved, displayed, or updated by database manipulations comprising selections and actions. The entities also exist at different levels of abstraction: at different scales, appropriate for different levels of investigation; in analytic or graphic form, depending on whether they are to be

used for machine or visual processing; and as applied, basic, or primal constructs, appropriate for different levels of use.

Combining concepts from database management and analytical cartography into a geographic database model not only facilitates the analysis and design of geographic databases but also is a step towards a general theory of geographic information management and analysis.

# Acknowledgment

Intellectual guidance was provided by Tom Poiker, Arthur Roberts, and Wo-Shun Luk.

Considerable financial support was provided by Henry and Joan Feuchtwanger, England; Simon Fraser University, B.C.; The Canadian Department of Energy, Mines, and Resources; and The University of Calgary, Alberta. Additional financial support was provided by The Spatially Oriented Referencing Systems Association; The Canadian Department of National Defence; and Intergraph Canada.

To all who have helped me in any way, I thank you.

# Table of Contents

Approval	ii
Abstract	iii
Acknowledgment	v
Table of Contents	vi
List of Tables	ix
List of Figures	x
<b>1 Introduction</b>	<b>1</b>
1.1 Background	1
1.1.1 Geographic Information Systems	1
1.1.2 Database Systems and Database Models	2
1.1.3 Geographic Information	4
1.1.4 Geographic Databases	5
1.2 Purpose and Objectives	6
1.3 Overview	7
<b>2 Review of Related Work</b>	<b>12</b>
2.1 General Database Models	12
2.1.1 Relational Model	12
2.1.2 Functional Model	14
2.1.3 Entity-Relationship Model	15
2.1.4 Hypergraph-based Data Structure	15
2.1.5 PROBE	16
2.1.6 Object-oriented Semantic Association Model	17
2.2 Geographic Data Models	18
2.2.1 Nyerges' Cartographic Information Structure	18
2.2.2 Goodchild's Geographic Analytic Data Model	19
2.2.3 UCSB's Knowledge-based Geographic Information System	20
2.2.4 Germany's Authoritative Topographic-Cartographic Information System	21
2.2.5 Alves' Object-oriented Geographic Data Model	21
2.2.6 BC's Spatial Archive and Interchange Format	22
2.3 Discussion	23
<b>3 Logical Database Models</b>	<b>25</b>
3.1 Introduction	25
3.2 Database Systems	26
3.2.1 Database System Basics	26
3.2.2 Database System Goals	27
3.2.3 Database Models	29
3.3 Database Model Statics	30

3.3.1	Objects	31
3.3.2	Entities	32
3.3.3	Attributes	32
3.3.4	Relations	33
3.4	Database Model Dynamics	36
3.4.1	Database Operations	36
3.4.2	Primitive Operations	37
3.4.3	Queries and Transactions	38
3.4.4	Data Manipulation Languages	39
3.5	Further Database Model Concepts	39
3.5.1	Selection	40
3.5.2	Constraints	40
3.5.3	Dependency and Redundancy	41
3.5.4	Attribute-Mapping Dependency	41
3.5.5	Computational Dependency	42
3.5.6	Abstraction	43
3.5.7	Encapsulation	44
3.5.8	Relativism	44
3.6	Database Model Use and Implementation	45
3.7	Database Model Types	46
3.8	Conclusion	48
<b>4</b>	<b>Geographic Data Concepts</b>	<b>50</b>
4.1	Introduction	50
4.2	The Geographic Characterization Domains	51
4.2.1	Elemental Data Types	51
4.2.2	The Topical Domain	52
4.2.3	The Spatial Domain	54
4.2.4	Structure of the Spatial Domain	54
4.2.5	Operations within the Spatial Domain	62
4.2.6	The Temporal Domain	65
4.2.7	Combinations of Domains	67
4.2.8	Topical-Spatial Functions	68
4.2.9	Topical-Temporal and Spatial-Temporal Functions	70
4.2.10	Layer Combinations	71
4.2.11	Summary	76
4.3	The Geographic Abstraction Domains	76
4.3.1	The Generalization Domain	77
4.3.2	Variability of Scale	79
4.3.3	Recording and Presentation of Scale	80
4.3.4	Scale and the Characterization Domains	81
4.3.5	The Realization Domain	81
4.3.6	Relations Between Levels of Realization	83
4.3.7	Realization and the Characterization Domains	84
4.3.8	The Construction Domain	85
4.3.9	Relations Between Levels of Construction	87
4.3.10	Construction and the Characterization Domains	88



4.3.11 Summary	89
4.4 Conclusion	89
<b>5 A Geographic Semantic Database Model</b>	<b>108</b>
5.1 Introduction	108
5.2 GSDM Entities	108
5.2.1 The Feature Entity	109
5.2.2 The Profile Entity	111
5.2.3 The Layer Entity	111
5.2.4 The Composite Entity	112
5.2.5 The Graphic Map Entities	113
5.2.6 Summary	113
5.3 GSDM Attributes	113
5.3.1 Topical Attributes	114
5.3.2 Spatial Attributes	115
5.3.3 Temporal Attributes	116
5.3.4 Generalization Attributes	116
5.3.5 Summary	117
5.4 GSDM Relations	117
5.4.1 Semantic Relations	117
5.4.2 Topologic Relations	118
5.4.3 Abstraction Relations	119
5.4.4 Summary	119
5.5 GSDM Operations	120
5.5.1 Selections	121
5.5.2 Actions	123
5.5.3 Queries and Transactions	123
5.5.4 Summary	124
5.6 Conclusion	125
<b>6 Discussion</b>	<b>138</b>
6.1 GSDM v. General Database Models	138
6.2 GSDM v. Geographic Data Models	140
<b>7 Conclusion</b>	<b>144</b>
7.1 Summary of Research	144
7.2 Implications of Proposed Model	145
7.2.1 Geo-Database Specification, Analysis, and Design	146
7.2.2 Geo-DBMS	146
7.2.3 Geo-Conceptual Modelling	146
7.3 Possible Future Projects	147
7.3.1 Model Development	147
7.3.2 Model Testing	148
7.3.3 Model Comparison	148
7.3.4 Model Implementation	149
<b>Bibliography</b>	<b>150</b>
<b>Glossary</b>	<b>165</b>

# List of Tables

1 Database Models	11
4.1 Geo-referencing	92
4.2 Raster Data Structures	92
4.3 Layers	93
4.4 Layer Management	93
6.1 General Database Models	142
6.2 Geographic Data Models	143

# List of Figures

1.1 Geographic Information System	9
1.2 Database System	10
4.1 Hierarchy	94
4.2 Class Relations in a Hierarchy	94
4.3 Interlocking Hierarchies	94
4.4 Class Relations in a Lattice	94
4.5 Spatial Incidence	95
4.6 Spatial Adjacency	95
4.7 Spatial Overlap	95
4.8 Spatial Enclosure	95
4.9 Hierarchical Spatial Partitioning	96
4.10 Region Relations in a Hierarchy	96
4.11 Exhaustive and Exclusive Spatial Partitioning	96
4.12 Region Relations in a Lattice	97
4.13 Non-exhaustive and Non-exclusive Spatial Partitioning	97
4.14 Topological Geo-referencing	98
4.15 Hierarchical Geo-referencing	98
4.16 Coordinate-based Positioning	98
4.17 Measurement-based Positioning	98
4.18 Plane Coordinates	98
4.19 Geodetic Coordinates	98
4.20 Temporal Contiguity	99
4.21 Temporal Intersection	99
4.22 Discrete Layer	100
4.23 Continuous Layer	100
4.24 Continuous History	100
4.25 Discrete History	100
4.26 Composite Layer	100
4.27 Independent Layers and Redundancy	101
4.28 Metrical Overlay of Two Layers	102
4.29 Topological Overlay of Two Layers	102
4.30 Scale Dependence	103
4.31 Scale-dependent Dimensionality	103
4.32 The Real World	104
4.33 Mental Map	104
4.34 Graphic Map	104
4.35 Analytic Map	104
4.36 Relations Between Levels of Realization	105
4.37 Many-to-One Applied-to-Basic Relations	106
4.38 One-to-Many Applied-to-Basic Relations	106
4.39 Several Features may Share the Same Geometry	106

4.40 One-to-Many Basic-to-Primal Relations	107
5.1 Elements of a GSDM Schematic	126
5.2 Generic Feature	127
5.3 Compound Feature	127
5.4 Heteroplethic Feature	127
5.5 Heteropleth Types	127
5.6 Profile Types	128
5.7 Compound Profile Components	128
5.8 Layer Types	129
5.9 Compound Layer Components	129
5.10 Composite Types	130
5.11 Compound Composite Components	130
5.12 Primal Attribute Types	131
5.13 Topical Attribute Components	131
5.14 Spatial Attribute Components	132
5.15 Spatial Extent Types and their Components	132
5.16 Spatial Vector Object Types	133
5.17 Vector Network	133
5.18 Spatial Raster Object Components	133
5.19 Temporal Attribute Components	134
5.20 Temporal Extent Types	134
5.21 Temporal Vector Object Types	135
5.22 Temporal Raster Object Components	135
5.23 Generalization Attribute Components	136
5.24 Map Scale Attribute Types	136
5.25 Spatial Topology Attribute Components	137
5.26 Temporal Topology Attribute Components	137

## Chapter One

# Introduction

This thesis presents a modern approach to managing data within geographic information systems.

## 1.1 Background

### 1.1.1 Geographic Information Systems

A *geographic information system* (GIS) is a complex but integrated collection of hardware, software, data, principles, and procedures for collecting, processing, analyzing, archiving, or displaying data or information of interest on the natural and human phenomena existing in particular regions of the world. Although a relatively new discipline, the study of such systems has evolved to the stage that there are now several texts available [Aranoff 1989; Burrough 1986; Clarke 1990; Maguire *et al.* 1991; Star & Estes 1990; Tomlin 1990]. Potential applications of a GIS can be found in fields such as geography, forestry, agriculture, exploration, urban and regional planning, marketing, municipal engineering, and tourism. Potential users of a GIS include managers, planners, scientists, administrators, and ordinary citizens. The technology of a GIS relies heavily on technologies within geography and computing, such as cartography, geodesy, remote sensing, database systems, software engineering, and computer graphics.

The components of a GIS can be described as being a network of technical subsystems and a large archive of related geographical data, or geo-data (Fig. 1.1). The technical subsystems include an *interaction system*, a *geo-data collection and input system*, a geo-

*data manipulation and analysis* system, a *geo-data storage and retrieval* system, and a *geo-data output and display* system. Because a GIS will likely have much more data than will be used at any one time, data must be held on long-term storage devices and retrieved and held in short-term memory when needed. Thus, the storage and retrieval subsystem is of vital importance.

Although both geographic and computing technologies have progressed to a state of maturity, their integration to enable GISs to meet the increasing demands for advanced geographic information management and analysis tools still has significant hurdles to overcome [NCGIA 1989; Rhind 1988; Tomlinson & Boyle 1981]. This thesis attempts to bridge the gap between geography and computing by providing a common language, or model, for the storage and retrieval of geo-data.

### **1.1.2 Database Systems and Database Models**

The function of data storage and retrieval in most information systems is provided by a *database system* [Date 1985; Korth & Silberschatz 1991]. It essentially consists of three things (Fig. 1.2):

- a large collection of related data held on a mass storage device, i.e., a *database*;
- a software package that defines the structure of and controls access to the stored data, i.e., a *database management system* (DBMS); and
- various programs that give the information system usefulness in a particular setting, i.e., *applications*.

Traditionally, databases have been oriented towards relatively simple kinds of business applications and have been based upon relatively simple filing techniques.

What distinguishes different DBMSs is a guiding principle or theory. The conceptual framework, or theory, that dictates the number and type of different classes of storable data, their relations, and the query and altering operations that are available is called a *database model*. It is a kind of algebra for data.

It is important to distinguish two broad categories of database models (Table 1), *logical*

and *physical*. They differ according to their level of abstraction, that is, how closely they refer to the real-world concepts of an application domain, such as geology, or to the concrete details of the underlying computer technology. Logical database models are oriented towards human understanding and help us specify what is going on, what data or information is stored, and what can happen to it. Physical database models are oriented towards machine concepts and specify how the data are stored and retrieved at the computer system level.

Logical database models themselves can be further categorized according to their *level of abstraction*. The earlier ones are relatively low-level and are called *datalogical*, *syntactic*, or *record-based* models because they deal with files, records, fields, and links. They include the conventional hierarchical, network, and relational database models. The newer models are at a higher level of abstraction and are described as being *infological*, *semantic*, or *object-based* because they deal with sets, objects, attributes, and relations. These include the functional, semantic, object-oriented, and logic-based database models.

The problem with conventional database models is their limited semantic expressiveness [King & McLeod 1985]. The employment of overly simple data structures to model a complex application environment inevitably involves a loss of information. Record-based models fail to distinguish different generic kinds of entities and relations and often have complicated query languages. Conventional DBMSs place much of the burden of performing data management tasks, such as integrity maintenance, on separate application programs that ought to be doing other things. It leads to a duplication of effort and to program-data dependence, which are contrary to the goals of database systems. Hierarchical and network DBMSs, for example, tend to force a user to navigate through a series of physically defined records and links in order to access data, rather than allow a user to simply state what information is required. They also tend to impose a single, rigid view of information, rather than allow alternative perspectives. And a relational DBMS, while having no physical links between records, still forces a user to specify complex links when accessing information.

A semantic database model is one that is relatively high-level because it allows data to more accurately and directly reflect real-world objects or concepts [King & McLeod 1985]. It defines both the structural and behavioral characteristics of stored data — structure being defined in terms of objects, attributes, and relations, and behaviour in terms of query and transaction operations. Also, integrity constraints are more easily definable. These semantic data concepts will be explained further in §3.

### **1.1.3 Geographic Information**

The problem with GISs as opposed to ordinary information systems is that their requirements are more difficult to meet. Business information tends to consist of names, descriptions, prices, dates, etc., and can be represented using simple records of words or numbers. It also tends to be one-dimensional, meaning that each name, description, price, or date can be expressed by a single value drawn from a linear array of possible values. Geographic information on the other hand is multi-dimensional, multi-temporal, of varying quality, and voluminous. It can have complex structural properties and processing problems. A useful GIS will hold information on many themes within a region, such as topography, geology, vegetation, population, economics, engineering, and utilities — each theme concerning different kinds of continuous variables and discrete features (e.g., areal, linear, surficial). Not only is geographic information thematically multi-dimensional, but it is spatially three-dimensional; everything is positioned somewhere in the world. Also, because of the nature of most geographic phenomena and limitations in measuring and processing, geo-data often have varying degrees of uncertainty associated with them. And to suit different scales of investigation, data on the same regions should be available at different levels of detail. Finally, the world is dynamic and there will often be the requirement to study changes over time, so another dimension is added to the problem.

By the foregoing, I do not mean to imply that the distinction between geo-data and all other types of data is black and white. Rather, I mean to emphasize the complexity of geo-data versus the relative simplicity of the data in many non-geographic information



systems. In fact, certain other scientific or engineering data can be equally complex, and the temporal requirement can apply to many non-geographic cases.

#### 1.1.4 Geographic Databases

Just as a GIS is a special kind of information system, consider a *geographic database* as being a special kind of database that holds geographically related data, and a *geographic database management system* (geographic DBMS) as being a special DBMS that controls the nature and content of a geographic database.

A useful geographic DBMS should provide a means for designers to define the template of a geographic database and for the users to retrieve and update geographic data. It should know what data items are allowed to exist, how they may be related, and how they may be manipulated. Ideally, a geographic DBMS would be able to handle many problems: large volumes of data, possibly distributed over several sites; numerous and complex geographic entities, relations, operations, and integrity constraints; data uncertainty; multiple scales of representation; multiple classes of users; a highly graphical interface; fast access to a two or three dimensional spatial domain; and concurrent access by many users [Frank 1984; Frank 1988].

The science and technology of analyzing, designing and implementing geographic DBMSs is relatively new. There has been little or no conceptual framework developed for them. The development of geographic DBMSs has so far taken place on an *ad hoc* piecemeal basis. Two important factors help explain the lack of adequate geographic databases. First, designers have been constrained by low-level record-based data models. Such models require considerable user intelligence and effort to manage meaningful geographic objects. Also, their access mode is one-dimensional, resulting in slow responses for accesses in the two-dimensional geographic domain. A conventional DBMS, for example, will allow records to be accessed based on the provision of a name, a number, or a department — all 1D attributes — whereas a geographer might require access based on a position in 2D space. The second major impediment to geo-DBMS development is (as outlined in §1.1.3 above and elaborated upon in §4) that geographic

information is relatively complex when compared with most non-geographic information, and is thus difficult to conceptualize.

Different approaches have been taken in developing geographic DBMSs. Some have built special-purpose DBMSs, from the ground up, which can efficiently handle spatial data [Davis & Hwang 1986; Frank 1984; Kleiner 1989; Matsuyama *et al.* 1984; Nyerges & Smyth 1983; Samet 1984; Schek & Waterfield 1986; Tamminen 1984]. Others have added a spatial component to existing DBMSs [Abel 1988; Abel & Smith 1986; Hagan 1981; Laurini & Milleret 1989; Lorie & Meier 1984; Waugh & Healey 1986]. In most cases emphasis has been on physical or datalogical aspects of spatial databases, rather than semantic, or infological, aspects. Other important geographic data issues, relevant to geographic databases, have been or are being explored in detail, though not necessarily within a database context. Concepts of subdivision and coordination of space [Beatty 1980; Maling 1992], of scale and uncertainty [Jones & Abraham 1987; Miller *et al.* 1989], and of temporality [Frank *et al.* 1992; Langran 1989] have been developed, but in an isolated rather than integrated manner.

Since geographic data can be so complex the appropriate data model should be able to handle such complexity so as to make them simpler. There appears to be a need for models that are both semantic and geographic [Bouillé 1978; Digital Cartographic Data Standards Task Force 1988; Dueker & Kjerne 1989; Feuchtwanger 1985; Feuchtwanger 1989a; Norris-Sherborn 1984; Nyerges 1980; Salgé & Schlafer 1989; Shapiro & Haralick 1980; Tuori & Moon 1984].

## **1.2 Purpose and Objectives**

The purpose of the thesis is to develop the framework for a geographic semantic database model by integrating appropriate concepts from logical database modelling and geographic data processing.

The thesis has many objectives and constraints. The objectives are that the thesis is

- specifically geographic, geared to GIS, not to MIS (management information

systems), CIM (computer integrated manufacturing) or other information systems in business or industry; geography offers a unique set of challenges;

- meant to be broadly applicable to GIS problems, not concerned with any particular geographic theme or application;
- aimed at storage and retrieval of information, not its manipulation and analysis, collection and input, display and output, or user-interaction;
- about a model of data not of an actual region; no sample datasets will be examined;
- conceptual, not experimental; no prototype system is built;
- infological, not physical; it concerns meaningful objects and operations, it does not address hardware or software aspects such as storage efficiency or access speed;
- oriented towards databases not knowledge-bases; there is little on expert system rules;
- a synthesis rather than an analysis; it integrates many ideas, it does not investigate any particular issue to the depth being done by others; and
- comprehensive, dealing with issues of descriptive, locational, and historical data, with issues of scale and uncertainty, with different kinds of maps, and with alternative encoding structures.

The constraints exist to keep the project down to a manageable size and to avoid duplicating work done by those mentioned above.

### **1.3 Overview**

The body of the thesis report is in five parts.

The first (§2) is a brief review of some existing general purpose database models and of some other researchers' concepts of geographical data processing.

Next, in §3, is a synthesis of the important high-level concepts involved in general-purpose databases. It describes database objects that are further categorized into entities, attributes, and relations, and database operations that can be performed on objects.

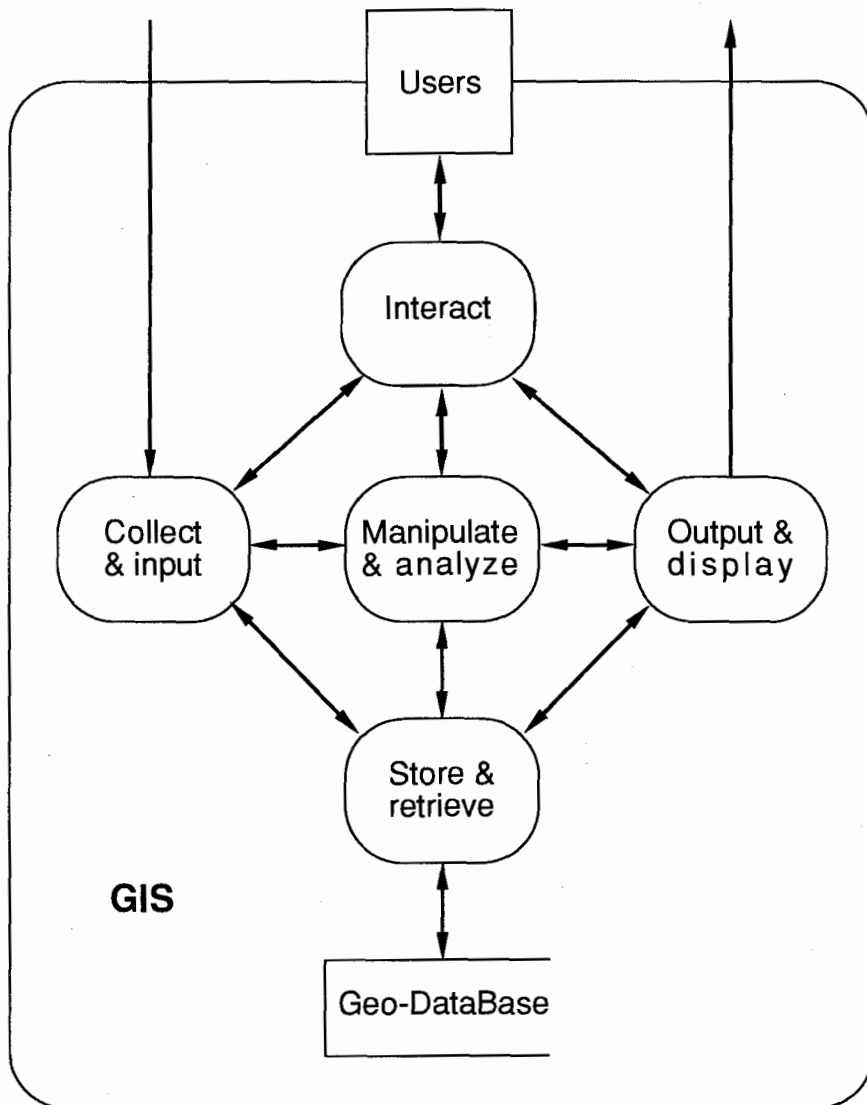
Other advanced concepts such as abstraction, relativism, and selection are also presented.

The third main part, §4, is a look at the particular requirements of geographic data storage and retrieval. The different kinds of geographic phenomena and information and how they are viewed and logically organized are examined within different domains. Three domains for characterizing geographic data — topic, space, and time — and three domains for abstracting geographic data — scale, realization, and construction — are described.

In §5 the outline of a geographic logical database model is presented, in terms of geographic entities, such as the feature, profile, and layer, and in terms of their geographic attributes, relations, and operations, within the characterization and abstraction domains.

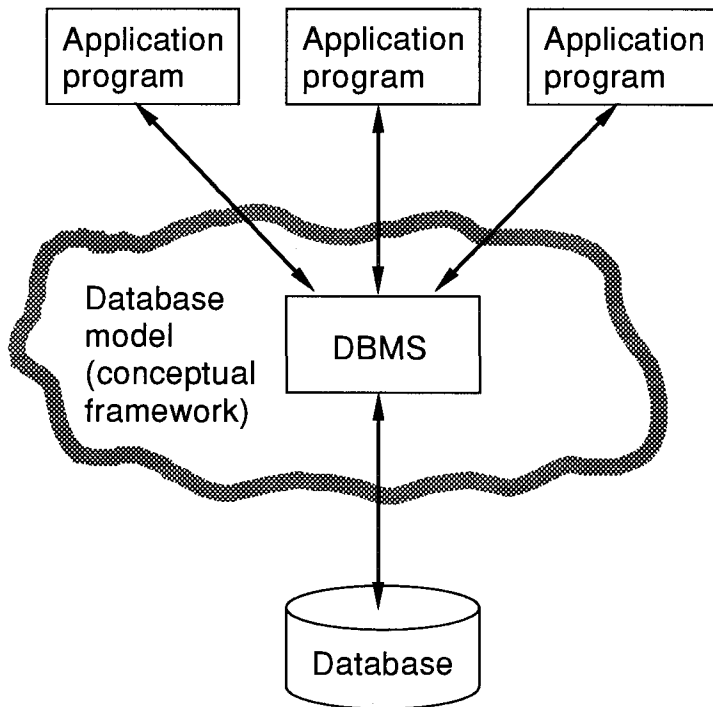
Finally, in §6, the proposed model is briefly compared with existing ones.

The report contains many technical terms. Some are in common use in the GIS literature while others are specific to this thesis. Most are included in the glossary, following the list of references.



**Figure 1.1 Geographic Information System**

The components of a generic GIS include its users, five sub-systems performing different geo-data processing functions, and a geo-database. Geo-data flow between the components.



**Figure 1.2 Database System**

Various application programs access a database via a database management system, according to concepts in a database model.

**Table 1 Database Models**

<b>Level of Abstraction</b>	<b>Type</b>	<b>Subtype</b>	<b>Examples</b>	<b>Typical Components</b>
high (human)	logical (specification) "what"	semantic (infological)	entity-relationship semantic functional object-oriented	entity relation attribute operation
		syntactic (datalogical)	relational network hierarchical	file record field pointer
low (machine)	physical (implementation) "how"		sequential random	disk cylinder track block

## Chapter Two

# Review of Related Work

This chapter is intended as a review of a sample of other people's work on two types of data modelling — database modelling in general and geographic data modelling in particular. Projects or models are chosen because each appears to have significantly similar objectives to my own, as a whole. Other research and development contributions that relate to parts of mine are not described here, but they are referred to where relevant in subsequent chapters. Note that many of the terms used here are explained in §3 and §4 and in the Glossary, while others are specific to the section where used.

## 2.1 General Database Models

A number of general-purpose database models, of differing levels of abstraction, have been proposed. Six of them are briefly reviewed here. The more important concepts are synthesized and elaborated upon in §3.

### 2.1.1 Relational Model

Perhaps the best-known database model is the relational one. Numerous detailed descriptions exist [Bradley 1981; Brodie 1986; Codd 1970; Date 1985; Korth & Silberschatz 1991; Martin 1976; Ullman 1988]. It has been implemented as a number of commercially available DBMS products — Oracle, dBase, Ingres, Sybase, etc.

The relational database model is based on mathematical principles of "relations" and "tuples." According to the model, a database consists of a number of different relations



which are essentially table-like files of data. Each relation is a set of tuples, of similar types, where each tuple is essentially a row of data in the table. For example, a cadastral database might have tables for land parcels, city blocks, roads, and easements, and each row in one of the tables represents a single parcel, block, road, or easement.

A relation's logical structure is defined by a group of attributes, where an attribute is essentially a column heading in the table. All attributes are atomic, meaning they are single, non-decomposable units. Each attribute belongs to an attribute domain which defines the data type (typically these are Binary, Integer, Real, Character, String, and Date) and perhaps the allowable range of values recordable for that attribute. So, the attributes for the Land-parcel table might be Parcel-ID, Boundary, Area, Owner, Block, Value, Use, etc., while attributes for the Road table might be Road-ID, Width, and Construction. Links between two tables are allowed when they have attribute domains in common. A row in the Parcel table is related to one in the City-block table, for example, if the Parcel table's Block attribute corresponds to the City-block table's Block-ID attribute.

Operations on a relational database are performed using a declarative data manipulation language. Central to the specification of new relations and the processing of existing ones is the "select" statement which identifies particular occurrences and combinations of tuples. Getting all the values of parcels owned by Smith might require a statement like this:

```
select Land-parcel [Value] where Land-parcel [Owner = "Smith"].
```

Although, in theory any data can be stored in a relational database, it has been debated that in practice it is only efficient for relatively simple applications, requiring alphanumeric data [Glen 1992; Hammer & McLeod 1975; King & McLeod 1985; Rybeck 1990]. In GISs it is often used for certain data in conjunction with a specialized spatial database to provide an overall geographic database facility [Morehouse 1985; Westwood 1989]. Extensions to the relational model have been proposed — including

adding more complex data types and more complex data operators [Codd 1979; Laurini & Milleret 1989; Loitsch 1991; Lorie & Meier 1984; van Oosterom & Vijlbrief 1991].

### 2.1.2 Functional Model

The functional database model is a little-known model based on mathematical principles of "functions" [King & McLeod 1985; Norris-Sherborn 1984; Shipman 1981]. Its unique characteristic is that entities, attributes, relations, and operations are all modelled by functions or derived functions.

A functional database consists of sets of objects, which are either entities or attributes, and functions, which are mappings between objects. Derived functions are inversions, compositions, and restrictions of other functions.

Each type of entity, such as a land parcel or a city block, is represented by a zero-argument function declaring it as such. Subtypes of entities are declared in a similar manner. An attribute of an entity is represented by a single-argument function, mapping an entity onto a data type. For example, a Width attribute maps a Road entity onto the Numeric data type, and an Area maps a City-block onto Numeric. Relations between entities are represented by single- or multi-argument functions between the entities. A Block relation, for example, maps a Land-parcel entity onto a City-block entity, while a Parcels relation maps a City-block onto the inverse of the Block relation. Derived attributes may be defined by function composition. For example, the Block-area attribute maps City-block onto the sum of the Areas of the Parcels of the City-block. Retrieval operations are formulated in the same manner that entities are declared. For example, specifying

print Value (Land-parcel) where Owner (Land-parcel) = "Smith"

would get the values of parcels owned by Smith.

The model has been extended experimentally to include a limited number of spatial data

types and operations, such as point, line, area, distance, inside, intersects, and enclosure [Norris-Sherborn 1984].

### 2.1.3 Entity-Relationship Model

The entity-relationship (E-R) model is a well-known, informal, high-level database model and has been described in detail in many places [Chen 1977; 1980; 1983; 1985]. It is intended primarily as a logical database design tool and is not implemented as a DBMS product.

Essentially, the E-R model consists of "entities," "relationships," and "attributes." It has no data operation components. Entities are items about which information is held, such as city blocks, land parcels, and roads, and each entity is described by a group of attributes. Attributes, such as area, width, use, and type, are of the usual data types (real, integer, string, etc.). Relationships are logical connections between entities and are represented explicitly. For example, a city block *contains* a parcel, and a person *owns* a parcel. If a relationship is to have attributes, such as the date a person gains ownership of a parcel, it becomes a composite entity. Some extensions of the E-R model allow for sub-types of entities (i.e., they allow generalization hierarchies), such as if a historical parcel is a type of parcel.

The E-R model is perhaps the first of the high-level data models and the easiest of all data models to understand. It has been used in a geographic context [Feuchtwanger & Poiker 1987; Tuori & Moon 1984] and a special geographic extension has been proposed [Bédard & Paquette 1989].

### 2.1.4 Hypergraph-based Data Structure

The Hypergraph-based Data Structure (HBDS) is claimed to be both a high- and low-level database and knowledge-base model, based on mathematical concepts of "sets" and "graphs" [Bouillé 1978; 1984; 1986; Bouillé & Rugg 1983; Satharanond 1981].

Unfortunately, descriptions of the HBDS are often rather complicated and it is little understood. This is partly because the descriptions unnecessarily contain mixtures of both types and individual occurrences of database elements.

Like the E-R model, the HBDS consists of entities, relations, and attributes. Sub-types of entities may exist and these inherit the properties of their super types. Attributes can be either stored directly or computed using procedures that derive values from others. Any database element can have conditions constraining how it is accessed, or rules comprising computational procedures that are initiated if certain conditions are met.

The HBDS has been used for geographic applications based on networks of points, lines, and regions, and has been implemented as an experimental DBMS [Bouillé 1984].

### **2.1.5 PROBE**

PROBE is a very high- and low-level database model for complex applications [Dayal & Smith 1986; Orenstein 1986; Orenstein & Manola 1988]. It has a number of interesting capabilities.

PROBE supports abstraction and encapsulation, that is, it allows various levels of data types — primitive ones and compounds of primitives — and the operations or procedures associated with each type are packaged along with them.

It supports recursive operations, i.e., operations that are defined partially in terms of themselves. Suppose land parcels can be made up of smaller land parcels, which can be made up of other parcels, and so on. Then an operation to determine a parcel area would be defined as "determine the areas of the components" and the problem repeats until the areas of the smallest parcels are determined some other way.

PROBE also allows "spaces" of multiple "dimensions" to be defined. So, a 2D or 3D cartesian space (X,Y or X,Y,Z) can be defined, or even a 4D space-time (X,Y,Z,T)

"space." Primitive data types include Point, Interval, and Region and these may exist within the "spaces." (A Region being a multi-dimensional interval in a space.) Operators associated with these types include Overlaps, Precedes, and Contains.

### 2.1.6 Object-oriented Semantic Association Model

The Object-oriented Semantic Association Model (OSAM\*) is a very high-level experimental database model, designed for complex applications [Su 1986; Su *et al.* 1988; Alashqur *et al.* 1988]. It appears to incorporate aspects of the E-R model, HBDS, and PROBE, above, and makes a significant contribution to my thesis.

OSAM\* consists of several different kinds of entities, attributes, relations, operations, and rules. Generally, entities are described by attributes and by other entities. They may also be classified into sub-types and super-types, allowing generalization lattices. The Land-parcel entity may, for example, be described by ID, Area, Value, Use, etc., attributes, and by its City-block entity. A sub-type of Parcel may be Historical-parcel and this may also be a sub-type of a Historical entity. Other, special types of entities are allowed, including composite entities that involve a relation between other entities, and entities that summarize groups of other entities.

Attributes can be atomic or compound, and stored or computed. Compound attributes are defined by aggregations, sets, lists, or arrays of other attributes. For example, Location is an aggregation of Easting and Northing; Occupants is a set of Persons; Boundary is a list of Locations; and Image is an array of Pixels. Operations are performed by declarative queries, e.g.,

context Land-parcel (Owner = "Smith") retrieve Value  
would get the values of parcels owned by Smith.

OSAM\* has been described in a geographic context [Feuchtwanger 1989b] in more detail.

## **2.2 Geographic Data Models**

A number of reasonably comprehensive models of geographic or cartographic data have been proposed. Six of them are briefly reviewed here. The more important concepts are synthesized and elaborated upon in §4.

### **2.2.1 Nyerges' Cartographic Information Structure**

Nyerges defined a high-level cartographic database model and implemented it as an experimental DBMS [Nyerges 1980]. While the model is based heavily on linguistics concepts and its emphasis is on maps as digital graphics rather than as analytical constructs, it was one of the first high-level geographic data models.

The model recognizes general database concepts, such as entity types and instances, database schemas and views, attributes, relations, and operations. There are a number of primitive and compound entities, the compounds being collections of primitives. Example primitives include the Point, Pixel, and Symbol, numerous types of Lines, and the Polygon, while compound examples include Title, Scale, Grid, Legend, and Feature. Entities are characterized by attributes and relations.

The model recognizes topical, spatial, and graphical characterizations: for example, a Feature is characterized as being a Road, at a particular Location, and as being represented by a red Tone. Different levels of cartographic generalization, or scales, are recognized, as are analytic and graphic forms of entities. A Feature, for example, is an analytic entity because it represents something in the world, while a Symbol is a graphic entity because it represents something on a map. The model recognizes application specific, general, and primitive levels of software abstraction. A Road is an application specific example of a general Feature and a Line is its primitive form.

### 2.2.2 Goodchild's Geographic Analytic Data Model

Goodchild defined an informal, medium-level geographic data model, in support of spatial analysis, rather than storage and retrieval [Goodchild 1987a; 1987b].

The model consists of four primitive map features — Node, Arc, Polygon, and Pixel; compound features, which are relations between pairs or triplets of other features; and continuous surfaces, or map coverages, such as an irregular set of points and a regular array of pixels. It also consists of attributes, describing the features. The attribute domains are Nominal, Ordinal, Interval, and Ratio, according to the scales of measurement used in geography, instead of the data types used in computing.

Features have a fixed number of generic "reserved" attributes, and any number of "other" attributes, specific to an application. Primitives have both "geometric" reserved attributes (e.g., a Polygon's Perimeter, Area, and Centroid), and "topologic" reserved attributes (e.g., node-arc-polygon adjacencies). Other attributes of a primitive might be City, Population, Industry, etc. For compound features the reserved attributes are the two features involved in the relation, while other attributes might be Distance and Flow-rate. While not explicitly supported, features with a more complex structure, such as a path or tree of links, can be implied via careful use of attributes.

Four basic classes of spatial analysis that examine attributes of features are described. There are those analyses that require (i) the topical attributes of one class of features, (ii) the topical and spatial attributes of one class of features, (iii) the attributes of compound features, and (iv) the attributes of many classes of features. There are also two classes of spatial analyses that create new features: (i) those that create new types of features from one or two existing types, and (ii) those that create new compounds of one or more existing types.

### 2.2.3 UCSB's Knowledge-based Geographic Information System

The Knowledge-based Geographic Information System (KBGIS) is an experimental high- and low-level geographic database and knowledge-base system developed at the University of California at Santa Barbara [Menon & Smith 1989; Peuquet 1984; Smith & Pazner 1984]. It has two complementary components, topical and spatial, and is apparently capable of inductive and deductive learning.

The topical side consists of a network of inter-linked "property trees," i.e., topical attribute domains are arranged in a generalization lattice. Each node in a property tree has a name and a data type, e.g., Land-cover, Forest, and Waterbody are of the Nominal type, while Elevation and Precipitation are of the Numerical type. A node is linked to a number of sub-nodes if the sub-nodes represent mutually exclusive and exhaustive attribute values of the parent, such as when Land-cover can be broken down into Forest, Waterbody, Snowfield, Pasture, etc. An alternative node-subnode linking is when the parent class consists of a combination of attribute values, such as a node High-forest being linked to the nodes called "Land-cover = Forest" and "Elevation > 500m."

The spatial component of the database consists of a series of "spatial trees" which are quadtrees (i.e., four-way, hierarchical raster structures). The coverage for each map variable is a quadtree, where each node is recursively subdivided into four more nodes until its topical content is uniform. Also recorded at each node are "spatial spectrum" data, indicating the clustering and scattering of its component nodes.

One interesting thing about this model is the way entities are represented. They seem to be represented by a "meeting of the tree leaves!" An entity is defined by topical, locational, and topological attributes, the first two corresponding to the leaf nodes of the property and spatial trees. The topical attribute is a series of primitive properties, linked by the "and" or "or" conjunctions, such as "Elevation>500 and Forest=Pine or Forest=Spruce." The locational attribute is a set of pixels, corresponding to leaves in the



spatial trees. The topological attribute indicates spatial relations, such as Contains/Inside, Near, Far, Neighbor, North, and East, to other entities.

Database queries allow for the retrieval or display of the locations having a given property, the property at a given location, or the properties within a given rectangle.

#### **2.2.4 Germany's Authoritative Topographic-Cartographic Information System**

The Authoritative Topographic-Cartographic Information System (ATKIS) is a nationwide GIS research and development project of the Federal German Republic State Surveys Working Committee [Hesse & Leahy 1991]. Its purpose is to be a data archive and transfer standard, rather than a database model.

ATKIS consists of digital landscape models (DLMs), which are analytic maps, and digital cartographic models (DKMs), which are graphic maps. The models exist at three scales — 1:25 000, 1:200 000, and 1:1 000 000. Each DLM contains a tri-level type hierarchy of themes, compound entities, and entities. Themes include Survey control points, Settlement, Transportation, Vegetation, Waterways, Areas, and Relief. Compound entities have topical and spatial attributes, component entities, and relations to other entities. Basic entity types include Point, Line, Area, Raster, and Mixed. Topical attribute types are Quantitative and Qualitative. The spatial attribute can be a Vector or a Raster. No recursiveness is supported. Relations may be 1:n, n:1, 1:2, or 2:1. Each DKM contains symbols, or graphic entities, which are cartographically transformed topographic objects.

#### **2.2.5 Alves' Object-oriented Geographic Data Model**

Alves described a high-level, object-oriented, geographic data model [Alves 1990], i.e., one that supports abstraction and encapsulation.

The model consists of analytic entities, which represent real-world objects, such as

roads, rivers, and climate stations, and have topical and spatial attributes. Two types of analytic entities are maps and features. A map is a compound entity comprising a set of features of a particular theme. The model recognizes that each real-world object may be mapped in a multitude of projections and scales, thus its spatial attribute can have multiple values. Six types of spatial attributes for maps, called "geometries," are supported: Polygonal map, Image, Grid, Point map, Contour map, and Triangular mesh. Each "geometry" is composed of "elements," such as Point, Line, and Polygon; is manipulated by operations, such as Display; and can be manipulated as an indivisible unit (by a DBMS or other user). Each of a map's features is related to the elements from its geometry.

The same set of features can be combined with any number of geometries, e.g., towns on one map may be represented as points, while on another as polygons. A continuous surface, such as a terrain, can be represented by a map whose geometries may take on many allowed forms, i.e., image, grid, point map, contour map, or triangular mesh, and whose operations may include value and contour interpolation, volume calculation, etc.

The model also consists of graphic entities — visualizations of analytic entities — including five primitive classes: Point, Curve, Area, Text, and Bitmap.

### **2.2.6 BC's Spatial Archive and Interchange Format**

The Spatial Archive and Interchange Format (SAIF) is a high-level geographic data model for data exchange and archiving, developed by the Ministry of Crown Lands, Province of British Columbia [SAIF 1991a; 1991b; Sondheim 1991]. Although it is formally described it is not implemented as a DBMS; it is a data transfer standard.

It consists of objects whose structure and behavior are defined. Examples include Integer, Boolean, String, Topical taxonomy, Relation, and Feature. Objects belong to classes, or types, having similar structure and behavior. Classes often have subclasses and superclasses. Object instances can be constrained to take one of a finite set of

possible values. A compound object is made up of other objects according to a nested structure, allowable structures being Record, List, and Set. Aggregation defines the internal structure of an object, while generalization defines subtypes of objects. Subtypes inherit all characteristics from supertypes. A feature is a compound object representing any real or artificially defined entity which can be referenced through a coordinate system. It has a "geometry," an ID, and an topical attribute. The "geometry" describes the kind of spatial primitive (e.g., Point, Line, or Polygon), its location, associated topology, and other spatial relations. The ID amounts to a simple identification from a standardized set of feature classes. Only existing kinds of geometry and features may be used. The topical attribute provides additional descriptions according to a user-defined taxonomy. A composite feature is one that is composed of other features of the same or differing types. Its geometry is a grouping of the component geometries.

An interesting feature of SAIF, considering that it is a data transfer standard, is that it supports encapsulation, i.e., the constraints and manipulations that apply to an object are considered an integral part of the object's class definition.

## **2.3 Discussion**

The model proposed in this thesis represents an amalgamation of various aspects of the above models as well as certain additional concepts, as discussed in the following two chapters.

How suitable are the general database models, described above, for managing geographic information? Because they all include, in some form or another, general entity, attribute, and relation types, special (geographic) types can be defined for a geographic database. However, one should not be required to re-define such types for each problem just because the model does not include them explicitly. As database design tools, I believe the relational and functional models are not easy to use because of

their limited number of explicit constructs. While they are based on mathematical principles, such formally defined models are less relevant to my thesis, because I am developing a framework for, rather than implementing, a database model. The slight increase in the number of constructs provided by the E-R model and the HBDS, and their diagrammatic nature, commends them to easier database analysis and design in general, while the ability to define multiple dimensions of information in PROBE particularly commends it to geometric problems. Finally though, because of its rich set of high-level informational constructs that enable one to analyze and design databases that closely match an application environment, and its diagrammatic nature, OSAM\* is most relevant to the rest of this thesis.

How suitable are the geographic data models, described above, for guiding the storage and retrieval of data in a GIS? All of the models appear to be suitable, at least as a basis, for development into high-level database models. (Goodchild's "analytic data model" is perhaps the exception because of its analytical rather than archival focus.) The most stimulation for this thesis came from Nyerges' "cartographic information structure" and UCSB's KBGIS because of their relatively early development and their encompassing of so many specifically geographic concepts. While my work appears quite similar to Germany's ATKIS, Alves' "object-oriented geographic data model," and BC's SAIF, it has evolved somewhat simultaneously with and independently of them and is even more encompassing of geographic concepts. For example, a weakness of all of the models but SAIF is their lack of explicit temporality. Conversely, all models but SAIF and Goodchild's allow for multiple scales of representation. Finally, in this thesis, the number of different types of features and relations provided in all the geographic data models will be expanded to include one-dimensional profiles and three-dimensional composites.

## Chapter Three

# Logical Database Models

### 3.1 Introduction

Assume that an information system is a useful working model of aspects of some portion of the real world. It is built out of a database system for managing long-term data and a collection of application programs for data processing and user interfacing. As the demands placed upon the information system become more taxing (e.g., larger data volumes, greater numbers of users, and more complex applications), the requirement for better database and programming techniques becomes more pressing.

The conceptual framework for the design, construction and use of a database is known as a *database model*. Database work may be facilitated if the database model can capture the semantics of the real world situation naturally and directly [Borgida 1986]. Within a database model, the world may be comprehended in terms of entities, relations, and activities, with associated attributes and constraints. Such models vary according to the closeness with which they may represent reality.

This chapter is intended as a review and synthesis of the important concepts of logical databases, such as objects, schemas, and operations, which concern what information is recorded and what happens to it. Physical aspects, i.e., those that concern how databases are actually implemented, are not covered. It is a combination of material from standard database texts, semantic database literature, and my own thoughts. All concepts are illustrated using examples from a geographic point of view, although a comprehensive discussion of geographic data specifically is reserved for the next chapter.

## 3.2 Database Systems

Simply put, a database system consists of a *database* (i.e., a large, organized collection of data) and some hardware, software, and users. The basics and goals of both database systems and database models are detailed in many texts [Bradley 1981; Brodie *et al.* 1984; Brodie & Myopoulos 1986; Date 1985; Korth & Silberschatz 1991; Martin 1976; Ullman 1988; Yao 1985]. Some are explained briefly here for completeness.

### 3.2.1 Database System Basics

The software that manages a database is a *database management system* (DBMS). It provides a common and controlled means of accessing the data and is used directly or indirectly by all database users. It is a more concrete implementation of a (more abstract) *database model*.

The design of a database, i.e., a blueprint allowing a particular content, is known as the *database schema*. It is the template within which the data will be slotted, and it remains relatively constant over the life of the database. The content of a database, i.e., a set of related data, is known as the *database view*. It will vary over time as the data keep pace with the changes in the world outside.

Commonly, a database is said to exist at three *levels of abstraction*. At the lowest level is the *internal* (or *physical*) *database*. It is the most concrete, in terms of computer hardware, but the least sensible form of the database, in human terms. Above the physical level, which is not further considered, are the two *logical* levels, concerned with information or data content. At the intermediate level is the *conceptual* (or *main*) *database*. It is a complete, integrated dataset seen by the entire community of users. Different portions of the main database are the *external databases*, at the highest level of abstraction. These are subsets and derivations of the main database and are seen by

individual groups of users. Each user group may have a different view of the data for particular application areas.

To illustrate the above concepts, suppose there is a *cadastral* information system. At the heart is a cadastral database holding a variety of data on land parcels. Anyone who accesses the information does so via a DBMS. The database schema specifies that parcels, roads, land uses, building codes, leases, property boundaries, utilities, etc., are all allowed to exist and have certain characteristics. The database view is the set of all the particular pieces of information on the roads, buildings, and other entities, at any instant. It exists at three levels of abstraction. The conceptual database is the main database, holding all the information as a whole, for the community of users. Different types of information on parcels, say, are of interest to different users — surveyors are concerned with lot boundary measurements and owners; planners want to know about census data on inhabitants; and engineers need the locations and specifications of utilities — and so are held in different external databases. Inside the computer system the information is held as pieces of data stored in the internal database on magnetic disks.

### 3.2.2 Database System Goals

All database systems attempt to achieve certain goals of logical performance and functionality. These are usually some or all of the following:

*Integration.* Several separate but related files are combined into one unified whole, a database.

*Integrity.* The database is complete, correct, and consistent because there is protection against illegal or inappropriate operations being performed on data. For example, no two parcels of land should overlap and no parcel should have more than one owner.

*Controlled redundancy.* Data are not duplicated, at least not in such a way that inconsistency is possible. For example, the fact that parcel X is used for purpose

Y should not be recorded in more than one place. If it were recorded in a second place and the fact were changed at one of the places, the database would be inconsistent. Some redundancy is necessary, however, for reasons of efficiency in data access. If so, it is controlled.

*Independence.* Programs are independent of the way data are stored, and data are independent of the way programs are implemented. Changes to one will not affect the functionality of the other. A more efficient spatial search algorithm, for example, should not affect the identity number of a parcel.

*Abstraction.* The same information exists at different levels of complexity. Details of data viewed at one level are hidden from those at the next higher level. The internal, conceptual, and external levels of abstraction embody this important concept.

*Database language.* A unified means of specifying both the structure of the database and the operations performed on the database is provided. The database language is part of the DBMS.

*Evolvability.* The database structure and content can evolve with changing knowledge or specifications of the application environment without any loss of integrity. As more municipal departments become part of the cadastral information system, the database schema is expanded to include the new types of information with minimal impact on the current setup.

Other goals are more physical in significance:

*Efficiency.* Speedy access to the data is provided. Also, the data take up minimal storage space.

*Concurrency.* Several users may access the same data at the same time. A handful of different citizens may each be getting data on a neighborhood block without being aware of one another.

*Distribution.* One large data set is stored or made available at several different sites. The database is physically located in different municipal departments but appears as a logical whole. Also, those citizens browsing the database may each



be located in different parts of the city.

*Security.* Data are protected against unauthorized access and against loss due to system failure. Access to confidential personal information is restricted to those with proper authority. Also, backup copies of the database are kept in case of fire.

### 3.2.3 Database Models

All DBMSs are devised within some paradigm for storing and retrieving data. The paradigm, known as the database model, can be described as having two broad aspects, *static* and *dynamic*, and as belonging to two broad camps, *syntactic* and *semantic*.

Matters concerning what data can be in a database are referred to as the *structural* (or *static*) aspects of the database model, and those concerning what can happen to the data are referred to as the *behavioral* (or *dynamic*) aspects of the database model. In the cadastral example, database model statics concern parcels, zones, utilities, etc., and their attributes and inter-relations, while database model dynamics concern retrieval of and changes to data, such as parcel subdivision and transfer, building applications, and utility repairs.

There is a clear distinction between "reality," what is happening in the real world, and the "database," what is happening in the information system. In reality there are very complex phenomena, things and events, while in a database there are data — recorded facts about some of those phenomena. As database models evolve, attempts are made to conceptually reduce some of the differences between reality and the database.

The earlier and still the most popular database models, sometimes referred to as *record-based*, or *datalogical* models are syntactic in the sense that their physical structure is a dominant feature. Examples include the hierarchical, network, and relational database models. They tend to have constructs reflecting the underlying computer or data processing system. They manage data in terms of files, records, fields, and pointers. The

newer and still evolving database models, sometimes referred to as *object-based*, or *infological* models, are semantic in the sense that they tend to have more meaning. Examples include the object-oriented, semantic hierarchical, functional, binary relational, and entity-relationship database models. Their constructs tend to reflect the application environment being served by the computer system. They manage data in terms of object classes, entities, attributes, and relations, and attempt to represent in a more direct manner the phenomena of interest in the world.

One thing to note is that although two extremes have just been described, there is really a continuum of modelling along which database models are evolving. The relational model, for example, has undergone various extensions to make it more semantic [Codd 1979].

The remainder of this synthesis of database modelling concepts describes the semantic or object-based paradigm. Note that whenever terms like objects or entities are used it should be assumed that it is always digital data that are stored in the database and not the actual things that they represent!

### **3.3 Database Model Statics**

The structure of a database is the concern of the database model's *statics* [Borgida 1986; Brodie 1984; Brodie 1986; King & McLeod 1985; Su *et al.* 1988]. It is specified as a schema by means of a data definition language (DDL), often a subset of the database language. Ideally, the state of the database should reflect the state of the world that is being modelled by the database. This section describes database information as being composed of various different objects, i.e, entities and their attributes and relations.

### 3.3.1 Objects

Any feature, item, or concept of potential interest or significance — and which can be recorded in a database as information or data — is a *database object*. Examples of such objects might include district, town, road, year, crop, yield, and cost. A distinction is often made between an *entity*, an *attribute*, and a *relation* as different kinds of database objects. Something that exists, has an identity and a description, is an entity, such as a city or university. Something that serves only to describe an entity, is an attribute, such as population or wealth. Something that links or associates two or more objects, is a relation, such as the link between a school and a school district.

Two fundamental forms of object are *types* and *instances*. A set of all possible objects that may exist and have similar characteristics is referred to as an object type, or class. Examples of object types might include Highway and City. A particular occurrence of an object is called an object instance. Individual instances of the above types might include Highway 7 or 66 and the City of Vancouver or Hull. The relation between instances and types is called *classification*, where instances are classified into types. (Alternatively it is called *instantiation*, as in Ottawa 31 G/5 is an instance of Topographic-map-sheet.) Although instances and types have been distinguished, and are strictly very different concepts, often for simplicity, the word "object" will be used for either concept and the meaning will be clear from the context. Generally a database contains objects, but specifically the database schema will consist of some object types and the database view will consist of a very large number of object instances.

Based on complexity, two fundamental classes of objects are often distinguished, *primitive* and *compound*. An object that is not considered to be composed of any other object is a primitive, or atomic, object. It is treated as an individual item of information and is operated upon as a single unit. An object that is somehow composed of other objects is a compound, or nonatomic, object. Such may be the case when a university, a complex object, is made up of several buildings and other simple objects.

An important concept in semantic data modelling is that there is a one-to-one correspondence between a phenomenon in the world and an object in the database. No matter how complex the phenomenon is, it can always be represented by one database object. Of course, a complex object has to be composed of, or be described by, other objects via attributes and relations, and such "concretization" continues until everything is in the form of attribute values, i.e., digital data. A telephone pole, a river, a width, a region, an ownership, and a metropolis, for example, may all be database objects, even though they may not all be printable directly as values.

### 3.3.2 Entities

An object that has some independent significance in the database is an *entity*. It is characterized by some descriptive or statistical attributes and by some relations with other entities. It can only be accessed, created, or deleted via its attributes or relations. Entity types in a topographic database might include roads, rivers, built areas, forested areas, control points, and benchmarks.

### 3.3.3 Attributes

Any useful property or characteristic of an entity, such as a name, height, or weight, is generally termed an *attribute*. Two fundamental forms of attribute are *values* and *domains*. A particular instance of an attribute is termed an attribute value. The attributes Crop-type or Depth may take on values of "wheat" or 5.4, respectively. The possible range or set of values that an attribute may take is termed an attribute domain. For example, "Mon, Tue, ... Sun," make up the domain for the Day attribute, and Month comprises "Jan, Feb, ... Dec" (or 1, 2, ... 12). Attribute domains may be subject to further restrictions called *constraints*. Depending on an attribute's semantics, domain elements will be restricted to plausible values, e.g., Temperature may be restricted to range from -40 to +30.

The two main types of attributes are *numeric* and *symbolic*. They can represent the two kinds of geographic data, qualitative and quantitative, respectively. Subtypes include the data types commonly found in programming languages — real, integer, character, boolean, etc. Another useful data type, perhaps, is the BLOB (binary large object), a long string of bits that may be any complex data type in encoded form.

Attributes may also be deemed to have *single* or *multiple* values. A multi-valued attribute takes a group of values of a similar type. For example, a position may have two components, eastings and northings, or a spectral signature in remote sensing may have four or more components.

Any attribute that is used to specify some subset of entities of a particular type is an *identifier* for that type. An identifier may be a combination of attributes, such as a unique parcel number (UPN) comprising plan, block, and lot numbers. A special type of identifier that uniquely specifies the entity among the set of entities is called a *key*. Commonly, an entity will have a name as an identifier and a serial number as a key. Two important types are distinguished, and they are what I call *internal* and *external* keys. An internal key is assigned by and known to the system only, not the user. In a sense, it is a physical not a logical database element. An external key is a unique identifier useful to and accessible by its users.

### 3.3.4 Relations

Any meaningful connection, association, or grouping of objects is termed a *relation*. Marriage, membership, ownership, adjacency, and subdivision are all examples of relations between two or more entities. (Note that "relation" here is not the same as the "relation" of the relational database model, a table-like file structure.) Relations have either *structural* or *semantic* forms.

A structural grouping of data elements and links into a compound item is termed a *data*

*structure*. There are many different data structures and they can be found in the programming literature [Aho *et al.* 1982; Koffman 1988]. Examples include the set, a variable-sized group of similar items; the list, an ordered set; the record, a fixed-size group of different items; the array, a fixed-size list; the tree, a hierarchically organized set; and the network, a cyclically organized set. Also included are more specialized structures, such as the triangulated irregular network (TIN) used in terrain modelling [Peucker *et al.* 1976].

A structural grouping that also has other defined characteristics is termed a *semantic relation*. Various ones have been identified [Brodie 1984; Egonhofer & Frank 1989; King & McLeod 1985; Su *et al.* 1988], and four common ones are now described:

*Generalization*. Some object types can be further classified into subtypes. For example, fir, spruce, and pine are subtypes of the more general type conifer, and transportation line may be a super-type for the special types highway, railway, and waterway. The relation between the general type and the special types is a *generalization* or *specialization*, depending on the viewpoint. All subtypes are assumed to have some characteristics in common. In fact they are said to inherit them from the super-type. Such *inheritance* means that, for example, if the general type region is defined to have an area then all special types of region also have an area as well as their own specific characteristics. Generalization, or its converse, specialization, is sometimes called a "has-subtype" or an "is-a" relation, respectively.

*Aggregation*. A compound object that is characterized or described by other objects of different types is an *aggregate* object. A city, for example, can be defined as a name, a location, and a population. The relation between the aggregate object and the components is an aggregation. It is commonly used to relate an entity to its attributes. It is sometimes called a "has-a" or a "part-of" relation, depending on the viewpoint.

*Composition.* A compound object that is composed of several other objects of the same type is a *composite* object. A province, for example, is composed of a group of districts. The relation between a composite object and the member object is a composition, an association, or a partitioning. It is sometimes called a "set-of" or a "member-of" relation, depending on the viewpoint. The fact that characteristics of the composite object, such as population mean and standard deviation, depend on those of constituent objects, is called *propagation*.

*Interaction.* A compound object may represent some other linkage or relation between two or more objects. In this case the relation is said to be an *interaction*. For example, land ownership represents an interaction between a land parcel and a land owner, and two roads may cross at an intersection. It is sometimes known as an association or simply as a "relationship" since it is the most generic kind of relation.

An important property of any relation is the number of instances of each related type which may participate in an occurrence of the relation. It is known as the *cardinality*, or connectivity, of the relation. Commonly, cardinalities of relations are one-to-one, one-to-many, and many-to-many, often written 1:1, 1:m, and m:m, respectively. For example, marriage is a one-to-one relation, genus-species is a one-to-many relation (because a genus includes many species), and region-species is a many-to-many relation (because a region supports many species and a species inhabits many regions).

In the general case, however, a more complete specification of cardinality would indicate the number of instances of each related type which are either *required* or *allowed* to participate in the relation. In other words, the cardinality must include the lower and upper limits to the number of entity instances involved on each side of a relation occurrence. For example, the cardinality of a lot-block relation may be 0-20:1-1, meaning that between zero and twenty lots belong to only one block; or a node-link

relation may be 2-2:0-m because a link must join exactly two nodes and a node may join zero or more links.

### 3.4 Database Model Dynamics

The behavioral, or operational, aspects of a database, i.e., the retrieval of data from the database and the transitions from one database state to another, are the concern of the database model's *dynamics* [Borgida 1986; Brodie 1984; Brodie 1986; King & McLeod 1985; Su *et al.* 1988]. The dynamics of the database model are the mechanisms for manipulating objects, attributes, and types. Dynamics are specified via *operations* using a data manipulation language (DML), another part of the database language. In theory, the same concepts that applied to statics apply here, i.e., there are operations, properties of operations, and relations between operations. Also, the dynamics attempt to model activities of the application world. However, in practice there has been less emphasis placed on developing semantically expressive dynamics. Much more of an information system's dynamics are handled outside of the DBMS by application programs than are its statics.

#### 3.4.1 Database Operations

Any action or process of interest involving database objects is termed an *operation*, or procedure. It is an action that either queries or alters the database. Examples of operations might include "display a map," "update the crop yields," or "summarize a census district." As with objects, operations are classified into *types*, and specific occurrences of an operation are said to be *initiated* when they happen.

Two fundamental types of operations, *primitive* and *compound*, are distinguished according to their level of complexity and usage. A single operation that is not composed of any others and is not initiated by a user, is termed a primitive operation. Primitive operations are provided as a basic part of a DBMS and examples are given



below. An operation that is seen as a single and complete unit to a user, but is made up of a group of other operations, is a compound operation. For example, "list the owners of parcels facing high traffic roads," and "suggest the best location for a new supermarket," are complex operations.

As with object types there can be subtypes of operations. An operation will be a subtype of another if it operates on objects that are subtypes of the other's objects, and if the operation performed is a special case of the other. For example, a function to determine forest yields might have different forms depending on the predominant tree types.

### 3.4.2 Primitive Operations

The lower level, primitive, operations are the simple building blocks for more meaningful database activities. Four kinds are distinguished — calculate, retrieve, update (i.e., create/delete/change), and interact (i.e., input/output):

*Calculate.* Derive new data by processing some given data. Calculate, for example, the mean and variance of a set of values. Many different kinds of such processes, e.g., arithmetic, geometric, statistical, textual, etc., have been identified and used in programming languages although not all are provided as part of a database language. Traditional models of computation follow a set of instructions in a defined order, and are called *algorithmic*. Computational models that attempt to emulate human intelligence follow a set of rules in a non-predefined order, and are called *heuristic*.

*Retrieve.* Get a specified part of the database (i.e., some set of objects) and assign its value to a variable. "Select the buildings over 3 stories," is an example.

*Update.* Modify the content of a specified part of the database using new data. That is, create a new object, delete an old one, or change an existing one. "Insert

an instance of an infestation," "remove a person from residence," or "alter the value for the locust population," are examples of update operations.

*Interact.* Communicate with a user or device. Receive data from a user and assign it to a variable, or send a given variable's value to a particular user or device. "Get a value from the keyboard," and "Print the name of a (retrieved) region," are examples of such interactions.

Note that the calculate primitive applies to data that have already been input or retrieved and the output primitive applies to data that have already been retrieved. Also, both the retrieve and the update operations require first that an appropriate part of a database be specified, or selected. Such selection is described in §3.5.1. Although primitive operations are not directly initiated by users, most meaningful database work requires a combination of primitives.

### 3.4.3 Queries and Transactions

Higher level database operations that users initiate to either interrogate or alter a database are referred to as *compound operations*, or *events*. They include *queries* and *transactions*. An operation that peruses a database is termed a *query*. A query, such as "display all pine forests above 200m" or "print the names of all people in section S4," does not change the content of a database. An operation that corresponds to an actual event in the real world is termed a *transaction*. A transaction, such as an increase in road width, a change in land use, or a growth in population, alters the content of the database. Transactions that transform the database from one consistent state to another are called *consistency-preserving transactions* (CPTs) [Zhang 1989]. For example, a transaction that represents the logging of a forest stand must leave the land cover model with a logged rather than treed stand. Of course, all transactions should be consistency-preserving, but not all may be properly designed that way.

Both queries and transactions are built from other operations — queries from a

combination of retrievals, calculations, and interactions, and transactions from a combination of other transactions, primitives, or programming language constructs. Both are specified by users via a data manipulation language.

#### **3.4.4 Data Manipulation Languages**

Often, what are called *query languages* are actually data manipulation languages because they deal not just with queries but also with transactions. DMLs vary according to how users must formulate queries and transactions. Two broad approaches are the *procedural* and the *declarative*. Expressing database operations as a series of basic operations is said to be *procedural*, in which case a high-level operation must be formulated using many steps. The other approach focusses on the "what" more than the "how" by expressing the database operation in a more abstract form, leaving out the details. It is said to be *declarative*. Even with some declarative languages, where most operations can be specified with single statements, an intimate knowledge of the data structure is still required. In which case, a high-level operation must be formulated using a very complex statement.

### **3.5 Further Database Model Concepts**

More concepts of database models, such as the specification of parts of a database, restrictions on forms of data, dependencies between database objects, simplification of information, and alternative views of the same information, are now presented. Some may be found in general database literature [Bradley 1981; Date 1985; Korth & Silberschatz 1991; Martin 1976], others in the more advanced literature [Borgida 1986; Brodie *et al.* 1984; Brodie & Mylopoulos 1986; King & McLeod 1985; Su *et al.* 1988; Ullman 1988; Yao 1985].

### 3.5.1 Selection

Essential to both database statics and dynamics is the specification of a subset of a database — either the isolation of parts of a database that already exists or the definition of a database view that is to be derived from an existing one. The mechanism for specifying a subset of a database is known as a *selection*. A selection usually contains a *predicate* that restricts its content. A predicate is a set of conditions that must hold true for the specified subset of the database to exist. "Select all main roads in Gotham City" is a predicated retrieval. A selection is used both in the retrieval and update primitives and in schemas defining external databases. Suppose we have a conceptual database of topographic information. A selection may be "all roads," and it could be used simply for a one-time query from the database for road information, or it could be used to create an external database of road information for many subsequent uses.

### 3.5.2 Constraints

Conditions on objects and operations, expressible in a schema, and used to prevent violations of integrity, are termed *constraints*. Ideally, a database having integrity reflects only a possible, probable, and legal state of the world. Constraints restrict the possible instances of a database; they specify that only a subset of all possible database occurrences is legal.

Three types of constraints have been identified [Brodie 1984]. Conditions that are part of the data model are *inherent* constraints. For example, inherent in the hierarchical model is that a node has one parent and many children. Conditions that are specified directly by the data model are *explicit* constraints. For example, it may be explicitly stated that a road width may never be less than 3m or more than 30m. Those that are logical consequences of other constraints are *implicit* constraints. For example, if a road number is defined as a key attribute of a road entity, then the fact that no two roads can have the same number is implicit.

### 3.5.3 Dependency and Redundancy

Also related to the preservation of database integrity are the notions of dependency and redundancy. When the existence or value of one database element is determined by or dependent on the existence or value of one or more other elements, there is *dependency*. A railway station, for example, depends on the existence of a railway. Note that when "A depends on B," it is often written as " $A \leftarrow B$ ," or conversely as " $B \rightarrow A$ " for "B determines A."

When any fact is recorded more than once in a database, there is *redundancy*, and potentially the problem of wasted space and a loss of consistency. A multi-thematic database that stores topographic data explicitly with each thematic overlay is guilty of redundancy. (Thematic maps often use a topographic base map as a spatial reference.) Fortunately, the cause and cure of redundancy go hand-in-hand, i.e., dependency not only causes redundancy but should permit its proper management [Ullman 1988].

### 3.5.4 Attribute-Mapping Dependency

The kind of dependency most often talked about concerns the mapping between attributes, i.e., the number of values of an attribute that are associated with the given value of another attribute at any instant. This dependency between attributes really reflects the semantics of the entities being described. I call it *attribute-mapping* dependency.

When a given value of attribute X uniquely determines the value of attribute Y there is said to be a *functional* dependency of Y on X. It happens when there is a 1:1 or m:1 relation between attributes X and Y. If, for example, in a national topographic database the city attribute mappings C.name-to-C.size and C.name-to-C.province are 1:1 and m:1, respectively (meaning that each city name is associated with only one possible size and one province), then both attributes C.size and C.province are functionally dependent on

attribute C.name. I prefer the term *univalued* dependency because, first, it is more consistent with the following, and second, "functional" implies some kind of derivation.

When zero or more values of attribute Y are associated with a given value of attribute X there is said to be a *multivalued* dependency of Y on X. It happens when there is a 1:m or m:m relation between attributes X and Y, and is actually a generalization of a functional dependency. Continuing the above example, if the mappings C.name-to-C.college and C.name-to-C.airline are 1:m and m:m, respectively (meaning that each city name is associated with a set of colleges and airlines), then both attributes C.college and C.airline are multi-dependent on attribute C.name.

### 3.5.5 Computational Dependency

Some useful database information may be derived computationally from other database objects. The information can be in the form of attributes of or relations between entities. Such information, being a derivation or subset of the overall database, forms part of an external database view. Generally it could be considered to be either derived from the main database when needed or stored in the external database semi-permanently. In either case, the information is what I call *computationally* dependent on other data. Some examples of computational dependency are now given:

*Measurements* → *Coordinates*. In a database for a survey control network the locations of control points are usually held as coordinate attributes. The coordinates, however, actually depend on a set of distance and direction measurements that have been made between the control points.

*Details* → *Summary*. Statistical summary information is often more useful than the detailed, raw data. Mean and variance, for example, may summarize forest stand sizes, and depend on them for their values. In turn, the stand sizes may depend on stand boundary coordinates.

*Globe → Map.* A world map depends on the underlying model of the world. It is the result of a cartographic projection and symbolization process. Both the map and the globe may be held in a database, but one is derived from the other.

*Region Adjacencies.* Suppose that counties are considered to be composed of a set of districts that in turn are represented by vectorized polygons. The adjacency of counties depends on the adjacency of their districts, which depend on the polygon adjacencies. Similarly, whether two generic regions overlap is dependent on the coordinates of those regions.

There are two ways such a dependency may be properly managed. In one, the dependent object is never stored, but it is automatically computed from the independent objects whenever necessary. In the other, the dependent object is always stored, but it is recomputed automatically whenever the independent objects change. I call this *controlled dependency*.

### 3.5.6 Abstraction

The concise representation of something complex, where important aspects are shown and irrelevant details are hidden, is known as *abstraction*. Although it should already be apparent that abstraction exists throughout the realm of databases, a particularly important concept is the notion of the *abstract entity*, usually known as an abstract object.

An abstract entity has an internal key; it need not have an external key. It is thus useful for representing a particular forest region or geological outcrop, for example, that may have many properties of interest to a user, such as its classification, location, or area, but has no relevant unique identifier. Of course, it can have an external key if an application requires one.

From a user's point of view, relations exist directly between abstract entities themselves, not via their identifiers. To say that a river's tributary is another river, for example, is simpler and more meaningful than to say that a river's tributary is the identifier of another river.

### 3.5.7 Encapsulation

Although objects and operations have been described separately — objects being entities and their defining attributes and relations — it should be seen that there really are only a limited number of meaningful operations that can be performed on each object. For example, a land parcel will have various characteristics, such as ownership, taxation, usage, and size, as well as various appropriate operations. These might be change in ownership, subdivision into smaller lots, determination of adjacent parcels, or determination of overlapping soil regions. In a sense, some operations can be thought of as types of attributes, making up part of the definition of an entity. The ability to define operations applying only to particular types of objects and to require that all access to those objects is via initiation of the defined operations is called *encapsulation* [Ullman 1988].

### 3.5.8 Relativism

By now there may be some confusion as to the differences between some of the concepts that have been described. An object can be seen as basic or composite, or as an entity or an attribute. A relation can be seen as being one-to-many or many-to-one, or as an attribute of an object or an operation. The distinction between various pairs of database concepts really depends on the user's point-of-view. The ability to allow alternate conceptualizations of the same piece of information by different users is known as *relativism*. Take the case of a cadastral lot. An Owner may be seen as just an attribute of a Lot entity to one user but to another it is seen as an independent Person entity, joined



by a relation Owns. Also, Area may be an attribute of Lot or an operation to be performed on Lot.

### **3.6 Database Model Use and Implementation**

Although database models are conceptual tools there are a number of practical issues to consider, such as how they actually relate to a working database environment. They can vary according to the degree to which they are used or implemented, from remaining as a conceptual tool for database design to being fully automated as a DBMS.

Designing a database includes both logical and physical design phases [Hevner & Yao 1985; Yao *et al.* 85]. The logical database design phase works from a high level of abstraction to a relatively low level in three main steps. First, requirements specification, involves an analysis of information system requirements and a description of the necessary information. Second, schema design, means taking the requirements and producing external and conceptual database schemas using a data model, independent of any DBMS. Third, restructuring, involves converting the above schemas into ones compatible with the database model supported by the actual DBMS to be used. Physical database design encompasses such things as file storage allocation, index optimization, performance evaluation, and security.

Each high-level database model can be used as a tool for designing database schemas, leaving the implementation to a lower level form. Schema building is possible by means of the formal or informal, graphical or textual, database language that accompanies the model. In a simple form, the model remains as a conceptual tool only, requiring whatever schemas that are produced to be manually converted to lower level, computerized ones. The entity-relationship model is a popular example of such a model [Chen 1985]. In an advanced form, the model may be implemented as a CASE (computer-aided software engineering) tool for building other DBMS schemas.

Rather than serve as a tool for designing other schemas, a semantic database model could be implemented as a DBMS. Then the second and third logical database design stages above could be combined into one process. The ideal method of such an implementation would be starting from the ground up, producing a full system with many layers of software above a basic operating system. Another method would take the shape of a partial extension to an existing DBMS. The easiest way of implementing a new database model is to build a partial layer of software above another DBMS. Implementation of database models as software packages is a physical database issue, another extremely complex problem entirely, and beyond the scope of this thesis.

For what application purposes are database models designed? There are different degrees of applicability that database models can exhibit. As with programming languages, the higher the level of abstraction, the more specific is the problem that can be solved, and the easier is the solution of such problems. Take a high-level, geographic database model, the object of this thesis. From a computing point-of-view, the model must be more special-purpose than general-purpose, i.e., it must facilitate *geographic* database design and use. From a geographic point-of-view, it must be general and simple enough to be useful for *most* geographic purposes, i.e., it must not be so special or unduly complicated that it excludes certain applications. Finding a happy medium between generality and speciality is what I call the model designer's applicability problem.

### **3.7 Database Model Types**

Database models usually vary according to the degree to which they exhibit the different concepts described above. The earlier broad category of models are the syntactic, record-based models and the later broad category of models are the semantic, object-based models.

The earlier models include the hierarchical, network, and relational database models [Bradley 1981; Date 1985; Korth & Silberschatz 1991; Martin 1976]. They are well-known and well-defined, but support very few of the above concepts explicitly. The result is that many extra constraints, dependencies, relations, and operations have to be either explicitly specified with the database schema or programmed externally to the DBMS. In each case, more work must be performed by users for each application.

The newer models are less well-defined and less well-known, and examples include the object-oriented, semantic hierarchical, functional, binary relational, and entity-relationship database models [Brodie *et al.* 1984; Brodie & Myopoulos 1986; Khoshafian & Abnous 1990; Korth & Silberschatz 1991; Ullman 1988; King & McLeod 1985]. They attempt to represent an application environment's informational concepts more directly, resulting in more functional, flexible databases with higher integrity. As research and development progresses, the newer models are likely to become more popular and available as commercial DBMSs.

Resulting from the development and integration of artificial intelligence (AI) and database technologies are *knowledge-base systems* [Brodie & Myopoulos 1986; Ullman 1988], database systems with more human-like intelligence. The challenge is to combine the techniques of efficient storage and retrieval of large volumes of relatively simple data with those of making logical inferences from relatively few but complex relations.

At a practical level, the concern is with database models for database design and use. The databases are (central) components of information systems, and along with programming languages, are meant to model reality. But programming languages too, whether conventional or AI-oriented, require conceptual frameworks to assist them in their modelling. Since the distinction between database and programming languages is rather technological, and both have similar conceptual requirements, there is room for more general conceptual tools, or conceptual models, applicable to both database and

programming languages, for modelling reality [Borgida 1986; Brodie *et al.* 1984]. In the future, perhaps *conceptual modelling* will be realized and involve more direct simulation of phenomena, things, activities, and events of the world, independently of database or application programs.

Traditionally, database models have been applied toward business and commercial information systems, and have used relatively simple objects. The more complicated scientific and engineering applications, such as computer integrated manufacturing (CIM), geographic information processing, and "hyper media" publishing, require database models with much more complex modelling capabilities. Time, space, and very complex objects and events are being incorporated into the newer special-purpose database models [Dayal & Smith 1986; Orenstein & Manola 1988; Su *et al.* 1988].

### **3.8 Conclusion**

In this chapter, I have synthesized some important general concepts of computerized information management, making them relevant to geographic data processing. Such concepts can be summarized as follows.

A database is an organized repository for information on some part of the world of interest. Logical database modelling is the formal conceptualization of those aspects of databases that represent structure and behaviour of the phenomena being represented. Phenomena are considered to be represented by objects and operations. Objects comprise entities being characterized by attributes and relations. Operations are performed on objects and comprise queries and transactions. Different database models vary according to their levels of abstraction, the newer ones providing high-level constructs that can easily be used to model reality. Database modelling concepts will continue to develop, resulting in even more sophisticated, useful, and easily built information systems.

As well as produce the above synthesis, I have conceived certain database principles of my own: internal and external keys (§3.3.3); attribute-mapping, computational, and controlled dependencies (§3.5.4 and 5); and the model designer's applicability problem (§3.6).

Since the goal of the thesis is to propose a logical database model suitable for GIS, the concepts in this chapter (concerning information or data in general) need to be tempered to suit the specific requirements of geographic data management. I intend to incorporate most of these concepts, together with geo-data concepts, within my model. The informational requirements of geography are investigated in the sequel.

## Chapter Four

# Geographic Data Concepts

### 4.1 Introduction

Consider geographic phenomena as things or events both existing in the world and being considered of importance or interest. They are very complex but can be described or represented in some manner as geographic information or data. The management of such data can be a formidable task, but should not be.

The thesis objective is to propose a high-level geographic database model, a theoretical framework for data storage and retrieval within a geographic information system (GIS). While the previous chapter covered general database concepts, this chapter is intended as a review and synthesis of important concepts from surveying, geographic data processing, and cartography, that are deemed particularly relevant to geographic databases.

The complexity of geographic information can be managed by grouping related aspects into fundamental classes called *domains* of information. I will show that geographic information can be logically viewed according to two families of such domains — *characterization* domains and *abstraction* domains. That is, geographic information has topical, spatial, and temporal components, and varies in its level of generalization, realization, and construction. These two families of geographic data domains will be described.

## 4.2 The Geographic Characterization Domains

A geographic phenomenon has certain characteristics that distinguish it from others. The characteristics, describing the *what*, *where*, and *when* of the phenomenon, can be recorded as data in the form of attributes and relations. A family of related characteristics are considered here to constitute a *characterization domain*. Three such domains describe different fundamental aspects of any geographic object or event. There are *topical* characteristics, describing what the thing is, *spatial* characteristics, describing its location and extent, and *temporal* characteristics, describing dates and durations [Berry 1964; Dangermond 1982]. The domains are considered orthogonal to one another because each is largely independent of the others. The structure of and operations within each of the three characterization domains, as well as their inter-relations, are described in this section.

### 4.2.1 Elemental Data Types

Before discussing the three specific domains of geographic information, it is worth considering the general elemental types of geographic and other scientific data. These data types, or variable types, are often referred to as measurement scales and are described in many technical geography texts [Dent 1990; Ebdon 1985; Hammond & McCullagh 1978]. Broadly speaking, there are *qualitative*, or symbolic, data and *quantitative*, or numeric, data. Qualitative data usually involve discrete classes or categories. They are further divided into *binary*, or boolean, values, i.e., yes/no; *nominal* values, such as "building," "telephone line," or "lake;" and *ordinal* values, such as "road" classified into "dual highway," "hard road," and "gravel road." Quantitative data usually involve continuous statistical values. They are further divided into *interval* values, such as temperature in °C, and *ratio* values, such as deaths per thousand people. It can be seen that the data types take their names from the kinds of meaningful operations that may be performed on them. The other, special types of geographic data — the topical, spatial, and temporal types — can be built out of the elemental types and are described below.

Elemental operations, or those that can be performed on elemental data, include input, calculate, retrieve, update, and output, and are discussed in more detail in §3.4.2. A calculation is an operation that takes certain data (called arguments, or operands), does some computations, and gives resulting data. It is used in combination with retrievals and updates to effect queries and transactions. A retrieval gets data from the database given some criteria. An update either stores new data, deletes old data, or changes existing data, in the database.

#### 4.2.2 The Topical Domain

The first geographic characterization is termed here the *topical* domain. Other candidate names might be the *taxonomic*, *thematic*, statistical/categorical, or simply *attribute* domains. Information on a geographic phenomenon in such a domain answers the questions "what" or "how much" is the thing. The domain is multi-dimensional. This is because geographic phenomena are usually categorized into themes, or subjects, such as topography, terrain, bedrock, vegetation, land use, or census, with some common characteristic distinguishing the themes, and because there are often many ways to classify or measure something within a given theme.

Depending on the application, different classes of geographic variables may be worth being considered as special cases. In remote sensing [Lillesand & Kiefer 1987], for example, data exist within a 2D spatial domain, a domain having spectral and radiometric dimensions, a temporal domain, and also perhaps a land cover (or some other) classification scheme. In this thesis, only the spatial and temporal domains are considered to be important enough for special consideration in a general geographic sense (see §4.2.3 and §4.2.4). Thus, the spectral-radiometric domain and the various classification groups would be considered parts of the multi-dimensional topical domain.

**Structure of the Topical Domain.** The topical domain usually has a *hierarchical* structure (Fig. 4.1). Themes are subdivided according to a classification scheme, or taxonomy,



into classes, such as roads, rivers, commercial land, and residential land. Classes may further be classified or stratified based on qualitative or quantitative characteristics. Examples of subjects that may use hierarchical taxonomies include topography, geology, biology, and demography.

Usually, a particular application has a fixed number of predetermined, named levels in the classification scheme. A four level hierarchy for classifying digital topographic data [Canadian Council on Surveying & Mapping 1984], for example, has Classes, Categories, Features, and Attributes. However, I argue that, generally, the structure of most classification schemes is *recursive*; at all levels, one class is subdivided either into several other classes or into a group of elemental types. Class division iterates in a similar fashion until enough detail for the level of study is reached.

Conventional taxonomies are strictly hierarchical, i.e., they conform to a tree structure (Fig. 4.2), where each node (class) in the tree has only one parent (superclass). However, for those complex geographical applications where many different themes are being considered at once, such hierarchical classification schemes may be inadequate. They can be too simple or restrictive when something is considered to immediately belong to more than one class or theme. A river, for example, may simultaneously belong to the transport route, national boundary, drainage channel, and waterbody classes [Egonhofer & Frank 1989]. The taxonomic result is an interlocking of hierarchies known as a network, or *lattice*. A lattice allows a node to have more than one parent (Figs. 4.3 and 4.4).

**Operations within the Topical Domain.** Given a particular classification scheme it is possible to retrieve certain information: the identity, description, or value of a particular item; the class or classes to which a particular item belongs; the characteristics of a particular class or category; the possible kinds of things belonging to a particular class; the general classes within which a particular class belongs; etc.

There are many simple textual, mathematical, and statistical computations that can be performed on topical data. Quantitative operations include sorting, calculating sums,

differences, maxima, minima, products, totals, percentages, means, variances, etc., and assigning categorical classes to numerical ranges. Although they must be considered in any model or system, they are well-documented in various computing or technical geography texts [Dent 1990; Ebdon 1985; Hammond & McCullagh 1978] and so can be excluded from this discussion.

### 4.2.3 The Spatial Domain

All geographic phenomena can be positioned or located in space, and have their spatial form and extent described. This primary domain of information may be included within a general information domain, but since space is of paramount importance in geography the *spatial* domain is identified here. Clearly it could also be called the *geometric* domain. Information on a geographic phenomenon in such a domain answers the questions "where" or "what shape" is the thing. Although the spatial domain is three-dimensional, it is often more convenient for geographers to separate it into sub-domains. Thus, phenomena are positioned within a two-dimensional horizontal domain and within a one-dimensional vertical domain (orthogonal to the horizontal).

### 4.2.4 Structure of the Spatial Domain

There are a number of important concepts and techniques that address how the spatial domain is structured and how spatial data are recorded.

**Elements of Space.** The types of elements into which horizontal geographic space can be partitioned are categorized according to their individual dimensionality into *regions*, *lines*, and *points* [Peucker & Chrisman 1975; White 1984]. Regions are 2D elements, or 2-cells; lines are 1D elements, or 1-cells; and points are 0D elements, or 0-cells. In the case of 3D space, *solids*, or 3-cells, can be added to the list of elements. Relations between any two spatial elements may be that they bound, contain, or overlap one another.

**Relations in Space.** Spatial elements can be related to one another in many different

ways. The following taxonomy of spatial relations is based on others [Burton 1979; Claire & Guptill 1982; Cox *et al.* 1980; Pullar & Egonhofer 1988]. Five main kinds are identified: joint, disjoint, intersect, neighbor, and collateral. The first three are mutually exclusive and exhaustive. Only topological relations are considered here; metrical ones are considered as operations (see §4.2.5). Except where noted, relations may be between elements of the same or different dimensional types.

*Joint, or Contiguous.* Two elements touch, or meet, one another. Two types of joints can be distinguished:

*Incident, or Boundary.* The elements meet directly and are of different dimensional types (Fig. 4.5). It may be known as boundary because those elements incident on an element define its boundary. Examples of incident pairs at a purely geometric level are line-region, line-point, and point-region; while more applied examples might be pipe-valve, road-intersection, lake-shore, and basin-divide.

*Adjacent.* The elements are not incident, but meet "indirectly" via an element of lower dimensionality (Fig. 4.6). Geometric examples include line-line, via a common incident point, and region-region, via a common incident line; while lake-forest, via shore, and lot-street, via property line, are applied examples.

Notice that adjacency can easily be derived from incidence. And that the distinction between the two depends on the scale of investigation. At a larger scale, the road-intersection example would be an adjacent relation if both entities were areal.

*Intersect.* Two elements have some space in common. Three types of intersects can be distinguished:

*Overlap.* Some space is common to part of each element (Fig. 4.7). Line-region, region-region, and line-line may be overlapping pairs of geometric elements. Applied examples may include overlapping forest land and private land, province and river, or river and road.

*Enclose, Cover, or Include.* One element's space completely encompasses another's (Fig. 4.8). For example, region-points, region-regions, and line-points are basic enclosures, and country-cities, country-parks, and canal-locks are

applied ones. Note that the relation is asymmetrical; depending on the way it is viewed it is either "contains" or "is inside."

*Equal, or Coincident.* The two elements share exactly the same space, such as when a basin and a park are the same regions, or when a road and a boundary are the same lines.

*Disjoint, or Discontiguous.* Two elements do not touch or have any space in common.

*Neighbor.* Two elements are "topologically close." Two types of neighbors can be distinguished:

*Adjacent, or Primary neighbor.* The elements are joint, as defined above.

*Proximal, or Secondary neighbor.* The elements are disjoint but are connected via a common joint element. For example, Alberta and Manitoba are proximal provinces, via Saskatchewan.

Note that the proximal relation can easily be derived from the adjacent one. Neighborhood can also be defined in a metrical manner (see §4.2.5).

*Collateral.* Two elements are within a common enclosure, i.e., they are within the same larger element. For example, collateral points are within the same region, or cities within a province are collateral. The collateral relation can easily be derived from the enclose relation.

**Partitioning of Space.** Geographic space is usually partitioned into meaningful regions and boundaries or lineaments according to natural characteristics or to cultural designations. Other regions, e.g., map sheets, may be artificial in the sense that they are created merely for the convenience of data management.

As in the topical domain, a *hierarchical* system is usually in place [Beatty 1980] whereby each partition, or region, is successively subdivided into smaller ones until the resulting regions are small enough for the level of investigation (Figs. 4.9 and 4.10). For any given spatial partitioning scheme, there is usually a fixed number of named levels in the system of spatial subdivision, e.g., meridian, range, township, section, etc., or

country, province, region, and district, and the smallest subdivision defines the resolution of the addressing system. However, I claim that, generally, spatial subdivision is a *recursive* mechanism; at any level in the structure a region is part of a larger region and contains other smaller regions. A quadtree is an example of a recursive spatial partitioning system [Samet 1984].

An important property of many spatial partitioning systems is that every point in the study region is accounted for by one and only one spatial element (except those points on a boundary between regions). Such a property is known as *exhaustiveness* and *exclusiveness* because space is exhaustively partitioned into mutually exclusive regions. An exhaustive partitioning system is also said to be continuous because there are no gaps in space (Fig. 4.11). In graph theoretic terms, such a partitioning is known as a *planar map*.

Some partitioning schemes are not exhaustive, i.e., they have some space unaccounted for. Perhaps the most common scheme of all, postal addresses of street names and house numbers, is discontinuous within regions; it partitions towns into places along a linear network. Other schemes are not exclusive. In a geographical analysis where several themes are being considered, regions often overlap with one another, or a place may be contained in more than one region. For example, a land parcel can simultaneously be in a cadastral lot, an electoral ward, a census district, a soil zone, a geological zone, etc. Such a non-hierarchical subdivision of space occurs when there is a network of interlocking hierarchies (Fig. 4.12) and is known as a *lattice* [Kainz 1988]. Some partitioning schemes are neither exhaustive nor exclusive, i.e., they allow overlaps and gaps (Fig. 4.13). A wildlife habitat study, for instance, will have some places where no species of interest live, and will often have several species' habitats overlapping.

Many geographical studies have data that are collected and stored in arbitrary regions, or *tiles* [Cook 1978], divided by artificial boundaries, or *seams*. When a geographic database hides the details of the data collection or storage boundaries from the users, the database is said to be *seamless* [Aronson 1989]. Although there may be map sheets or other boundaries existing for convenience of the technical storage method, such

boundaries should not be allowed to interfere with a user's concept of regions relevant to an application.

**Geo-Referencing.** Describing the position of geographic objects is referred to as *geo-referencing*, or *geo-coding*. There are two fundamental ways of doing so. One is symbolic in that it involves naming the spatial partitions (i.e., points, lines, or regions) housing the objects or identifying an object's relations to other objects. A postal address is an example of a symbolic geo-code. The other kind of geo-referencing is numeric in that it involves specifying an object's geometrical attributes. Map coordinates represent a common form of numeric geo-coding. Mostly, geo-referencing methods are used in combination.

Positioning objects in terms of other objects is termed here *relational* geo-referencing and can be done topologically or hierarchically. The *topological* method positions objects of a planar map in terms of other objects in the same planar map (Fig. 4.14). The other objects could be called *control objects* and are somehow positioned themselves, independently. A common structure is to position regions by reference to their bounding lines and to position the lines by reference to their bounding points [Gold 1989; Peucker & Chrisman 1975]. The points themselves are positioned using coordinates (see below). The *hierarchical* method simply positions an object on one level of a hierarchy in terms of control objects that are on a lower level in the hierarchy [Pullar 1989]. For example, a country may be positioned by referring to its provinces rather than by repeating the coordinates of the boundaries common to country and province (Fig. 4.15). The relational methods themselves require no storage of coordinates, but do require relations. When the positions of the independent, control objects change, so do the positions of the dependent objects automatically change. Other advantages are that many spatial searches become quicker because less coordinate computations are necessary and that checking the consistency of a spatial partitioning becomes easy [White 1984].

Positioning an object using some metrics is termed here *metrical* geo-referencing. It is done either absolutely and independently using coordinates, or relatively using

measurements from control objects. The *coordinate* based approach is very common. It requires no links but does require storage of coordinates (Fig. 4.16). It means that objects can be independently processed, i.e., they stand alone. It also means, however, that there is much redundancy, especially in the case of common boundaries mentioned above, and thus there is the potential for inconsistency, if one boundary were to move. Other disadvantages are that spatial searching becomes quite slow because it is all based on coordinate computations and that consistency checking is virtually impossible. The *measurement* based approach is very rare. It is an application of the idea from surveying that coordinates all depend on a network of measurements and control points [Buyong & Frank 1989]. Thus, control objects are coordinated and dependent objects are positioned relative to them using the measurement network (Fig. 4.17). It has the same advantage as the topological method described above, as well as the advantage of potentially having greater accuracy due to the use of raw survey data. The disadvantage might be the extra computation required to derive the dependent coordinates.

It can be seen that both relational and measurement based positioning are examples of the principle of controlled dependency (see §3.5.5). Redundancy is reduced or controlled and consistency is maintained. The positional values of dependent objects either are not stored but are derived each time they are required, or are stored but are recomputed when appropriate.

**Coordinate Systems.** Place names have some meaning to humans and we usually know where things are well enough for most practical purposes. However, the exact spatial position and extent of a place needs to be determined for unambiguous studies and for machine recording and processing.

Any point can be uniquely and unambiguously positioned in space using a pair (or triplet) of *coordinates* drawn from a *coordinate system*. Such systems are usually either plane or geodetic. *Plane* coordinate systems locate objects with respect to a flat surface (Fig. 4.18). They are usually rectangular and hence conceptually and computationally very simple. (A less common alternative to rectangular coordinates are polar coordinates.) Their great disadvantage is that the world is not flat and plane coordinates

are progressively less appropriate as the region of study gets larger or accuracy requirements increase. *Geodetic* coordinate systems locate objects with respect to a spherical or ellipsoidal surface (Fig. 4.19). Such a system is globally continuous because a single coordinate system is used consistently throughout the globe. It is far more appropriate for geographic objects because an ellipsoid is such a close approximation to the shape of the earth. The disadvantage of geodetic coordinates is that computations on the ellipsoid are very complicated.

To map objects that have been positioned in geodetic space into plane space, a *map projection* is required. There are dozens of projections, each preserving certain geometric qualities of mapped objects while distorting others. To map large areas using plane coordinates, either one projection is used and large errors are introduced, or the region is split into zones. Each zone has its own projection and coordinates that are incompatible with adjacent zones. The details of coordinate systems and map projections can be found in many surveying and cartography texts [Jackson 1980; Maling 1992; Richardus & Adler 1972; Robinson *et al.* 1984; Snyder 1987].

**Encoding of Space.** For the machine recording of an object's spatial extent two alternate fundamental methods are used, *vector* and *raster*. The first is an outlining technique where objects are delineated using ordered sets of coordinates. It is known as the vector form because the coordinates are two or three element vectors. It is analogous to a picture being drawn using a series of pen strokes, and is sometimes called calligraphic or boundary representation. With the second technique space is considered to consist of a fine array of small, usually square, picture cells, or *pixels*, and an object is represented by the set of cells by which it is covered. It is commonly known as the raster form, but could be called the cellular, tessellation, or space-filling form. To continue the above analogy, the raster approach is analogous to a painting being covered in paint. In the 3D case, the raster method divides space into cubic, volume cells, or *voxels*.

The fundamental elements in the vector form of encoding directly reflect the dimensions of the spatial elements to which they correspond. Thus, they include the *0-cell* (point, node, or vertex), the *1-cell* (line, arc, or edge), the *2-cell* (area, polygon, or face), and the



*3-cell* (volume, polyhedron, or solid). With each there may be special types, such as straight lines, circular lines, squares, and triangles. There are numerous schemes for combining them into compound structures and relating them to one another in order to achieve different results in the trade-off between storage and processing. Such is a very important aspect of any physical data model, and many other studies have been done [Gold 1988; Peucker & Chrisman 1975; Shamos & Bently 1978]. They will not be elaborated upon here, suffice to say that a comprehensive geographic data model would allow a variety of alternatives to be used under different circumstances.

While the elements of the raster form of encoding are essentially all arrays of pixels or voxels, the method can be further classified in three important ways. The first is according to cell dimensionality — 2D or 3D, pixels or voxels. The second is based on the cell data types. Conceptually, these can be *qualitative* or *quantitative*. Thus, a "picture" contains qualitative (e.g., soil type) or quantitative (e.g., soil moisture) data. Practically, however, the cell data types reflect certain machine data types, i.e., *bits*, *bytes*, *words*, etc., where a "picture" contains 1-bit, 8-bit, or 24-bit data. The third raster classification method is based on the cell divisibility. There are *non-hierarchical* structures, such as the bitmap, and *hierarchical* structures, such as the quadtree. The simpler structure has a single cell-size; no matter how coarse or detailed the picture is it is uniformly divided into pixels. The hierarchical structure uses multiple cell-sizes; cells divide into smaller cells according to the detail of the picture. As with vectors, many different schemes have been proposed for improving raster storage or processing efficiency [Mark & Lauzon 1984; Peuquet 1978; Samet 1984] and they are not discussed further here.

Both vector and raster forms have many variants and both have certain advantages and disadvantages. Again, a comprehensive geographic data model would allow a variety of alternatives to be used under different circumstances.

#### 4.2.5 Operations within the Spatial Domain

Just as ordinary numeric and symbolic data can undergo basic kinds of operations, such as addition and multiplication, there are some basic operations that spatial data may undergo. Data that can be retrieved from a spatial data structure include, for example, the regions on each side of a line, the spatial elements overlapping another element, or the spatial elements inside a region. Examples of data that can be computed from a spatial data structure are the length of line, the angle of intersection between two lines, or the spatial expansion of an element. Changes can be made to a spatial data structure, such as divide a line into two by adding a point along it or join two adjacent regions by removing the dividing line. Following is a taxonomy of operations that can be performed on data within the spatial domain. It is based partly on the work of others [Burton 1979; Claire & Guptill 1982; Cox *et al.* 1980; Samet 1984; Tamminen 1984] and is organized into calculations, retrievals, and updates.

**Spatial Calculations.** Metrical processing of spatial data is termed here *spatial calculation* and almost always involves some measure of *distance*. In the simplest case the distance is euclidean, i.e., simply a straight measure between two positions. However, different constraints may be introduced. The measure may be constrained to be along a particular surface. In the simpler cases the surface may be horizontal, when plane or geodetic distance is measured, or it may be vertical, where height difference is measured. In the more complex cases the surface is 3D and the distance is measured along a continuously varying slope. The measure may also be constrained by particular features, such as being only along lines of a network, or such as being barred from certain regions.

Some spatial calculations take *two* entities as arguments and yield a *boolean* result, i.e., based on some spatial *comparison* between the two a yes/no result is produced. Essentially for each type of spatial relation (see §4.2.4 above) there is a possible comparison operation. The result of the comparison is the answer to the question "Are

the two entities related this way?" So, coincidence, contains/inside, overlaps, and joins are all examples.

Some spatial calculations take *one* operand and produce a *metrical* result. For example, the operation *size* produces the length, area, or volume of an entity.

Other spatial calculations take *two* operands and yield a *metrical* result. They include *Separation*, or *Shortest path*, the shortest distance separating two entities; *Angle*, the angle of intersection between two lines; and *Direction*, the direction of an entity from another, though this is rather ambiguous for anything other than points [Peuquet 1988].

Some spatial calculations take *one* operand and produce a *geometrical* result, i.e., they yield another spatial element as a result. Examples would include

*Envelope*, the smallest rectangle or box, whose sides are parallel to the coordinate axes, enclosing an entity;

*Centroid*, the geometric centre of an entity;

*Skeleton*, the centre line of a region;

*Expand*, or *Buffer*, the region bounded by a line a given distance away from an entity;

*Smooth*, the smoothed version of a jagged line; and

*Simplify*, the generalized, or less complex, version of an entity.

Finally, there are those calculations taking *two* operands and yielding a *geometrical* result. They include

*Intersection*, the space shared by two entities;

*Union*, *Combination*, or *Addition*, the combined space of two entities; and

*Difference*, or *Subtraction*, the space of one entity minus another.

The above lists are not necessarily exhaustive; there are also a host of other coordinate geometry (COGO) and computational geometry functions that are used in engineering surveying, computer graphics, and computer-aided drafting, design, and manufacturing

(CAD/CAM) [Bowyer & Woodwark 1983; Shamos 1978] — some of which may be useful here. However, the thesis focus is storage and retrieval, not processing.

**Spatial Retrievals.** Getting data from an existing spatial partitioning is termed here *spatial retrieval*. In the following, assume that a spatial partitioning exists in the form of geographic database entities, described by attributes and by relations to other entities. Also, that a *probe* is a variable that may assume the value of any type of spatial variable.

One class of spatial retrievals includes those that get *attributes* or *relations* of a given entity without requiring any metrical computation. Some are *metrical* in the sense that they get positional attributes of an entity. The location or height of a point and the delineation of a line or region are examples of metrical attribute retrievals. Many simple entity-based retrievals are *topological* because they get connected entities. Examples of topological relation retrievals include the points at each end of a line, the regions on each side of a line, the entities overlapping an entity, the region immediately containing an entity, and the topological neighbors of an entity.

Another class of spatial retrievals get *entities* that are *metrically related* either to given probes or to given entities. They are only made possible by doing some metrical computations, as well as some simple retrievals. *Position-based*, or probe-based, entity-yielding retrievals include the "range" query that gets the entities intersecting a rectangle probe; the "point-in-polygon-network" query that gets the region containing a point probe; and the "nearest neighbor" query that gets the nearest entity to a probe. The metrical neighborhood query is an example of an *entity-based* entity-yielding retrieval. It gets the entities within a given distance of an entity. It is derived by getting the entities intersecting an expanded entity.

**Spatial Updates.** An operation that adds, removes, moves, merges, or splits elements in a spatial partitioning is termed here a *spatial update*. The simplest updates are the creation or deletion of individual points, lines, or regions. Moving such features is achieved by combined creation and deletion. A line may be split by adding a dividing point, while two joint lines may be merged by removing the dividing point. Similarly, regions may

be divided or merged by adding or deleting lines, and solids may be divided or merged by adding or deleting regions.

#### 4.2.6 The Temporal Domain

All geographic phenomena can be located somewhere in time, or have their duration described. The third primary domain of geographic information then is the *temporal*, *historical*, or *chronological*, one. Information on a geographic phenomenon in such a domain answers the question "when is (or was)" the thing. The temporal domain is one-dimensional, and by itself is not complicated. It is when temporal data must be combined with other data that complexities arise. Since space is at least two-dimensional, and spatial data concepts have been thoroughly investigated, most concepts of time can be derived from those of space, especially if attention is paid to the sequence, or order, present in time.

**Structure.** Time can be partitioned using two basic types of *temporal elements* — *instants* and *periods*, representing points and intervals in time, respectively. Instants, or epochs, are 0D temporal elements while periods, or durations, are 1D temporal elements. The latter are bounded, or punctuated, by the former. Day and night, for example, are punctuated by dawn and dusk.

A temporal measuring scheme may be numeric or symbolic, and is always ordered. It is usually hierarchically structured to facilitate different scales, e.g., decade-year-month-day, week-day, and hour-minute-second, although different hierarchies may overlap. Sometimes, temporal elements are cyclic, e.g., the pattern of seasons repeats throughout a year, and hours throughout a day.

A relation between two temporal elements is termed here a *temporal relation*. The following taxonomy of temporal relations is based on a rigorous treatment of interval relations in 1D space, appearing elsewhere [Pullar & Egonhofer 1988]. Three major groups of relations — contiguous, discontiguous, and intersect — are mutually exclusive and exhaustive:

*Contiguous, or Joint.* The two elements "touch" in time, i.e., one begins at the moment the other ends. If a forest is harvested, its treed and logged states are contiguous in time. As with spatial contiguity, considerations of scale or dimensionality lead to two types of temporal contiguity, *incident* and *adjacent*, or *consecutive* (Fig. 4.20). The treed and logged versions of the forest are consecutive, but are incident on the harvest event, which ends the treed period and begins the logged period.

*Discontiguous, or Disjoint.* The two elements are completely separate in time.

Note that both the contiguous and discontiguous relations are asymmetrical; in each case one element is before or after the other.

*Intersect.* The two elements have some time in common. As with spatial intersect, there are three types of temporal intersects (Fig. 4.21):

*During.* One is completely within the time frame of the other, i.e., it occurs after the other begins and before the other ends. An accident occurs during the life of a power station, for example.

*Overlap.* One begins during the other but ends after it. For example, the times when a field was arable and when it was owned by Mr. Developer overlapped. Again the relation is asymmetric; one element is earlier, the other later.

*Coincident, or Equal.* The two elements are simultaneous.

**Operations.** Operations that can be performed on temporal data are termed here *temporal operations*. Alone, they are quite simple. Examples include determining how many days there are between two dates, the relation between the Pleistocene and Quaternary geologic temporal elements, and the sequence of particular dated events. The following taxonomy of temporal operations is based on the spatial one:

*Temporal Calculations.* Two kinds of calculations that can be performed on temporal data are considered here. First, are those that operate on single temporal elements. These simply include determining such things as the length, centre, or

extension of an interval. Second, are those that operate on pairs of elements. Boolean-yielding *comparisons* (or relational operators) of two elements include the comparison versions of the contiguous, disjoint, and intersect temporal relations, described above. The metric-yielding *separation* determines the period separating two elements. Finally, there are the *set operators*, yielding the temporal elements that are the intersection, union, and difference of two temporal elements.

*Temporal Retrievals.* An operation that gets data from an existing temporal partitioning is termed here a *temporal retrieval*. Two kinds of temporal retrievals are identified, *entity-based* and *probe-based*. For the first kind, assume that an entity is an existing element in a temporal partitioning. Included are both the retrievals that find out temporal attributes of an entity, i.e., when or for how long an entity existed, and those that get the entities either *topologically* related or *metrically* related to a given entity. In other words, related entities are those that exist before, during, or after an entity, or are consecutive with, overlap, or coincide with a given entity. For the second kind of temporal retrieval, assume a probe is any arbitrary temporal element. Again, entities metrically related to the probe can be retrieved.

*Temporal Updates.* The date of a geographic feature may be created, deleted, or changed. Also, there are temporal equivalents of joining and dividing features.

#### 4.2.7 Combinations of Domains

Meaningful geographical analysis and synthesis can only usually be done when combinations of data values, drawn from the three characterization domains, are considered at once. That is, when content, space, and time are combined.

A complete characterization of a geographic phenomenon can be seen as a set of points in what might be called "geo-space." That is, there is a combination of topical, spatial, and temporal data, describing what, where, and when is the phenomenon. Thus, a

complete "location" in geo-space consists of a content, a place, and a time. The *content* is a set of topical values, one for each topic being considered, and place and time are spatial and temporal values, respectively. The set of topical values for a series of themes at a given place and time could be seen as a thematic profile, or "core." For example, in a natural resources GIS the thematic profile for a particular location and date might include climate, rock type, soil type, vegetation cover, and terrain data.

It is usually instructive to study geographic phenomena, in a more holistic manner, as a combination of individual "geo-measures." Considering variation of geo-measures within all three characterization domains at once would be rather complex, so usually the relations between only two are investigated at one time, the other being held constant. The relations between domains can be considered as geographical, non-parametric *functions*, or *fields*, the variable in one domain depending on, or being related to, that in the other. So, there are topical-spatial functions, topical-temporal functions, and spatial-temporal functions.

#### 4.2.8 Topical-Spatial Functions

Perhaps the most common forms of geographical analysis consider the relations between theme and space at any given time. Based on the dimensionality of the independent variable, several such functions can be described — layers, profiles, and volumes.

**Layers.** Most conventional maps show spatial and topical variations of geographic phenomena with time standing still. Using a map we can determine the locational patterns of a certain phenomenon (mathematically, place is a function of content), or we can determine the state of various phenomena present at certain places (content is a function of place). The latter 2D function, relating topic to space, is called a thematic *layer*, or *coverage* [Chrisman 1981; Tomlin 1983] because it spatially covers a study area and produces a data value, at any given location, drawn from a particular topical domain. Since there are a huge number of possible topical domains that may be of interest at any time, there are that many layers. The number of different types of layers varies according to the different types of thematic variables that can be measured.



Depending on the theme represented or the phenomena modelled, there are discrete and continuous layers, and there are single-valued and multi-valued layers.

The discrete layer, or *discrete surface*, has a topical value that, first, may be qualitative or quantitative, and second, is constant within particular localities but changes abruptly across boundaries (Fig. 4.22). Such a layer may be a horizontal grouping of all features that belong to one theme, such as geology, and that cover a study area. So, a discrete surface can also be called a network of features and can be further classified into

- point networks, e.g., a network of horizontal survey control points;
- line networks, e.g., a road network;
- region networks, e.g., a collection of water, wooded, or built regions; and
- multi-typed networks, having a combination of feature types, e.g., a cadastral map.

For the sake of completeness, the space between particular features in any network can be considered to be valueless, or belong to a null feature. A region network is further classified according to the topical value type. Quantized regions, e.g., those given densities such as people per square kilometer, make up a *choropleth* coverage; and qualified regions, i.e., those grouped into discrete classes, make up a *categorical* coverage.

The continuous layer, or *continuous surface*, has a quantitative topical value that varies throughout space (Fig. 4.23). It can be considered as a semi-voluminous entity — the surface covering the volume. Three types of continuous surfaces can be distinguished, *smooth*, *rugged*, and *stepped*, depending on the continuity of the first derivative. For example, air pressure is a smooth surface, terrain elevation is a rugged surface, and population density is a stepped surface. Note that the choropleth discrete surface and the stepped continuous surface are essentially the same.

Even though two main subtypes of layers, discrete and continuous, are categorized according to the spatial continuity of the topical variable, some geographic phenomena are both discrete and continuous. Topography, for example, is often considered to be a continuous elevation surface and a discrete network of surface features.

Layers, or surfaces, can also be categorized as being *single-valued* or *multi-valued* functions, depending on whether the content at any place is defined by one or more values. Terrain elevation is given by a single value, but terrain slope is given by a pair of values. Wildlife presence, on the other hand, will have many values since any number of animal species may be present.

**Profiles.** Just as the layer can be considered as a 2D geographic function, with content depending on location and with theme and time being fixed, we can consider the map *profile* as a 1D geographic function, with content depending on distance along a spatial dimension, and time being fixed. The spatial dimension may be height, depth, or distance along any particular horizontal direction. Examples of such spatial profiles include a terrain elevation profile, cutting across a terrain elevation layer; a geological core; and columns of atmospheric or oceanic measures. The last three can be seen as profiles through a "true 3D" map. Like the layer, the profile may be discrete or continuous. For example, a geological core is discrete, and the change in temperature through an air column is continuous.

**Volumes.** As has just been indicated, some phenomena exist throughout the spatial domain, not just two of its dimensions. A *volume* can be considered as a 3D function, where content depends on position, and time is constant. In geology, meteorology, or oceanography, for example, a rock type, air pressure, or salinity can be measured anywhere within the land, air, or water, respectively.

#### 4.2.9 Topical-Temporal and Spatial-Temporal Functions

Different views of geographical phenomena can be obtained when time is brought into the analysis. Time-tables, time-series, and travel logs are all functions involving the temporal dimension.

If place is held constant, there is the relation between the topical and temporal domains to be considered [Langran & Chrisman 1988]. At any given place, we can measure the

value of some entity at given moments in time, e.g., temperature and pressure at a weather station, or conversely we can record the times certain phenomena occur, e.g., arrivals and departures at an airport. In the first case, mathematically, content is a function of time, and secondly, time is a function of topic. The former is essentially a *history* function and will vary according to the nature of the phenomena being measured in time. There are time-series of continuous phenomena (Fig. 4.24), and "time lines" of "versions" and "mutations," i.e., periods of constant content punctuated by instant changes (Fig. 4.25). Traditionally, the recording and analysis of such information has existed entirely independently of any GIS, but conceivably it will become integrated with the GIS of the future. It should be part of a complete model of geographic information.

When topic is held constant, there is the relation between time and space [Langran & Chrisman 1988]. On the one hand, we can observe when a particular phenomenon occurs at given places. In that case, time is a function of space, e.g., the times of high and low tides at different coastal locations. On the other hand, we can observe the position of a given object at certain times, i.e., position as a function of time. Although space-time functions are usually continuous they are usually recorded as though they were discrete, i.e., they are sampled. The study of the former function is less common, while the latter function, *motion*, is very common and is relevant to navigation. Again, while not yet prevalent in today's GIS world (navigation systems excepted), the temporal-spatial relation should be part of any comprehensive model of geographic information.

#### **4.2.10 Layer Combinations**

If a single layer of data, or a surface, is useful in geographical analysis then a combination of several different layers, drawn from different themes or from different times and relating to the same region, is even more so. For many analysis applications, the integration, or overlay, of two or more of these layers will be required. Conceptually, when we integrate two or more thematic layers or temporal layers for the same space, we have a single, disaggregated, *composite* layer (Fig. 4.26). It forms a

relatively complete model of an environment under study, and can be a "vertical" grouping of the different layers that cover a portion of the world. An example of such integration is when the overlay of two polygon networks yields a composite network of common, smaller polygons.

There are two types of composites, *thematic* and *temporal*. A thematic composite is a grouping of different thematic layers that cover a portion of the world at a given time. A multipurpose cadastre, for example, could contain a thematic composite of current ownership, topography, land use, value, etc., and a natural environmental database could contain a thematic composite of numerous environmental variables throughout a region for a particular time. A temporal composite comprises the changes in content, for a given theme, over time. So, a temporal composite for a legal cadastre, for example, could contain a whole history of land ownership over an entire region.

The physical storage of layers of data and their composites does not necessarily reflect directly the concepts. For some applications, combinations of layers are critical while for other applications, the different layers are otherwise quite unrelated and should be kept apart. And since separation or combination of layers has critical implications for storage and retrieval space and time, a comprehensive geo-data model must allow for different degrees of permanent integration to be specified. The concept could be called *variable integration* where relations between different layers may be totally, partially, or not stored. Note that the same principle applies to both thematic and temporal overlay.

**Management of Thematic Layers.** The simplest approach to handling multiple thematic layers in a geo-database management system is to keep all layers *apart* and stored *independently* of one another. The several small files resulting from this are easily managed on an individual basis. However, there is no capability for analyzing combinations of layers within the DBMS. If layers are to be overlaid and their combinations analyzed, it must be done outside the DBMS, perhaps by special application software or even only visually after being displayed on top of one another.

An important characteristic of some groups of thematic layers is that the same feature

belongs to different themes (Fig. 4.27). A lake, for example, might belong to relief, vegetation, forestry, and soils themes. Because some data are common to more than one layer, such data are duplicated in the independent layer approach. Remember, with duplication, and its inherent storage space inefficiency, there is the potential for inconsistency.

The alternative to independent layers is some form of *overlay*. The approaches to thematic overlay, or combining layers from different themes, may vary according to what type of overlay is performed and to how many layers are involved.

First, I discuss the overlay of two thematic layers. Two types of overlay, *metrical* and *topological*, are identified here.

The *metrical overlay* approach is to "physically" integrate the layers into a single layer (Fig. 4.28). Any feature that intersects those on the other layer becomes fragmented — divided into parts by the edges of the intersecting features. Thus, the resultant layer is a composite of fragmented features, or "greatest common thematic units." The complexity of the process of integrating layers is not discussed here but it is far from trivial and is an important problem in computational geometry. The advantage of integration is that analyses based on combinations of data from the two layers is possible. If elevation and vegetation are integrated, for example, regions that are both covered by spruce trees and more than 1000m above sea level can be retrieved easily. Another advantage of integration is that there is no duplication; the lake that appears on both the elevation and vegetation layers is represented once only. Again, the process of integration is not discussed further, except that it requires solving possible reconciliation problems. The disadvantage of integrating layers is that only one layer may be of interest to one particular analysis and the other layer's information becomes a nuisance. The composite requires a "polygon dissolve" process, effectively removing the other layer.

Introduced here is the notion of *topological overlay* (Fig. 4.29), an alternative to geometrical overlay of two thematic layers. This approach keeps the two layers apart but maintains the topological relations between them. That is, the various joint and

intersect relations, existing between features on different layers, are stored. The result is that individual features are not fragmented, so the advantage of manageable independent layers is kept. However, should a combination analysis be required, data that are critical to the integration process — topology data — are available. Note that the term "topological overlay" refers to the information that is recorded; metrical computations are still required to determine the topology. Also note that certain problems associated with feature duplication remain.

Now I discuss overlay when more than two layers are present.

If the combination of layers in a multi-layer database offers benefits, how many layers should be overlaid? One extreme could be called *total overlay*. That is, all layers are combined at the outset. This allows many combinatorial analyses to be performed easily but incurs a rather high pre-processing overhead. In the case of metrical overlay, a single, possibly huge and highly fragmented file will be required. As more and more layers are integrated, the elements in the resultant composite get smaller in size and larger in number. In the case of topological overlay a large set of interlayer relations will be required. Since not all combinations of layers are always relevant, a compromise (between having independent layers and overlaying them all) needs to be found. The *partial overlay* approach initially keeps all layers apart and performs pairwise overlay as needed for specific analyses. This keeps files at reasonable sizes and allows analyses of critical layer combinations.

Consider an environmental database consisting of a number of discrete-valued layers: elevation, soils, drainage, precipitation, vegetation, habitat, and so on. A total overlay would result in either a single composite layer (the metrical case) with many small elements of homogeneous elevation, soil type, vegetation, etc., or separate layers (the topological case) with many cross-references between every pair of elements that spatially overlap. Suppose combined elevation and soils information is often required while vegetation and habitat combinations are sometimes required. Partial overlay might metrically integrate soils and elevation layers, topologically integrate vegetation and habitat layers, and keep separate the drainage and precipitation layers.

An alternative to the layer-based approaches described above — either having independent layers with no overlay or having layers combined in some manner — is what could be called the *feature-based, or dependent layers*, approach. Essentially, there is a series of "virtual layers," dependent on base, or independent, layers. Each virtual layer comprises "real" and "virtual" features — the virtual features being dependent on features in other layers. Each independent layer comprises only real features. With this method there is no duplication because data common to many layers are stored only once; each feature is only actually stored in its base layer. Because there is no duplication, both potential inconsistency and storage space are reduced. However, because controlled dependency introduces extra relations and operations, retrieval and update become more complicated.

**Management of Temporal Layers.** Three approaches to handling temporal layers, i.e., the changes in the layers of a given theme over time, have been identified [Langran & Chrisman 1988].

The first, "time slicing" is similar to the independent layers approach in thematic layer handling, where layers, or "snapshots," are stored separately. Variations of the technique depend on the periods after which each "snapshot" is taken:

- after fixed intervals, i.e., a layer representing the state of a region is produced at regular intervals, say, every 10 years;
- after any event, i.e., a new layer is produced after any change has occurred; and
- after significant change, i.e., a new layer is produced only after a specified amount of change has occurred (or when significant budgetary allowance has been made!).

The second approach is to have one complete layer at the beginning and a series of "delta" layers — each new layer only containing those features that have changed in the interim. The complete picture at any time must be reconstructed by overlaying all the

delta layers, up to and including that time, onto the base layer. The three variations on time slicing also exist here.

The third approach is to overlay, or integrate, the temporal layers in a similar manner to that discussed with thematic overlay. Again, there are choices about what types of overlay are carried out and how many layers are to be involved.

#### **4.2.11 Summary**

Geographic phenomena have certain characteristics. That is, they can be described using recorded data and can be associated with other phenomena via relations. The characteristics can be grouped into three fundamental domains: topical, spatial, and temporal. That is, each piece of geographic information or geographic concept has thematic, geometric, and historical components, drawn from so-called characterization domains. Meaningful geographic information occurs when appropriate combinations of data from those domains are considered for particular phenomena. Conceptually, these combinations are geographic functions, and include profiles, layers, and composites. In terms of data management, characterization domains represent three fundamental ways that geographic data can be logically partitioned.

### **4.3 The Geographic Abstraction Domains**

As described above, geographic phenomena can be quantified and qualified by data drawn from characterization domains. However, it will be seen that geographic information has additional degrees of complexity, not handled by the characterization domains. I claim that geographic information must also be conceptually partitioned within *abstraction domains*. Abstraction is a powerful concept because it can take something that is complex or concrete, represent or show the essential elements or ideas, and hide from view all the details that are irrelevant for a particular purpose. Independent from the fundamental types of geographic data, described above, are three fundamental types of abstraction dimensions within which geographic phenomena or data may exist. The first, *generalization*, is well known in traditional cartography [Dent



1990; Ormeling 1984; Robinson *et al.* 1984]. Geographic studies exist at many different levels of generalization, or scales — global, continental, regional, local — and so do the appropriate maps and information. The second, *realization*, incorporates different versions of the world, from reality itself, through various kinds of maps of it, to people's conceptualizations. The third, *construction*, is familiar in computing (where it is known simply as abstraction [Shaw 1984]). Geographic data types range from meaningful concepts, such as city, industry, or river basin, down to machine constructs, such as bits and bytes, and each type is built of simpler types. The three domains are examined in detail in this section.

#### 4.3.1 The Generalization Domain

Practically all geographic phenomena are considered at some finite level of detail or complexity, and the scope of any geographic investigation has some finite extent. Also, geographic information has some degree of quality, fuzziness, or uncertainty that indicates its precision and accuracy. For example, there are global, national, regional, and local studies in human geography; there are geological, geomorphic, and engineering scales in physical geography [Hickin 1983]; and topographic maps at one scale are more detailed and reliable than those at a smaller scale. All such issues are encompassed in what is called here the *generalization*, or *scale*, domain. The word "scale" is used here as the general concept, not just the specific map scale found on maps, although the former does encompass the latter. I will claim that geographic data might be available at multiple scales and they must have some indicator of scale.

**Scale Basics.** Some simple definitions of the many important issues related to scale are as follows:

*Map scale.* The ratio of a distance measured on a map and the corresponding distance in reality is termed map scale, or representative fraction (RF). Large scale means a lower level of abstraction, and implies higher quality, i.e., more exact and detailed data. Small scale means a higher level of abstraction, and

implies lower quality, summarized data. (Note, such usage is contrary to that when large scale means something comprehensive or large in scope.)

*Scope.* The extent of any geographic investigation or study is always limited; it has a particular scope. The term "extensive in scope" then, implies wide spatial area, long period in time, or many thematic topics.

*Accuracy and error.* The closeness to which a recorded fact reflects the truth is termed accuracy, or correctness. Error is defined as the difference between data and truth. High accuracy (low error) is closer to the truth than is low accuracy.

*Precision.* The closeness to which a set of recorded facts are to one another is termed precision, or repeatability. High precision means a high degree of agreement among a set of values.

*Resolution.* The smallest item or value that can be distinguished or recorded is termed resolution, or detail. High resolution means greater detail. Lower resolution means less detail.

*Summary.* The characterization of a group of objects by a single attribute, such as the total, mean, or variance, is called summary. It is important for some kinds of analysis and should be incorporated in a GIS [Gahegan & Roberts 1988].

*Lineage.* All data come from somewhere, whether sensed from phenomena via instruments or derived computationally from other data. The background or history of a data set is referred to as the lineage of the data, and can be useful information, especially if data on the accuracy and precision themselves are uncertain. Lineage data could include the who, what, when, where, and how of both data collection and data processing.

*Quality.* The combination of accuracy, precision, and lineage is what is referred to here as data quality, or reliability. Higher quality means less uncertainty.

### 4.3.2 Variability of Scale

Intuitively, the total amount of data involved in a study depends on the breadth and the depth of the study. For this discussion, it can be said, in a non-rigorous manner, that the amount of data depends on both the scope and the scale of the study. *The three items, amount, scope and scale are inextricably related.* The relation is analogous to the equation in physics between mass and the product of volume and density, where mass, volume, and density represent amount of data, scope of study, and scale of study, respectively.

More detailed and reliable studies can be made as the scale increases. However, increasing the scale increases the costs of a study. This is because (a) higher accuracy and precision requires higher measuring and processing costs, and (b) larger amounts of data increase storage and retrieval costs. Ideally, all geographic studies or inventories would be most extensive in scope and of the highest quality. Information derived from such a perfect, albeit massive, database would be, by definition, the most reliable. However, the users of the data, human or machine, are limited in their capability to process large masses of data. The larger the amount, the greater are the storage costs and the slower are the computations. Faced with what could be called a *finite processor*, the set of data for use in a study or investigation must be practically limited, in scope and in quality. Different studies require different scales of investigation [Dueker 1988].

Since different extents and resolutions of data are appropriate for different studies or inventories, a model of geographic information must allow for and keep track of such variety in scale. In a good practical GIS, users work at whatever scale is necessary for the task. Given a particular investigation, the appropriate sized area and level of detail is automatically presented. Such an apparent lack of a fixed scale may be called *scalelessness*. When data for the same objects exists at different scales for different purposes, there can be said to be *multiple generalization*, or *multiple representation*. One characteristic of a multi-scale database is what can be called *scale dependence*, i.e., smaller scaled data depend on larger scaled data (Fig. 4.30), and summary data depend

on detailed data. Note that the spatial dimensionality of certain features depends on the scale at which they are viewed (Fig. 4.31). Of course, the data in a multi-scale model may not necessarily all be actually stored; smaller scale data may be derived from the other data when needed. The provision of multiple scales of data within the spatial domain is being researched [Brueger & Frank 1989; Jones & Abraham 1987] but scale also applies to the topical and temporal domains.

The classical cartographic approach to providing geographic data at differing scales is to produce maps of differing RFs. A process known as *cartographic generalization* is employed, which selects and simplifies certain features from a large-scale map and symbolizes them on a new smaller-scale map. In my concept of a GIS, graphic maps depend on analytic maps (see §4.3.5 below), so cartographic generalization is effectively achieved by two separate major operations, generalization and visualization. Generalization is a transformation between analytic maps and involves deriving less detailed data from more detailed data. Visualization is a projection and symbolization of an analytic map as a graphic map.

### 4.3.3 Recording and Presentation of Scale

Since we almost never know just what the truth is, we never really know how accurate data are. So, it is useful to show estimates of accuracy. An important principle of geoprocessing is *honesty*. Not so much that data must be accurate, although such may be a noble goal, but data accuracy must be realistically recorded and presented. For numerical data, elaborate rigorous statistical studies allow good estimates of data reliability, such as variance and covariance [Mikhail 1976], and for symbolic data, reliability usually involves some measure of probability of membership in a class [Chrisman 1989; Miller *et al.* 1989]. Even if reliable estimates of data accuracy are available, the problem of how to indicate them remains. Perhaps the simplest way is in the presentation of the data themselves — to use resolution or precision to imply accuracy. Intuitively we assume data presented to high precision are also of high accuracy. Unfortunately, however, the resolution and precision of processing normally exceeds the accuracy of data. Although round-off errors are possible during

computations it is possible that machines can carry and display numbers to very many significant digits, even when the original measurements are only reliable to a few digits.

Since the thesis is concerned with storage and retrieval, there is no detailed discussion of the processing of uncertainty data. Suffice to say that with numerical data, gaussian techniques appear to offer a comprehensive solution, but with symbolic data the problem is far from solved.

#### **4.3.4 Scale and the Characterization Domains**

The scale domain lies *orthogonally* to the topical, spatial, and temporal characterization domains. Geographic phenomena can be categorized, positioned, and dated, each to varying degrees of resolution and accuracy. A building, for example, can be classed simply as such, as a public building, or as a library; it can be located simply in Greater Vancouver, in Vancouver's Downtown Eastside, or in Main Street, Vancouver; and it can be dated simply as "old" or to 1926.

Because scale provides data about other geographic data, and it is orthogonal to the other geographic attributes, scale can be considered to be a *meta* attribute, providing data about data. The level of aggregation at which geo-data are given a scale meta attribute depends on the rigor of the study. It could be at the feature level, i.e., any geographic object is characterized topically, spatially, and temporally, and has a generalization designation. It could be at the level of the attribute, where each single attribute has its own accuracy qualifier. Or it could be that only each layer has its own scale designation.

#### **4.3.5 The Realization Domain**

All geographic phenomena and their representation exhibit different forms of reality, from the concreteness of things in the world to the abstractness of concepts in the mind, with various types of models in between [Lukatela 1989; Muehrcke 1990; Van Der Schans 1990]. Thus, geographic information can be partitioned within what is here

termed the *realization domain*. Each level of realization concerns geographic phenomena in different forms: real, surrogate, cartographic, or conceptual. A building, for example, may exist in a real town, be simulated by an object in a database, be represented by a symbol on a map, or be imagined in one's mind.

**Reality, Models, and Maps.** Assume that the *real world*, or part thereof, is to be studied for some purpose — be it management, conservation, or exploitation. The real world is highly complex, consisting of all geographic phenomena and their interactions (Fig. 4.32). It is the environment within which a GIS exists. It is not held as data or information because it is just there; it is reality. Rather than work directly with the real world, it is most convenient to work with a *model*, i.e., some simplified representation or simulation of it. Models can be theoretical or empirical, static or dynamic, physical or abstract, or in the many forms of *maps*. Maps are usually static, empirical models of geographic reality.

**Mental Maps.** Anyone who is thinking of some portion of their environment is holding a mental, or conceptual, map in their mind (Fig. 4.33). Some people can hold very extensive, detailed, and accurate mental pictures of their environments, especially those who spend much time travelling within it and who have a good memory or vivid imagination. Unfortunately, such mental maps are unreliable for most practical and administrative purposes and scientific studies, and so more formal and tangible models are required. Two fundamental forms of such maps are identified here, *graphic* and *analytic*.

**Graphic Maps.** The graphic map is the traditional cartographic form of a static geographic model. A visual representation of the world, i.e., a map, is necessary for many purposes — communication, archiving, learning, or decoration. Information is derived from the map by a human reader via an interpretation process. Typically a map is a vertical, orthographic view of the land, but all kinds of views can be included here — different map projections of large portions of the world, side views of profiles or cross-sections, or oblique-view 3D block diagrams of small regions. The map consists of a network of symbols, usually located using plane coordinates because the map is a

planar medium (Fig. 4.34).

The graphic map itself takes on many forms [Moellering 1980; Nyerges 1980]: A *real* map, such as on a sheet of paper or drafting film, is directly viewable and permanently tangible. A *virtual* map is (i) not permanently tangible, (ii) not directly viewable, or (iii) neither of the two. Example media for the three types of virtual maps are (i) soft copy, such as a computer screen image or slide projected image, (ii) photographic film or slide, and (iii) digital file or database. Even the last kind, the digital (graphic) map, has an important variant in my concept of geographic information. It is the analytic map.

**Analytic Maps.** The analytic map is perhaps the essential component of a GIS. It is the main map that is both archived and is the basis for all digital analysis and transformation. It is not human perceptible but is held in computer or mathematical form (Fig. 4.35). It could be called an analogue or a simulation in the sense that it is a surrogate for reality, consisting of geographic objects and operations. Ideally, objects are positioned within the model using geodetic coordinates because the model is meant as a substitute for parts of the world (which is spheroidal). An analytic map for a particular study region is conceptually complete; it holds all the information — all the themes throughout a region and throughout time — that there is to hold.

#### 4.3.6 Relations Between Levels of Realization

The relations between the above levels of realization are mostly dependencies (Fig. 4.36). Transitions between the different levels require certain complex but fundamental processes. Ultimately, any model depends on the world that it is meant to represent. The possibility of the world being altered because of some modelling activity is not considered here.

An analytic map may be obtained in any number of ways. It may be derived directly from the real world phenomena via surveying and mapping. It may be derived indirectly: from an existing mapsheet or a photogrammetric stereomodel via digitizing or scanning; or from another GIS via some data transfer process.

A graphic map is derived from an analytic map via a cartographic design and construction process, or *visualization*, entailing selection, projection, symbolization, and rendering. For any given analytic map of the world, there may be several different graphic maps existing at one time, each being valid and useful, although perhaps only temporary in existence. Clearly, each visualization is dependent on the underlying analytic map. Ideally, the analytic map changes with the real world, and the derived graphic maps automatically change with the analytic map.

To clarify the difference between graphic and analytic maps, it could be said that a graphic map is a static representation of a part of the world, while an analytical map is a dynamic simulation of it, holding much more information. An analytic map contains more extensive and detailed geographic data than could be conveniently displayed on one graphic map. Analytic maps are stored in a geographic database but graphic maps are not necessarily stored as database objects [Bédard & Paquette 1989]. Graphic maps are situated at the end of the traditional cartographic production process and at the beginning of map users' geographical analysis processes. Analytic maps, on the other hand, are at the centre of a geographical analysis and archival system. Such a system sees graphic maps as by-products and as vehicles for interaction with the analytic core. That is the paradigm shift taking place in cartography — the change in emphasis, from merely producing graphic maps, to maintaining analytic maps with a host of functional capabilities, including simulation, archiving, and visualization [Lukatela 1989; Muehrcke 1990].

#### **4.3.7 Realization and the Characterization Domains**

Like the generalization domain, I consider the realization domain to be *orthogonal* to the topical, spatial, and temporal domains because it is independent of and complementary to such characterizations. Geographic phenomena can be categorized, quantified, positioned, and dated to varying degrees of realization. A building may, for example, really exist at a specific location at a particular time, be simulated in time and



space within an analytic map, be symbolized on a particular graphic map, and be imagined to have some function at some place and time.

Unlike the generalization domain, however, where there is a continuum of levels or scales, the realization domain really only has two distinct levels relevant in digital data processing: analytic and graphic. Thus, designation within the realization domain is perhaps more appropriately made using different object types, analytic and graphic, than it is using a "meta attribute."

#### 4.3.8 The Construction Domain

The next abstraction domain follows from the principle of abstraction in software engineering where programs are built of networks of modules [Aho *et al.* 1982; Shaw 1984]. Each module's *specification*, or logical behaviour, is clearly defined and known to those modules that use it, and each module's *implementation*, or the details of how it works, are hidden from those modules that use it. Thus, geographic phenomena can be modelled using abstract data or object types, each existing at a different level of abstraction, from the high level of geographic concepts to the low level of computing concepts [Bédard & Paquette 1989; Claire & Guptill 1982; Edson 1975; Feuchtwanger 1985; Nyerges 1980; Salgé & Sclafer 1989]. Each object type may be used by some higher level types and is implemented using, or built out of, lower level types. Such an abstraction domain is referred to here as the *construction domain*, although perhaps it could also be called the application or specification-implementation domain.

Three levels are identified here, *applied*, *basic*, and *primal*, respectively containing *specific* objects, for particular applications, *generic* objects, provided as basic building blocks, and *elemental* types, or actual digital data types. Associated with the different levels of object types are different levels of users of the types, from geographers to programmers. A city block may be, for example, a place of commercial or residential activity to a town planner using a database, an areal feature to a database designer, and a series of vectors or rasters to a graphics programmer. Users of an abstract object see its

specification, while an abstract object's implementation, the details of its construction, are hidden from the users.

**Applied Level.** The highest level in the construction domain is the *applied* level. It is that portion of a geographic database seen by an end-user and consists of things of specific relevance to particular geographic themes or applications, such as topography, geology, vegetation, or cadastre. Within a topographic context, for example, applied, or *specific*, types might include roads, rivers, forests, lakes, schools, spot heights, contours, etc., as well as their specific topical and spatial attributes and relations — categories and intersections of roads, connectivity of rivers and lakes, enumeration and classification of forests and their intrusion by roads and rivers, elevations and enclosures of contours, and so on. Each applied type would have many specific instances, such as the Fraser, Columbia, or Skagit Rivers. Also, such topography exists at a particular level of generalization, at a map scale of say between 1:50,000 and 1:100,000. Finally, each applied-level type is implemented using alternative basic-level types.

**Basic Level.** The *basic* level of abstraction of the construction domain is the part seen by an application developer, i.e., one who customizes a geographic database for a particular class of usage. An essential part of this thesis is that *there are a few fundamental geographic objects that have some general characteristics, and are independent of how they are both used and implemented*. Examples of such basic, or *generic*, objects include the feature, layer, and profile, and their general topical, spatial, and temporal attributes and relations, as discussed in §4.2. All of them have some topical characterization, some form of spatial position and extent, such as punctual, linear, or areal, and some temporal description. Additionally, they have scale designations. In a geographic DBMS context they would be provided as the building blocks upon which a variety of applications may be constructed. In turn, they would be built using some of the alternative primal structures.

**Primal Level.** The lowest level of the construction domain is the *primal* level. It consists of elemental or atomic constructs that are relatively close to the machine level of encoding. Such constructs would be used by a geographic DBMS developer to build

basic geographic objects, above. Included are the general numeric and symbolic data types found in programming or data definition languages, such as Real, Integer, Logical, Character, and String. Also included are the special vector spatial structures, such as the node, arc, and polygon, and the special tessellation spatial structures, such as the bitmap and quadtree.

#### 4.3.9 Relations Between Levels of Construction

Unlike in the scale domain, where successive levels of abstraction are dependent on the previous level, or in the realization domain, where all maps depend on phenomena and graphic maps depend on analytic maps, successive levels of abstraction in the construction domain are, in a sense, *independent* of one another. It is important that several different applications are allowed to be built using the same basic constructs and that several alternative primal constructs be made available to represent basic objects. It will be seen that there are many-to-many mappings between levels.

There are three issues concerning the cardinality of *applied-to-basic* relations. First, a single basic geographic object class may be used to represent many different applied geographic object classes. For example, a generic class Regional-feature can be used for a forest stand, a geological outcrop, a cadastral lot, a terrain patch, and so on (Fig. 4.37). Second, remembering that the dimensionality of a feature depends on the scale of investigation, an applied object type may be constructed of more than one alternative basic object type. A city, for example, can be either a point feature or an areal feature (Fig. 4.38). Finally, several instances of applied objects may be simultaneously represented by the same instance of a basic object. For example, several applied linear features, such as county boundary, state boundary, river, and fault-line, may be spatially represented by the same basic line object (Fig. 4.39).

Similar comments can be made regarding *basic-to-primal* relations. Some primal data types are used to implement many different basic object types. For example, the data type Real can represent a coordinate, a parameter, a density, etc., and a raster data type can represent points, lines, or regions. Also, for any given basic geographic object there

may be more than one alternative primal structure used to spatially represent it. Many have been proposed over the years and because different structures are optimal for different purposes more than one may be desired, even for the same object. A region, for example, can be implemented as a polygon vector or as part of a bitmap raster (Fig. 4.40).

That there is a level of abstraction at which geographic concepts are expressed independently of the structures or data types used to represent or implement them may be called *construction independence*, the more applied level being independent of the more basic level. The basic level user must not be aware of, or be encumbered by, the particulars of the raster or vector implementation [Haralick 1980].

#### **4.3.10 Construction and the Characterization Domains**

Again, this domain is *orthogonal* to the characterization domains because descriptions, positions, and dates each exist at all levels of construction. At the applied level, a planning department, for example, is concerned with categories of building types and building uses, with planning zones and street addresses, and with commencements of permits and bye-laws. At the basic level, a geographic database designer uses numerical and descriptive attributes, areal entities, and dates to define such concepts. At the primal level, a geographic DBMS builder uses vectors, integers, reals, and strings, to provide the basic objects.

Also, since there are only three distinct levels identified within the construction domain, it is inappropriate for them to be defined by a "meta attribute," unlike within the generalization domain. The distinction between construction levels within a geographic information model becomes realized in a geographic DBMS — an implementation of the model. The applied level objects are user-defined; they are not provided by the system. The basic and primal level objects are system-defined, the former being provided by the DBMS and used by the database designer, and the latter being provided by the programming language used to implement the DBMS.

#### 4.3.11 Summary

As well as being grouped according to the type of data within multi-dimensional characterization space, geographic phenomena and information can be logically partitioned within a three-dimensional *abstraction* space. Simultaneously, they exist at some map scale and data quality; they take on differing levels of concreteness between reality and mental maps; and they are used at different levels of software construction. The *generalization* domain is a continuum from very large scale and high quality to very small scale and low quality. It provides an additional attribute for describing phenomena. The *realization* domain consists of reality, analytic maps, graphic maps, and mental pictures. The middle two levels distinguish two fundamental types of database maps. The *construction* domain consists of user-defined objects that are specific to given applications, generic objects that are central to a geographic database, and primal constructs that are hidden from users. Such a stratification provides a flexible approach to GIS database design and implementation.

#### 4.4 Conclusion

A synthesis of general logical database model concepts was provided in the previous chapter. The purpose of this chapter was to examine the complex nature of geographic phenomena and information in order to provide additional framework for defining a geographic database model. I have produced a synthesis of geo-data concepts that essentially views geo-data as existing within both characterization and abstraction domains. It can be summarized as follows.

Geographic phenomena can be characterized by three general classes of data — content, position, and time — drawn from topical, spatial, and temporal domains, respectively. Such characterization domains are often partitioned and organized in a recursive hierarchical manner although it is important to allow for non-hierarchical organizations. An individual geographic feature is defined by attributes and relations from each domain. Allowing variation within two domains while holding the third constant

yields a more complex entity, or geographic function. A map layer, for example, represents the relation between content and 2D position for a given topic and time, and a map profile represents the relation between content and 1D position or time. Combining map layers is a useful task in geographical analysis and a range of alternatives exist, in theory, for their management in a database.

A less obvious way of categorizing geographic information is according to three abstraction domains — generalization, realization, and construction. First, consideration of geo-phenomena and geo-data always depends on scale; there is no escaping variations in extent, detail, and accuracy of study. Each scale of representation depends on a larger scale, and the data scale should be provided as a meta attribute. Second, different degrees of reality exist, ranging from the real world, through a sophisticated simulation and a set of alternative graphic renderings, to a conceptual image. While important transformations between them exist within a GIS, most important types of digital maps are analytic and graphic, and both have significantly different purposes. Third, a geographic database model's components exist at different levels of software abstraction, ranging from primal items, manipulated by the computer, through basic geographic items, common to most of geography, to applied items, relevant to specific studies. Availability of these abstract geographic objects, with clearly specified functionality and optional implementations, should allow for flexible and efficient GIS construction.

My main contribution here is the organization of geographic information into six orthogonal domains: three characterization domains and three abstraction domains. A number of specific contributions have also been made:

- I generalized the notion of topical partitionings, i.e., classification schemes (§4.2.2), as being recursive and lattice-like.
- I identified spatial partitionings as being generally recursive and as being exhaustive or non-exhaustive and exclusive or non-exclusive (§4.2.4).
- I produced a taxonomy of geo-referencing (§4.2.4), symbolic and numeric being the two main kinds (Table 4.1).
- I produced a taxonomy of raster encoding (§4.2.4), based on dimensionality, data

type, and divisibility (Table 4.2).

- I produced a taxonomy of spatial calculations and retrievals (§4.2.5), based on number and type of arguments and type of result.
- I showed that most of what applies to the spatial domain also applies to the temporal domain (§4.2.6).
- I produced a taxonomy of layers (§4.2.8), based on continuity, data type, and number of values (Table 4.3).
- I generalized the notion of composite layers to apply equally well in the thematic and temporal cases (§4.2.10).
- I identified the principle of variable integration of layers and recommended its application (§4.2.10).
- I produced a taxonomy of layer management (§4.2.10), layer-based and feature-based being the two main kinds (Table 4.4).
- I generalized the notion of scale to encompass map scale and data quality (§4.3.1).
- I identified the relation among scale, scope, and data volume (§4.3.2) via the notion of the finite processor.
- I identified the two key levels of the realization domain, graphic and analytic, as being the two fundamental types of GIS maps (§4.3.5 and 6).
- I explained the notion of the abstract geographic object and the principle of construction independence (§4.3.8 and 9).

Any of these could be further explored in future work.

While I believe much of the above synthesis of geo-data concepts is applicable to geo-data processing in general, it was done in support of data management. So, in the next chapter, the beginnings of a geographic logical database model, the thesis objective, are proposed.

### **Table 4.1 Geo-referencing**

There are various methods of stating the location of an object:

- symbolic (qualitative)
  - direct (named partitions)
  - relational (in terms of others)
    - topological (bounding objects)
    - hierarchical (component objects)
- numeric (geometric)
  - coordinate based (direct)
  - measurement based (indirect)

### **Table 4.2 Raster Data Structures**

There are various methods of encoding space using an array of cells:

- spatial dimensions
  - 2D (pixels)
  - 3D (voxels)
- cell content
  - boolean (binary)
  - qualitative (nominal/ordinal)
  - quantitative (interval/ratio)
- cell division
  - simple (single cell-size)
  - hierarchical (multiple cell-sizes)
- encoding compression
  - none (uncompressed)
  - compressed



### Table 4.3 Layers

There are various types of 2D geographic functions (content ← location):

#### continuity

discrete (feature network)

points

lines

regions

multiple-types

continuous (surface)

smooth

rugged

stepped

#### content (topical value type)

boolean (binary)

qualitative (nominal/ordinal)

quantitative (interval/ratio)

#### cardinality (number of values)

uni-valued

multi-valued

fixed

variable

### Table 4.4 Layer Management

There are various approaches to the management of multiple layers:

#### layer-based (stored layers)

separate layers

overlaid layers

type of overlay

metrical (full geometry)

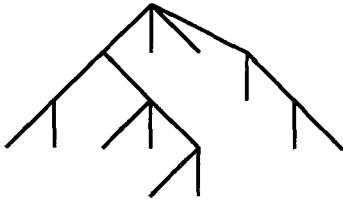
topological (spatial relations only)

degree of overlay

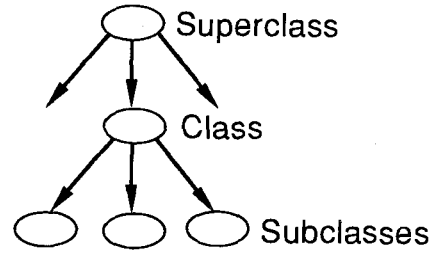
total (all layers)

partial (certain combinations)

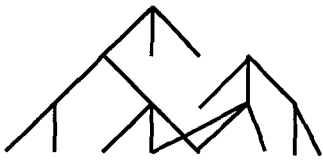
#### feature-based (virtual layers)



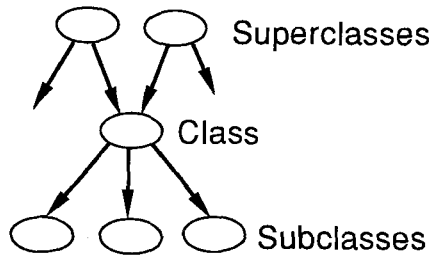
**Figure 4.1 Hierarchy**



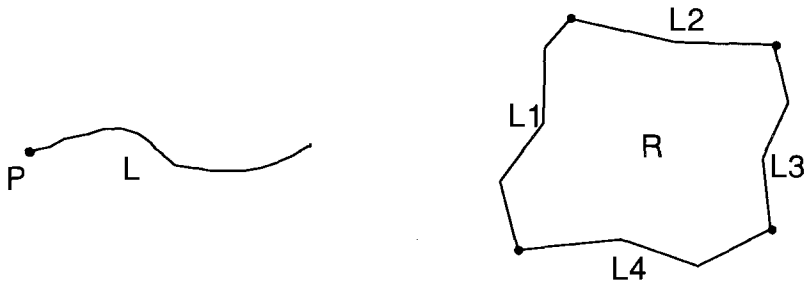
**Figure 4.2 Class Relations in a Hierarchy**  
A class has one parent and many children.



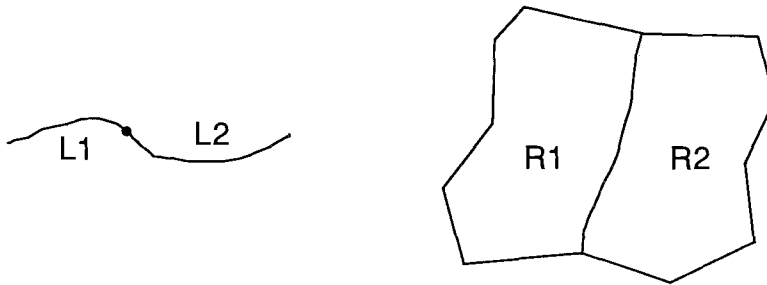
**Figure 4.3 Interlocking Hierarchies**



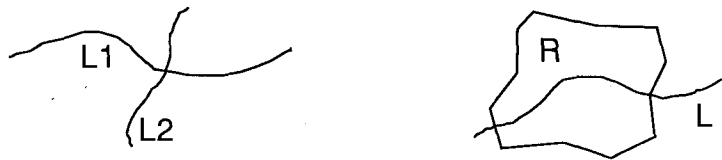
**Figure 4.4 Class Relations in a Lattice**  
A class has many parents and many children.



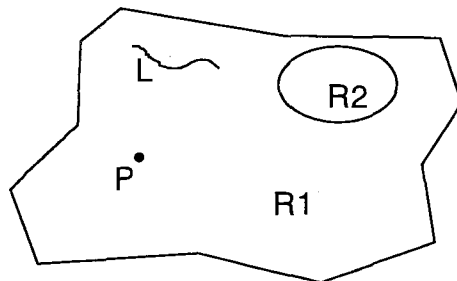
**Figure 4.5 Spatial Incidence**  
 Point P and line L are incident; and lines L1 to L4 form boundary of region R.



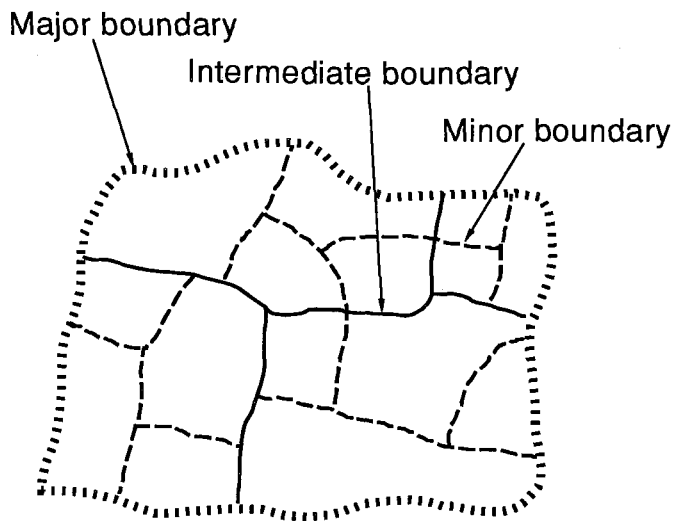
**Figure 4.6 Spatial Adjacency**  
 Lines L1 and L2 are adjacent; and regions R1 and R2 are adjacent.



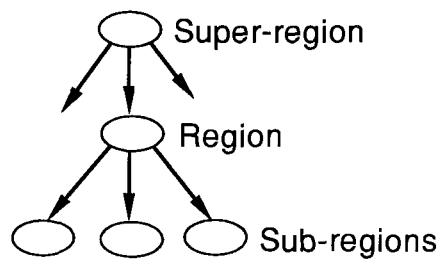
**Figure 4.7 Spatial Overlap**  
 Lines L1 and L2 overlap; and line L and region R overlap.



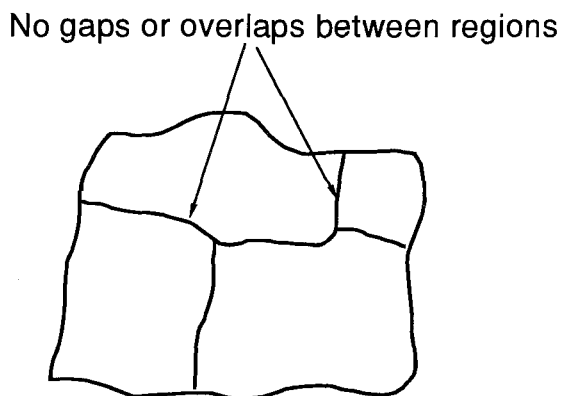
**Figure 4.8 Spatial Enclosure**  
 Region R1 encloses point P, line L, and region R2.



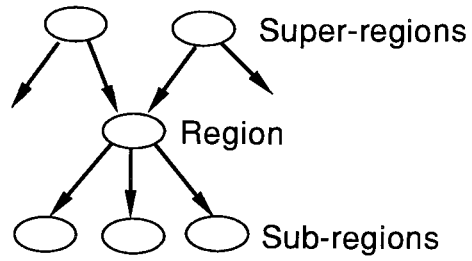
**Figure 4.9 Hierarchical Spatial Partitioning**



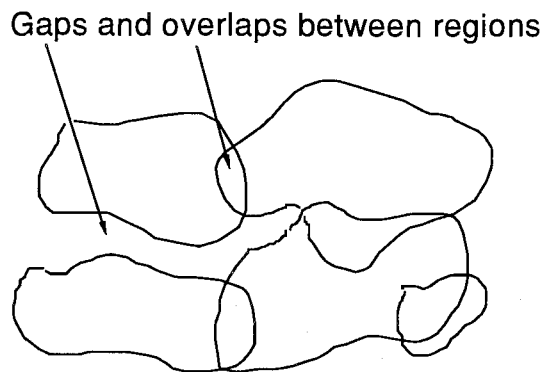
**Figure 4.10 Region Relations in a Hierarchy**  
A region has one parent and many children.



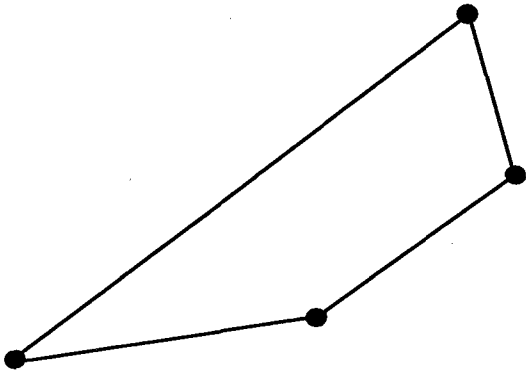
**Figure 4.11 Exhaustive and Exclusive Spatial Partitioning**



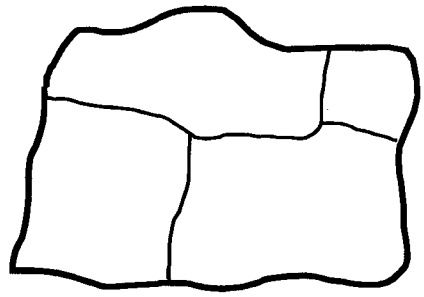
**Figure 4.12 Region Relations in a Lattice**  
A region has many parents and many children.



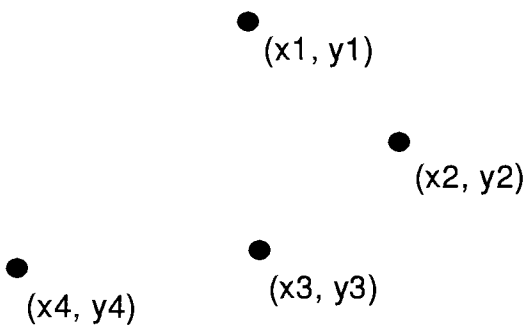
**Figure 4.13 Non-exhaustive and Non-exclusive Spatial Partitioning**



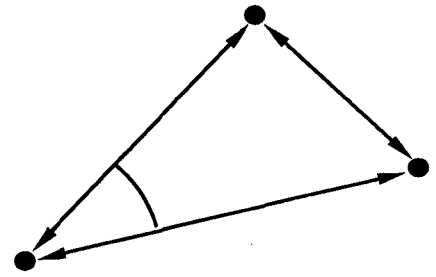
**Figure 4.14 Topological Geo-referencing**  
 A line is positioned by reference to bounding, positioned points; and a region by bounding lines.



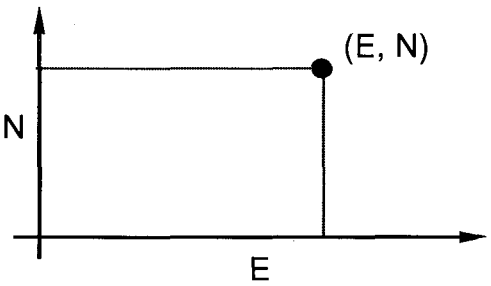
**Figure 4.15 Hierarchical Geo-referencing**  
 A region is positioned by reference to positioned sub-regions.



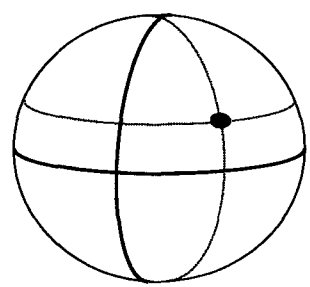
**Figure 4.16 Coordinate-based Positioning**  
 Items are positioned directly, with coordinates.



**Figure 4.17 Measurement-based Positioning**  
 Items are positioned indirectly, via computations of measurements to other coordinated items.



**Figure 4.18 Plane Coordinates**

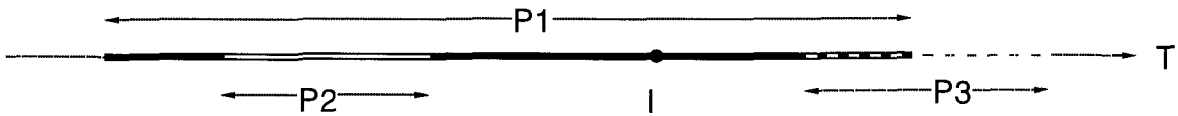


**Figure 4.19 Geodetic Coordinates**



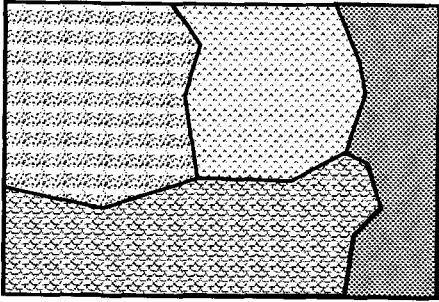
**Figure 4.20 Temporal Contiguity**

Instant I and period P1 are incident; and periods P1 and P2 are adjacent.

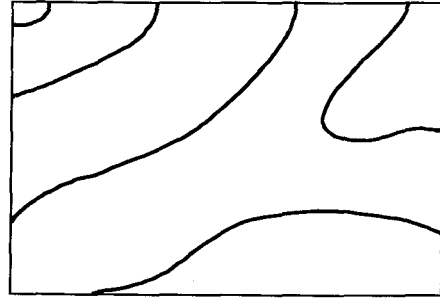


**Figure 4.21 Temporal Intersection**

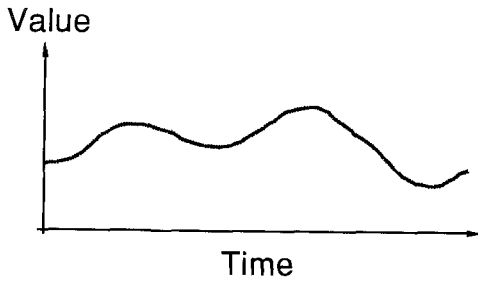
Period P2 and instant I are during period P1; and periods P1 and P3 overlap.



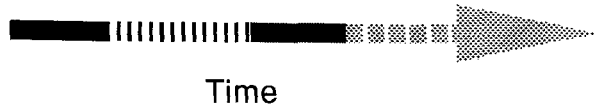
**Figure 4.22 Discrete Layer**  
Content is invariable within certain regions and changes abruptly across region boundaries.



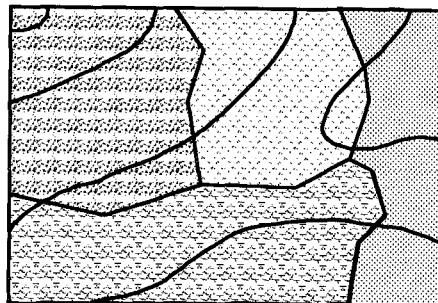
**Figure 4.23 Continuous Layer**  
Content varies throughout entire region.



**Figure 4.24 Continuous History**  
Content undergoes transitional changes.

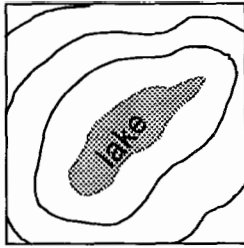


**Figure 4.25 Discrete History**  
Content is constant for certain periods and changes abruptly at certain moments.

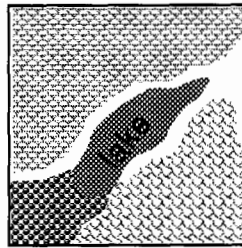


**Figure 4.26 Composite Layer**  
Content is multi-valued, rather than uni-valued.

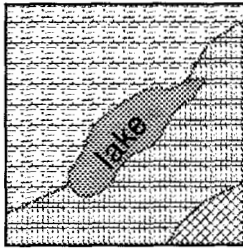




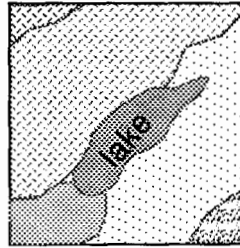
relief



vegetation

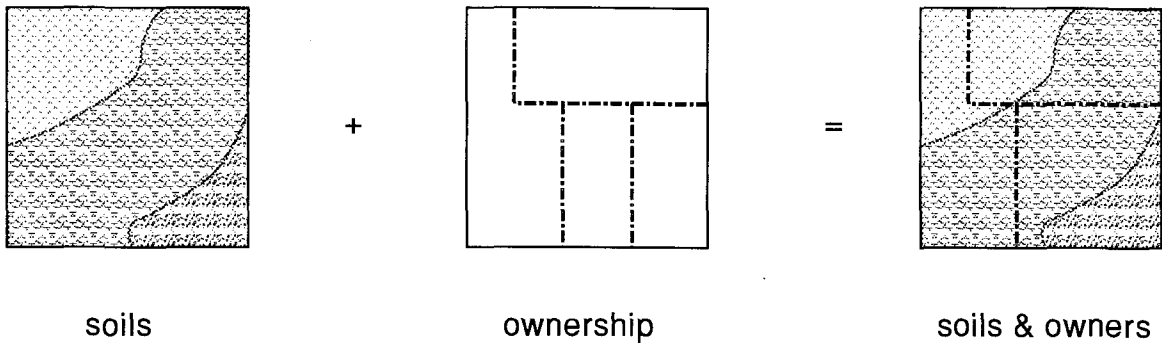


geology



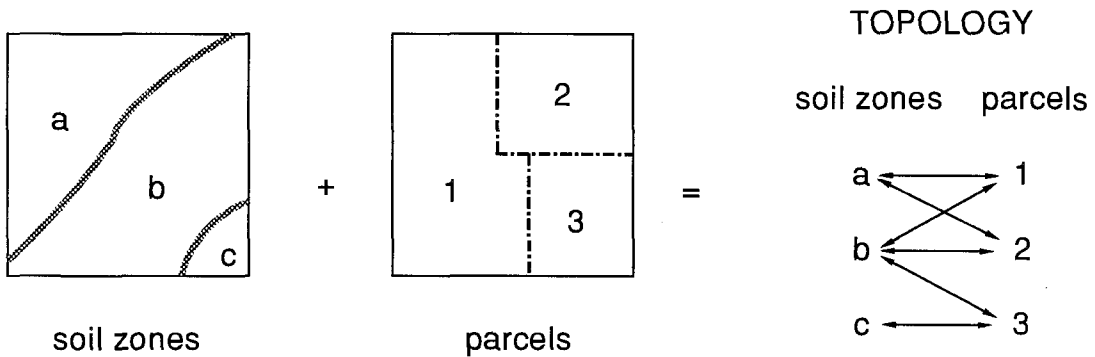
soils

**Figure 4.27 Independent Layers and Redundancy**  
The same feature may be duplicated on different thematic layers.



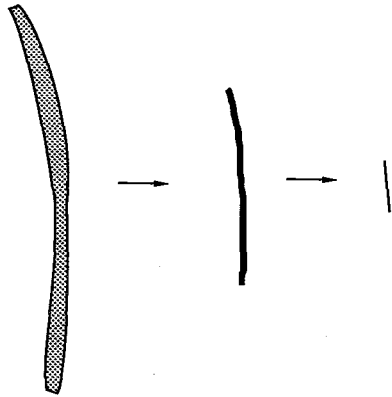
**Figure 4.28 Metrical Overlay of Two Layers**

The soils layer has three soil zones and the ownership layer has four parcels. The composite layer has eight different 'soil-owner' regions.



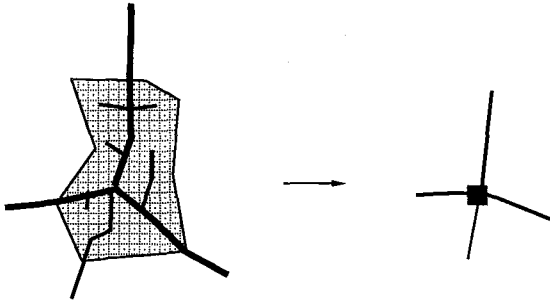
**Figure 4.29 Topological Overlay of Two Layers**

The soils and ownership layers remain separate but the spatial relations between their elements are maintained.



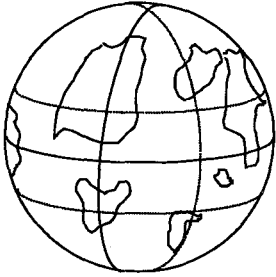
**Figure 4.30 Scale Dependence**

Smaller-scaled representations of a feature depend on larger-scaled representations.

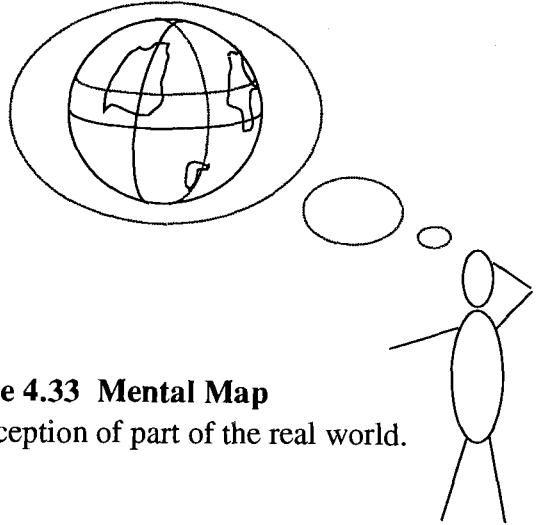


**Figure 4.31 Scale-dependent Dimensionality**

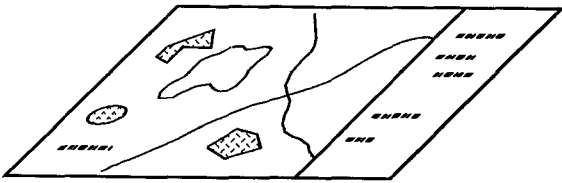
A city is a region or a node, depending on the scale.



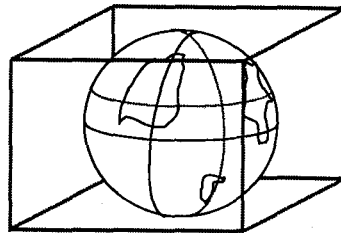
**Figure 4.32 The Real World**  
Infinitely complex.



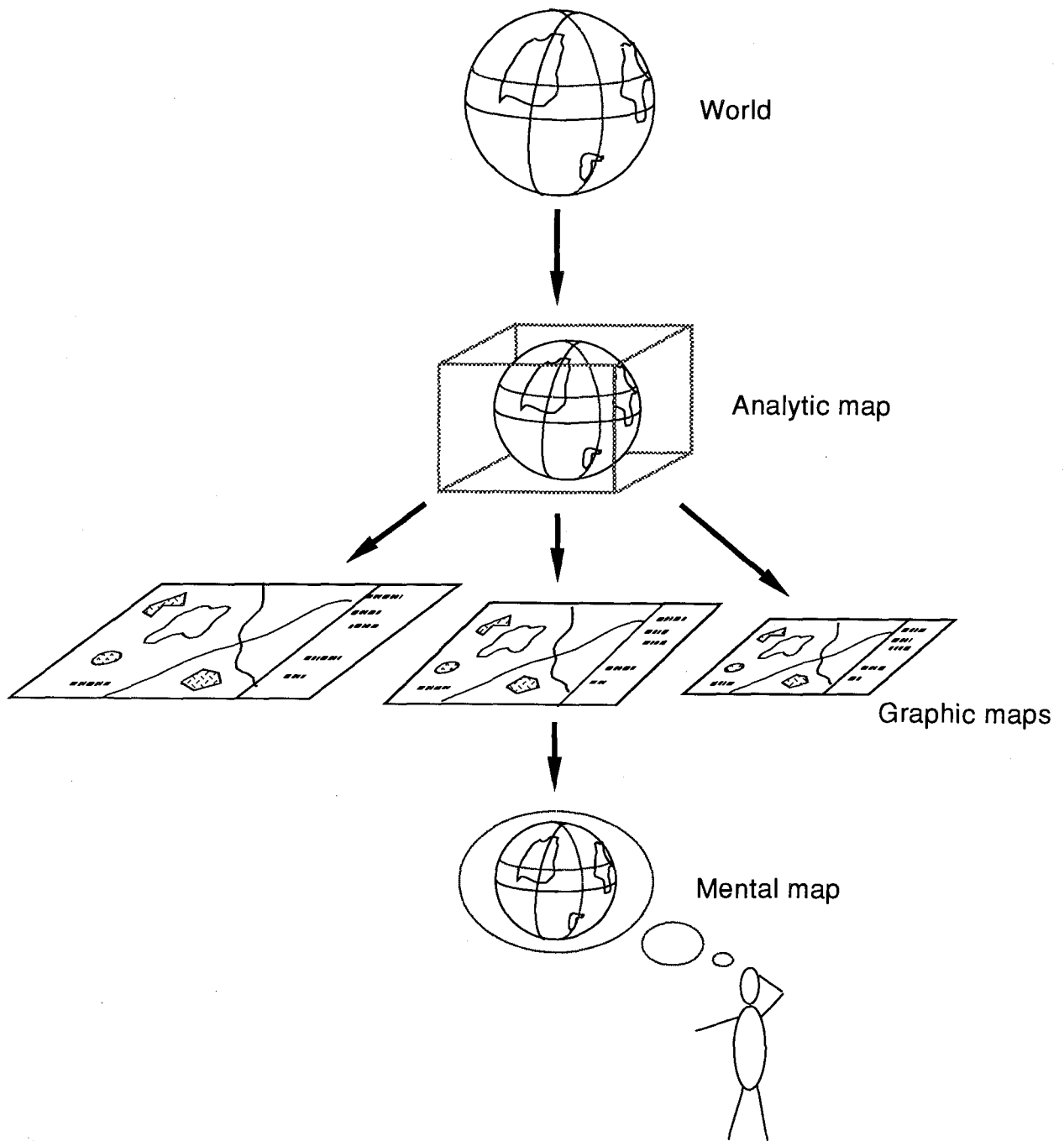
**Figure 4.33 Mental Map**  
A perception of part of the real world.



**Figure 4.34 Graphic Map**  
A model of part of the real world, intended for perception.

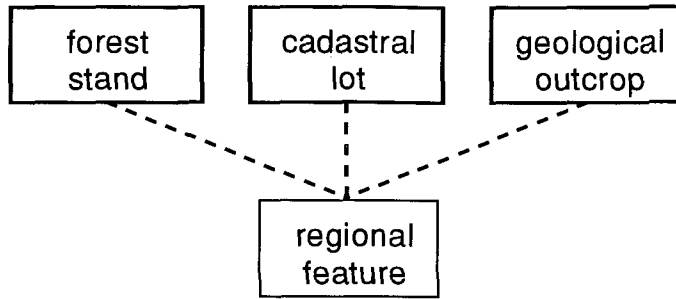


**Figure 4.35 Analytic Map**  
A model of part of the real world, for manipulation and analysis.



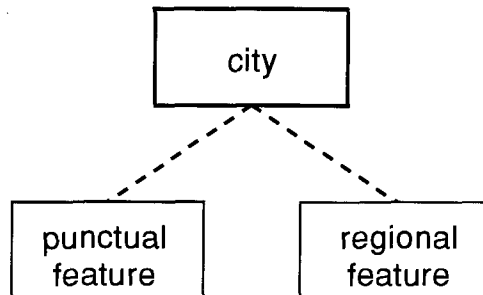
**Figure 4.36 Relations Between Levels of Realization**

Mental maps are interpretations of graphic maps which are visualizations of analytic maps which are representations of reality.



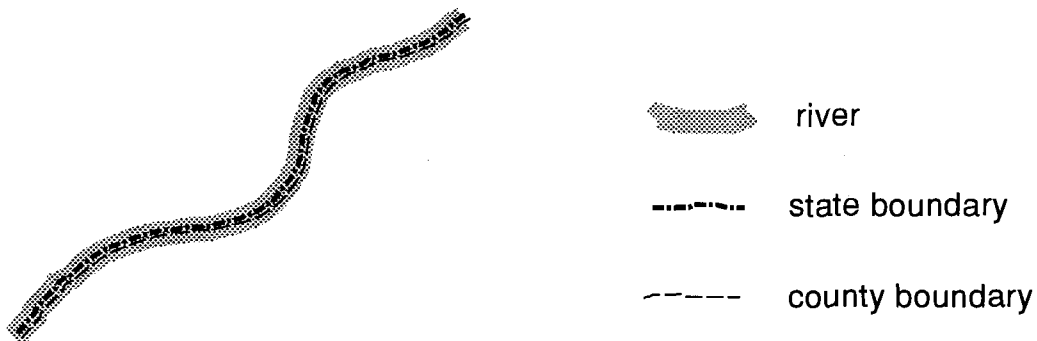
**Figure 4.37 Many-to-One Applied-to-Basic Relations**

Several applied feature classes are implemented using a single basic 'regional feature' class.



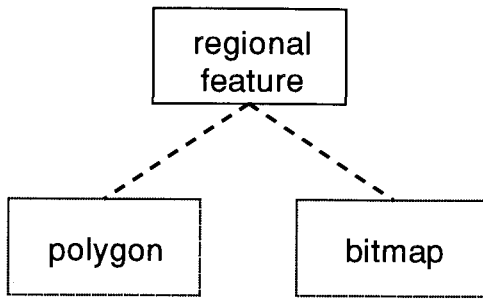
**Figure 4.38 One-to-Many Applied-to-Basic Relations**

A single applied feature class may be implemented using alternate basic feature classes.



**Figure 4.39 Several Features may Share the Same Geometry**

Several instances of applied linear features may be described by the same basic spatial object instance.



**Figure 4.40 One-to-Many Basic-to-Primal Relations**

The basic 'regional feature' class may be implemented using various alternate primal object classes.

# A Geographic Semantic Database Model

## 5.1 Introduction

The purpose of this chapter is to identify a comprehensive set of constructs for the geographic semantic database model (GSDM), the thesis goal. The objective is to merge the results of the previous two phases of the current research — the syntheses of general database models and geographic data concepts — into a broad-based informal model of geographic information management. As seen in §3, the components of general database models are entity, attribute, relation, and operation types. As seen in §4, geographic information specifically requires to be categorized according to topical, spatial, and temporal domains. Also, geographic database objects exist at different scales, make up both analytic and graphic maps, and exist at applied, basic, or primal levels of construction.

The following is a classification of geographic database objects (i.e., entities, attributes, and relations) and operations. Note that examples of such GSDM constructs are, except where stated, within the realm of the analytic map rather than the graphic map. The analytic map is used for digital analysis or inventory purposes, while the graphic map is for output, display, or interaction. The GSDM will be illustrated using *schematics* — diagrammatic representations of geographic database schemas (Fig. 5.1).

## 5.2 GSDM Entities

A GSDM entity is a database object representing a geographic phenomenon that requires characterization by other objects. Mountains, forests, rivers, floods, fires, maps,



forest growths, TINs, spot heights, and terrain surfaces are all possible examples of geographic entities. An entity type is represented in a schematic by a rectangle. Sections 4.2.7 to 4.2.10 provided a theory for defining different types of geographic entities. The most elementary entity is the feature, existing somewhere in topical, spatial, and temporal "spaces." The profile, layer, and composite are larger and more complex geographic entities, extending throughout one, two, or three characterization dimensions. GSDM entities are characterized by GSDM attributes, relations, and operations, and will be described in §5.3, §5.4, and §5.5, respectively.

### 5.2.1 The Feature Entity

The central component of the model is the generic geographic entity, or map *feature*. A geographic feature is something of interest that has a collection of topical, spatial, and temporal descriptions (called taxonomy, geometry, and chronology, respectively) and exists at a particular scale (Fig. 5.2). It can be created, accessed, processed, output, or deleted by users. All the above entity examples also work for features.

There are two fundamental ways of classifying features — both based on a feature's composition. One is based on a feature's decomposition into smaller features (Fig. 5.3). A feature that is not composed of other features is *elemental*. A feature that is composed of other features is a *compound* feature, and its descriptions may depend on its component features' descriptions. A route, for example, may be composed of a sequence of road links and intersections, or a country is composed of a network of provinces. The other way of classifying features is based on a feature's decomposition into topical attribute values (Fig. 5.4). A feature whose content is constant is *homoplethic*. A spot height, a set of lakes, or a soil region are homoplethic if they each have just one topical attribute value. A feature whose thematic content varies within itself is *heteroplethic*. A set of spot heights is heteroplethic because it has many height values and a multi-spectral image pixel is heteroplethic because it has many spectral values. There are three special types of heteroplethic features (Fig. 5.5) — the profile, layer, and composite — described in sections 5.2.2, 5.2.3, and 5.2.4.

So, there are four basic kinds of features. The simplest is the homoplethic elemental feature. Conceptually, it is a collection of points in "geo-space" having a common content. Thus, it could be described as a "greatest common geographic unit," something that has a discrete existence at a particular time and place. Examples of such features are the spot height, surface triangle, contour line, forest stand (or stands, if all are of the same tree type), stream segment (if there are no elevation data), grid, and graticule. The homoplethic compound feature is one that has only one attribute value but is made up of a set of other features. Examples include a mountain comprising a set of peaks and ridges, a forest of stands (if tree types are not given), a river comprising a network of stream segments, and a TIN of points (without heights). A heteroplethic elemental feature contains many attribute values but consists of only one feature. For example, an elevation profile, an elevation grid, a single band of a multi-spectral image, and a choropleth coverage can be represented as single entities having varying contents. The most complex of the four kinds of features is the heteroplethic compound feature. It is a collection of features, each containing a variety of attribute values. A cadastral map, terrain surface, choropleth coverage, elevation TIN, biogeoclimatic composite, and geological cross-section are all possible examples of grouped features with multiple values.

Note that the distinction between elemental and compound features is more syntactic than semantic; the distinction reflects whether or not a database entity is composed of other database entities. A set of lakes, for example, may be represented by a set of individual vector-based entities, but it may also be represented by a single raster-based entity. On the other hand, the distinction between homoplethic and heteroplethic features is more semantic; it reflects how much is known about something. A stream, for example, may be described as being just a stream and only have its location delineated, or it may also have its longitudinal profile described.

### 5.2.2 The Profile Entity

The map *profile* is a special type of "linear" heteroplethic feature. It is conceptually a 1D geographic function, with content depending on theme, place, or time. Thus, there are three types of profiles: *topical*, *spatial*, and *temporal* (Fig. 5.6). A topical profile is simply a set of values from different themes for a given place and time. Examples could include a multi-spectral image pixel, a biogeoclimatic zone, or a terrain unit having an elevation, slope, aspect, and roughness. A spatial profile is a sequence of values from one theme, at one time, along one direction in space — either horizontally or vertically. A sequence of elevations or spectral values along one direction of a map or image could constitute a horizontal spatial profile, while a sequence of rock or soil types or a sequence of water or air temperatures could constitute a vertical spatial profile. A temporal profile is a time-series of values from one theme at a given place. The sequences of various climatic measurements at a weather station could be temporal profiles.

A profile, being a subtype of feature, is also either *compound*, in which case it is composed of a sequence of other features (Fig. 5.7), or *elemental*, in which case it is not. Note that the contents of an elemental profile are stored together as a unit, and individually can only be recovered by application software, not by the DBMS. Finally, a profile may be part of a layer or a composite.

### 5.2.3 The Layer Entity

The map *layer* is a special type of areal heteroplethic feature. It is conceptually a 2D geographic function, where content, for a particular topic at a particular time, depends on position. A layer may extend in horizontal space, in which case it is called a *coverage*, or *surface*, or in vertical space, in which case it is a *section* (Fig. 5.8). A terrain surface, an image, and a choropleth are all coverages, while sections may be "slices" of soils, sediments, or rocks through some 3D entity.

A layer is also either *compound*, if it is composed of other entities (Fig. 5.9), or *elemental*, if it is a single entity. The compound may be of a network of features, such as a choropleth being a network of regions or a contour map being a tree of contour lines, or it may be of a sequence of profiles, such as an elevation grid consisting of a sequence of elevation profiles. If the layer is elemental, its content is stored as a single database object, using some sort of large or complex data type. Finally, a layer may be part of a composite entity.

#### 5.2.4 The Composite Entity

The most complex geographic entity, the map *composite*, is a special type of "volumetric" heteroplethic feature. It forms a relatively complete model of the environment under study, and is conceptually a 3D geographic function, extending both in horizontal space and in theme, height, or time. Thus, there are three types of composites, *topical*, *spatial*, and *temporal* (Fig. 5.10). A topical composite is conceptually a grouping of different topical layers that cover a portion of the world at a given time. For example, a multi-spectral image, a biogeoclimatic map, and a multi-purpose cadastre are topical composites. A spatial composite is a "true 3D" entity, with content depending on 3D position, for a given theme and time. Geology, oceanography, and climatology might require spatial composites. A temporal composite comprises the changes in content, for a given theme, over time. Land-use might be a suitable candidate for such a "carto-movie."

A composite may also be *compound* or *elemental*, depending on whether it is composed of other entities or not. As a compound, a composite may be a network of features, a network of profiles, or a set of layers (Fig. 5.11). An elemental composite, apart from sounding paradoxical, is likely to be a rather large or complex but single database object that is only capable of being analyzed outside the DBMS. For example, a geological mass may be represented by a 3D raster structure, where each voxel is coded for rock-type.

### 5.2.5 The Graphic Map Entities

Every analytic map entity, described above, has zero or more graphic map counterparts. A *graphic map* is a visualization of a part of an analytic map — a subset of themes, within some region, at a particular time, and at a certain scale and projection. It may simply be a single (digital) map sheet or file with no "intelligence," such as a display or plot file. That is, it has no knowledge of its components. It may also be an intelligent collection of graphic map elements, or cartographic features. These include the packaging, such as title, map scale, legend, neatline, grid or graticule, and the core contents — a network of text, point, line, and region symbols that make up the message of the map.

### 5.2.6 Summary

Entities are the main components of the GSDM. Their types range in complexity and include features, profiles, layers, and composites. The feature represents a continuous or discrete geographic phenomenon having a limited topical, spatial, temporal, and scale extent. The simpler kinds of features are homoplethic, having single topical attribute values, and are either elemental or compound entities. Other features are heteroplethic, having multiple topical attribute values. The profile represents a "linear" geographic phenomenon whose content varies throughout one dimension of theme, space, or time. The layer represents a regional geographic phenomenon having a specified theme extending throughout two spatial dimensions at a specific time. The composite is conceptually an integration of layers, based on multiple themes, positions, or times. The topical, spatial, temporal, and scale characteristics of the entities are explained in the following sections.

## 5.3 GSDM Attributes

A GSDM attribute is a database object that describes a characteristic of a GSDM entity. It can take on any value drawn from its attribute domain. An attribute type is

represented in a schematic by an ellipse. At the basic level of construction, attributes are presented below according to the three characterization domains: topical, spatial, and temporal.

At the primal level, according to the measurement scale, there are several different types of general attributes. There are *qualitative* attributes, including nominal and ordinal, and there are *quantitative* attributes, including interval and ratio (Fig. 5.12). Another breakdown of attribute types is that they are either *stored* directly or *derived* computationally. Ultimately, all attributes are built out of the data types and structures provided by the programming language or operating system, upon which the DBMS is built. Conventionally, such types include real, integer, string, character, boolean, BLOB, etc., and structures include the record and array. See §3.3.3 and §4.2.1. The distinction between primal and physical level constructs is relative, not definitive. Perhaps a workable definition is that physical constructs include those that are provided as built-in constructs by the underlying general-purpose data model. Because the sophistication of database models varies, the distinction between primal and physical varies too, with some models providing explicit support for geometric types.

### 5.3.1 Topical Attributes

A *taxonomy* defines the topical attributes of a geographic entity, i.e., it describes what it is. A taxonomy has two components (Fig. 5.13): a *content*, which is a collection of one or more particular thematic attributes, and a *tangibility*, according to whether the entity is physical or abstract in nature. If the feature is homoplethic, the content is simply one attribute. For example, a land-cover unit is described as "forest." If the feature is heteroplethic, the content will have many values. The content of a compound heteropleth will depend on the contents of the component features, such as the elevation of a TIN depending on the elevations of its component vertices, or the radiation intensity values of a remotely-sensed image depending on those of its constituent bands. The content of an elemental heteropleth will most likely be encoded within a BLOB.

### 5.3.2 Spatial Attributes

A *geometry* defines the spatial attributes of a geographic entity (Fig. 5.14). At the basic level of construction, it has two components, a *metrology*, describing the location, extent, size, and shape, and a *topology*, defining some spatial relations to other entities. If the geographic entity being described is elemental, its geometry can be stored directly, while if it is compound, its geometry can be computed indirectly from the geometries of the component entities.

*Location* has two components, an *envelope*, containing the minimum and maximum coordinates of the entity's extent, and a *centroid*, the coordinates of the centre of the entity. *Extent* is the exact space occupied by the entity, and if it is to be stored, it is done so using vector or raster structures, described below. *Size* indicates the entity's length, area, or volume, depending on whether its *shape* is nodal, linear, areal, or solid. *Topology* can have several components, such as incident, cover, and overlap, depending on what other entities touch, lie inside, or share space with the entity in question. See §5.4.2, below.

At the primal level of construction, different implementations of the basic attributes will provide different performance characteristics. See §4.2.4 for a more detailed discussion of the theory of spatial partitioning and encoding. A *spatial extent* is classified according to the two main ways of encoding the space: vector and raster (Fig. 5.15). Vector and raster spatial extents are composed of networks of one or more vector and raster objects, respectively. The subtypes of *spatial vector objects* are dimension-based (Fig. 5.16). Thus there are the *0-cell*, or node, the *1-cell*, or arc, the *2-cell*, or polygon, and the *3-cell*, or polyhedron. The numerous special types and schemes for combining them into compound structures and relating them to one another will not be elaborated upon here. (However, one example of how such elements may be structured *is* shown in Fig. 5.17. Here, straight lines are bounded by a pair of nodes and a pair of polygons; each node has a position; lines may have a computed length; and polygons may have an area.) A *spatial raster object* has two components: a group of *parameters* defining the

type of raster, and the *content* of the raster itself (Fig. 5.18). The type of raster varies according to dimensionality, cell divisibility, cell attribute type, envelope, and resolution (i.e., cell size).

Note that geometry will also contain topical values if the entity being described is an elemental heteropleth. This will be true in the case of, for example, a spatial profile whose spatial extent is a sequence of height values, or a horizontal layer whose spatial extent is a 24-bit raster image.

### 5.3.3 Temporal Attributes

A *chronology* defines the temporal attributes of a geographic entity (Fig. 5.19). It too has a *metrology*, defining the temporal extent, "size," and "shape" of the entity, and a *topology*, defining some temporal relations to other entities. The temporal *extent*, like the spatial extent, details the exact location in time. It can also be encoded in "vector" or "raster" form (Fig. 5.20). The temporal *size* indicates the entity's duration, depending on whether its *shape* is punctual or durational. Temporal vector objects are the 0-cell and the 1-cell (Fig. 5.21), and a temporal raster object has various *parameters* and a *content* (Fig. 5.22).

### 5.3.4 Generalization Attributes

A *scale* describes the generalization attributes of a geographic feature (Fig. 5.23). It includes *map scale* and *data quality*. See §4.3.1. Map scale is an overall indicator of what types of analyses or displays are appropriate for the feature (Fig. 5.24). It is expressed either numerically, as an upper and lower RF (e.g., 1: 10 000 - 1: 50 000), or ordinally, e.g., "large," "medium," or "small." Data quality includes the accuracy, resolution, and lineage of the other data recorded for the feature. Because it is data about data it is really a meta attribute. *Accuracy* of quantitative attributes is expressed in the same units, while accuracy of qualitative attributes may be expressed as a probability of inclusion within a category. *Resolution* may be implicit in the units of the attribute or be explicitly stated. *Lineage* is a sequence of the database transactions, with dates, that



have taken place to arrive at the current state of the feature. Thus, lineage will indicate whether the data are "raw," i.e., given to the system from outside, or have been derived from other data within the system.

### 5.3.5 Summary

A geographic entity has four main attributes — taxonomy, geometry, chronology, and scale — defining its topical, spatial, temporal, and generalization characteristics. The first three, respectively, define the what, where, and when of the entity while the last defines its scale of existence. These basic attributes are constructed of symbolic and numeric data types, and raster and vector data structures, which in turn are constructed out of built-in data types. Other characteristics of entities, how they may be related to other entities and how they are acted upon, are described in the next two sections.

## 5.4 GSDM Relations

A GSDM relation is a meaningful link or association between (usually) two GSDM objects. A relation is represented in a schematic by a line. It may be stored in the database or be derived when necessary. Semantic, topologic, and abstraction relations will be described below. At the syntactic level, relations between stored data items are usually described under the heading of data structures or data constructors. There are different types of sets, sequences, arrays, trees, and networks. They are built either by storing adjacent elements consecutively in storage or by using pointers — special data that refer to other elements. Relations in general were discussed in more detail in §3.3.4.

### 5.4.1 Semantic Relations

General-purpose, semantic relations include *has-subtypes*, *described-by*, and *composed-of*. The *has-subtypes* relation links a more general type, such as Waterbody, to its subtypes, Lake, River, Marsh, Lagoon, etc. Basic examples of this semantic generalization were shown in Figs. 5.5, 6, 8, 10, 12, 15, 16, 20, 21 and 24, where several subtypes "hang down" from a general type. Both the *described-by* and *composed-of* relations are forms

of semantic aggregation. Described-by connects an entity to its describing attributes, such as Lake to its Type, Location, Area, etc., or generically as in Fig. 5.2. Composed-of relates a compound feature to its spatial or temporal parts, such as Province to its component Regions, or History to its sequence of Events. Basic examples were shown in Figs. 5.3, 4, 7, 9 and 11, where components are "hanging to the side" of the compound entity. The relation can be used with different grouping structures according to whether the compound feature is a sequence, tree, or network of other features. Also, it has variants depending on the exhaustiveness, exclusiveness, homogeneity, and contiguity of the subdivision. See §4.2.4. Composed-of also relates a compound attribute to its component attributes, such as in Figs. 5.13, 14, 17, 18, 19, 22 and 23.

### 5.4.2 Topologic Relations

Additional to the composed-of relation are *bounds*, *overlaps*, *contains*, *inside*, and *during*, describing spatial and temporal topologies among entities. These were discussed in detail in §4.2.4 and §4.2.6. They are represented by components of the spatial and temporal topology attributes and by the *set* data structure.

Spatial topology is shown in Fig. 5.25. The *bounds* and *overlaps* attributes respectively provide the sets of other features that are incident on and overlapping the feature being described. The *contains* and *inside* attributes respectively describe what features the subject encloses and those other features within which it lies. Note that the contained features are those other than the ones required as parts in a semantic composition.

Temporal topology is shown in Fig. 5.26. The *bounds*, *overlaps*, *contains*, and *during* attributes provide the sets of other features that are incident on, overlapping with, occurring during, and containing the described feature, respectively. Again, *contains* refers to features occurring during but not constituting part of the subject.

### 5.4.3 Abstraction Relations

The last three GSDM relations exist within the three geographic abstraction domains. They are the generalized-as, realized-as, and implemented-as relations.

*Generalized-as* relates an entity at one scale of representation to the same entity represented at a smaller scale. For example, at a large scale, a drainage basin may be represented by a complex network of smaller basins, each with its own detailed geomorphometry; at a smaller scale it is a simpler entity with a less detailed characterization; while at the smallest scale the basin is simply a region with one summary geomorphometric attribute.

The *realized-as* relation links an analytic version of an entity to one of its graphic versions. An analytic entity is intended primarily for statistical analysis and simulation or modelling, and represents a phenomenon "as it is." A graphic entity is intended for human perception, i.e., for rendering on a cartographic output device or possibly for audio output.

The *implemented-as* relation lies in the construction domain. It functions to link an object at one level to its software implementation at a lower level. Applied level objects, such as a peak, DEM, stream, image, height, flow, and reflectance, are implemented as basic level objects, such as feature, layer, composite, and quantity. Basic objects, in turn, are implemented as primal or physical level objects, such as vector, raster, real, and integer. This way, users working at one level need not worry about the details of a lower level, and different vector or raster implementations of a layer, for example, may be tested or used for different performance characteristics without affecting the logical behaviour of the layer.

#### 5.4.4 Summary

Geographic relations define associations between geographic objects, and there are three main families. Semantic relations link object types to their sub-types, compound objects to their component parts, and entities to their describing attributes. Topologic relations link entities that are spatially or temporally joint or intersecting. Abstraction relations link objects at different levels of generalization, realization, and construction. Some relations may never be stored but may be derived at run time, in which case they are computed relations.

### 5.5 GSDM Operations

A GSDM operation is something that retrieves, processes, or updates information in a GSDM database. Generally, primitive database operations (discussed in more detail in §3.4.2) include retrieve, calculate, update, and interact, and are used in combination to effect queries and transactions. A retrieval gets data from the database given some criteria. An update either stores new data, deletes old data, or changes existing data, in the database. A calculation is given arguments, or operands, does some computations, and gives results. An interaction exchanges data with a user or device.

GSDM operations manifest themselves in two main ways. First, as *database manipulations* defined for particular object classes. These include retrievals, updates, and certain others. Second, as special types of attributes — *computed attributes*. For example, much of the geometry attribute of a compound feature is likely to be computed from attributes of component features rather than directly stored. Another way that GSDM operations exist is in the form of primitive calculations, such as comparisons, additions, and intersections, that appear in database manipulation predicates or in the definitions of computed attributes.

A database manipulation has two components: a *selection* of a particular part of the database, and an *action* to be performed. Suppose we have a multi-purpose cadastral

database and we wish to print the names of all the lot owners in a certain city block. The selection would isolate all the Lots of the Block and the action would print the Owners' Names.

A computed, or dependent, attribute will consist of a series of database manipulation operations that derive a value from other attributes when necessary. A city Block's Area attribute, for example, might involve retrieving and summing the Areas of its component Lots.

### 5.5.1 Selections

A selection logically isolates a specific subset of a database that is to be acted upon. As such, it is part of a database manipulation. However, it can also be used to define external databases, as subsets of a conceptual database. A selection is made by defining a subschema, a subject, and a predicate:

**Subschema.** This establishes a context for the operation. It is a subset of the database's schema, including all the entities relevant to the operation. All attributes of and relations between the entities in a subschema are also implicitly part of the subschema. In the given cadastral example, the subschema would comprise the Block and Lot entities, while their individual attributes and their composed-of relation would be implied.

**Subject.** This indicates the specific database item or items to be acted on. It may consist of entities, attributes, or relations, and so is either part of or implied by the subschema. In the example, the Owner attribute is the subject.

**Predicate.** This is the condition, or constraint, that distinguishes the particular instance, or instances, of the subject from the other database objects. It can be any boolean expression, holding true for the individual occurrences of the subject to exist. In the example, the predicate would be used to isolate the particular city block. This may be

Block (ID) = "VB138"

if a block's ID is given, or

Block contains Cursor

if a cursor is pointing at the block.

A predicate consists of a set of zero or more calculations. Each calculation will involve either (i) attributes and values, or (ii) entities and *probes* (see below). In the first case, the attributes must be part of the subschema, and the attributes and values must be type-compatible. In the second case, the entities must be part of the subschema, and the entities and probes must be type-compatible. In either case, the calculations may be topical, spatial, or temporal.

**Probe.** A certain object, though never stored as a database object, should still be defined. A *probe*, or cursor, defines an arbitrary set of topical, spatial, temporal, and scale extents and is used in defining database selections. It could be described as an *interaction object* because it enables a user to interact with a database. It is an extension of the notion of a query rectangle, so it may be punctual, linear, regional, or solid in its spatial extent; punctual or durational in time; and so on. During interaction with the database a user specifies a probe and asks for those database objects related to the probe, i.e., objects inside, touching, containing, or near it.

**Calculations.** A calculation performs the essential data processing component of any database operation. It may be a standard arithmetic or textual one (see §3.4.2 and §4.2.2) or a special geometric or temporal one.

Metrical processing of spatial data includes a variety of spatial calculations (see §4.2.5): those that take two entities as arguments and yield a boolean result, such as coincide, contains, and joins; those that take one operand and produce a metrical result, such as size; those that take one operand and produce a geometrical result, such as envelope, centroid, skeleton, expand, and smooth; those that take two operands and yield a metrical result, such as separation,

angle, and direction; and those that take two operands and yield a geometrical result, such as intersection, union, and difference.

There are two kinds of temporal calculations (see §4.2.6). First, are the unary operators, such as length, centre, or extension, and second, are the binary operators, such as before, after, separation, and intersection.

The amount of information a selection may yield can vary greatly. A selection may yield only one attribute or relation of a single feature, any combination of attributes and relations of a given feature, or all the features intersecting a "large" probe. A probe may be specified as having any combination of attribute value limits.

### 5.5.2 Actions

The action component of a database manipulation specifies what is to be performed on the database components identified in the selection. The most essential actions, applicable to all types of objects, are *retrieve* and the three kinds of *updates* — *create*, *delete*, and *modify*. The *retrieve* primitive simply assigns data that have been selected to some variable for later processing. The *create* primitive inserts new data at the place indicated by the selection. The *delete* and *modify* primitives, respectively, remove and change the selected items. Other actions, such as the various kinds of *interacts* primitives, will depend on the different types of object being manipulated. For example, *input*, *display*, and *print* will behave differently depending on whether the data are graphical or textual.

### 5.5.3 Queries and Transactions

Higher-level database operations, invoked by users of the DBMS, are termed queries and transactions. These are compounds of primitive operations and are roughly analogous to retrievals and updates, respectively. Various spatial and temporal queries and transactions can be performed. While they are summarized here, they were detailed in §4.2.5 and §4.2.6.

One class of spatial query includes those that get attributes or relations of a given entity without requiring any metrical computation. Some are metrical in the sense that they get positional attributes of an entity. Many simple entity-based queries are topological because they get connected entities. Another class of spatial query gets entities that are metrically related either to given probes or to given entities. They are achieved by doing some computations and retrievals. Position-based, or probe-based, entity-yielding queries include the range, point-in-polygon-network, and the nearest neighbor queries. The metrical neighborhood query is an example of an entity-based entity-yielding query. Spatial transactions include adding, deleting, moving, joining, and dividing point, line, or region features.

Two kinds of temporal queries are, first, those that ask when or for how long an entity existed, or those that get the entities existing before, during, or after an entity, and second, those that get entities related to a probe. As for temporal transactions, the date of a geographic feature may be created, deleted, or changed. Also, there are temporal equivalents of joining and dividing features.

**Higher-level operations.** There are many GIS operations that work at the level of the map profile, layer, or composite, and are relatively long and complex. These include operations such as digitize map, generalize map, join maps, produce graphic map, overlay maps, and separate layers. They are not further considered here as they are likely to involve considerable processing by specialized application programs or hardware that are outside the DBMS.

#### 5.5.4 Summary

GSDM operations consist of queries that extract information from and transactions that change the information in a geographic database. Both are made up of combinations of more primitive operations, i.e., retrievals, calculations, updates, and interacts. Practically, database manipulations are operations consisting of a selection, that isolates the part of the database to be worked on, and an action, that performs the actual



operation. The problem of GSDM operations is far from solved because it is unclear where the boundary between DBMS and application programming for the processing of complex entities should be drawn.

## 5.6 Conclusion

The previous two chapters provided a framework for defining a geographic semantic database model (GSDM). General database theory gives us database models consisting of classes of entities, attributes, relations, and operations. And my geographic semantic theory provides geo-information domains of content, space, and time, and provides abstractions of scale, of analytic and graphic maps, and of primal, basic, and applied construction levels.

This chapter provided a step towards specifying the GSDM. The model contains entities that represent geographic phenomena, that are described by attributes and by relations with other entities, and that can be retrieved, processed, or updated by operations. Depending on the nature of usage, GSDM constructs are either primal, basic, or applied, the basic ones representing generic geographic constructs. A fundamental type of entity is the feature, something that has a distinct topical, spatial, temporal, and scale characterization. Features may be grouped into larger or more complex features, such as the map layer, where content is a function of position for a given topic, time, and scale. Information on features comes in different scales, with the smaller scales depending on the larger ones. A feature's realization is either analytic if it is meant to simulate a real entity, or graphic if it is only for viewing. The topical, spatial, and temporal characterizations of an entity exist within taxonomy, geometry, and chronology attributes. Relations between GSDM objects represent different semantic, topologic, and abstraction linkages. Manipulation of database objects can be achieved using declarative constructs consisting of selections and actions.

The next chapter discusses my GSDM *vis-à-vis* other models.



entity type



relation type

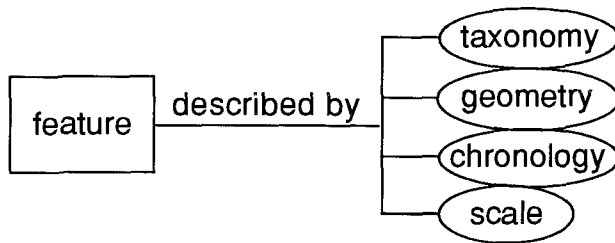


attribute type

**Figure 5.1 Elements of a GLDM Schematic**

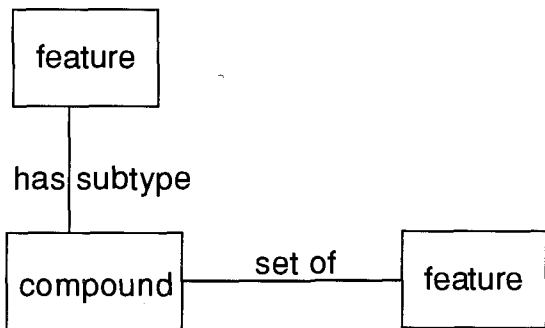
An entity is something that must be described. An attribute is something that only describes another thing. A relation represents an association between items.

[Chen 1985; Su *et al.* 1988]



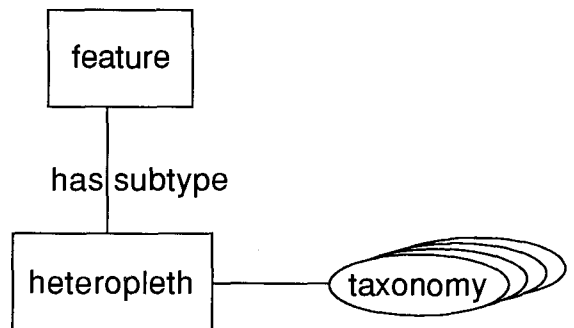
**Figure 5.2 Generic Feature**

Each geographic feature has an aggregation of topical, spatial, temporal, and generalization descriptions.



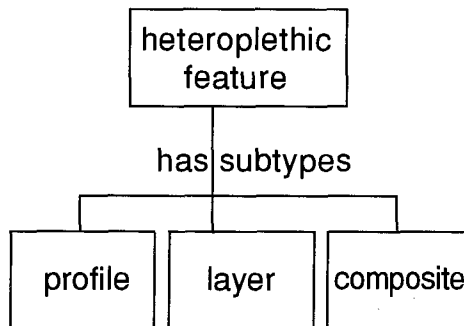
**Figure 5.3 Compound Feature**

A compound is a special feature that is composed of other features.



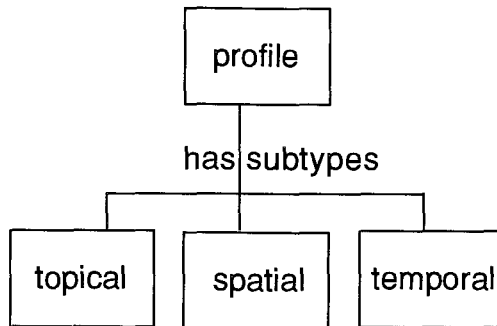
**Figure 5.4 Heteroplethic Feature**

A heteropleth is a special type of feature whose topical description consists of many values.



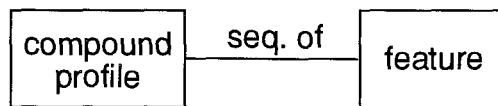
**Figure 5.5 Heteropleth Types**

The profile, layer, and composite are possible special types of heteroplethic features.



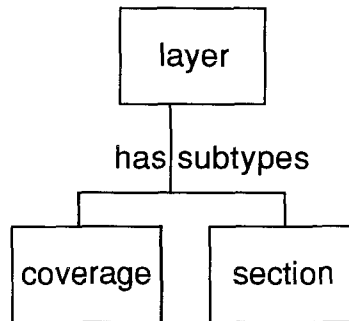
**Figure 5.6 Profile Types**

There are three kinds of profile feature — topical, spatial, or temporal — according to whether its content varies with theme, place, or time.



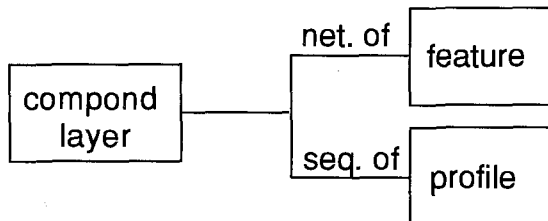
**Figure 5.7 Compound Profile Components**

A compound profile is composed of a sequence of features.



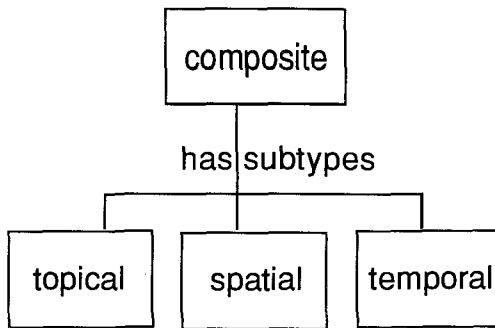
**Figure 5.8 Layer Types**

A layer feature is either a coverage, if it extends over a horizontal surface, or a section if it is a vertical face.



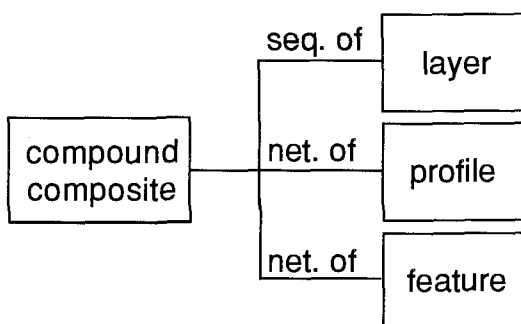
**Figure 5.9 Compound Layer Components**

A compound layer is composed of either a network of simple features or a sequence of profiles.



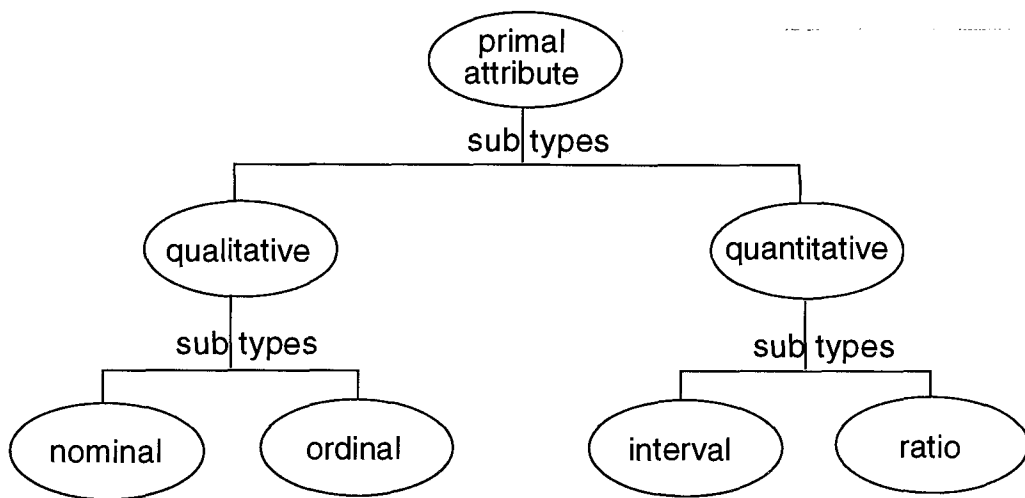
**Figure 5.10 Composite Types**

There are three kinds of composite features. A topical composite is made up of different themes for a region. A spatial composite has 'true 3D' variation of a theme. A temporal composite is a history of a region for one theme.



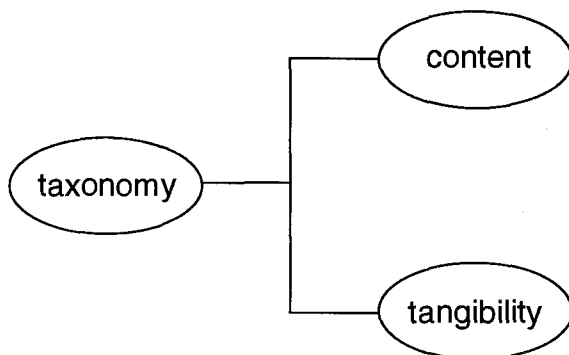
**Figure 5.11 Compound Composite Components**

A compound composite must consist of a sequence of layers, a network of profiles, or a network of simple features.



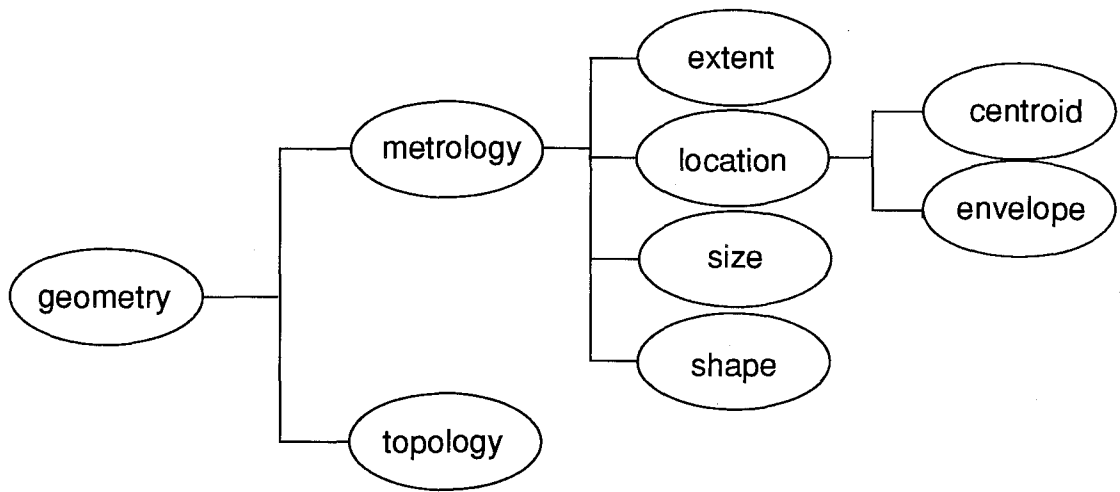
**Figure 5.12 Primal Attribute Types**

Primal attributes are either symbolic or numeric.



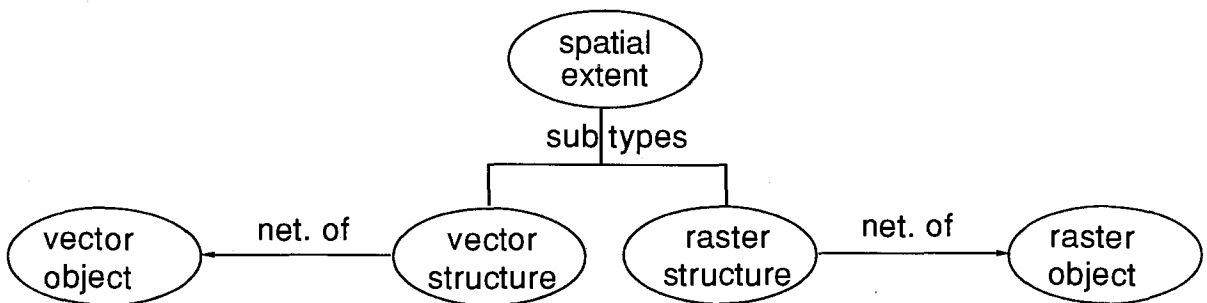
**Figure 5.13 Topical Attribute Components**

A 'taxonomy,' describing the thematic nature of a feature, has a 'content' and a 'tangibility.' Content details the value or values of the thematic variable describing the feature while tangibility indicates whether the feature is physical or abstract in nature.



**Figure 5.14 Spatial Attribute Components**

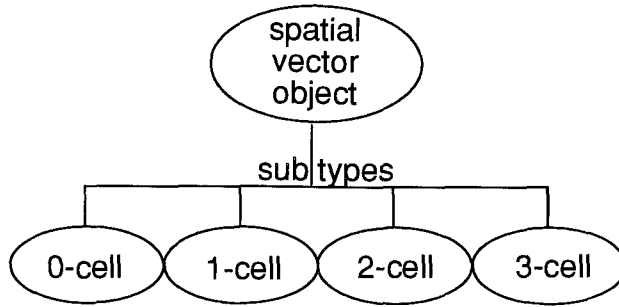
A 'geometry,' describing a feature's spatial nature, comprises a 'metrology' and a 'topology.' Spatial form is detailed in the 'extent' component of metrology and summarized in the 'location,' 'size,' and 'shape' components. Spatial relations may be provided in topology.



**Figure 5.15 Spatial Extent Types and their Components**

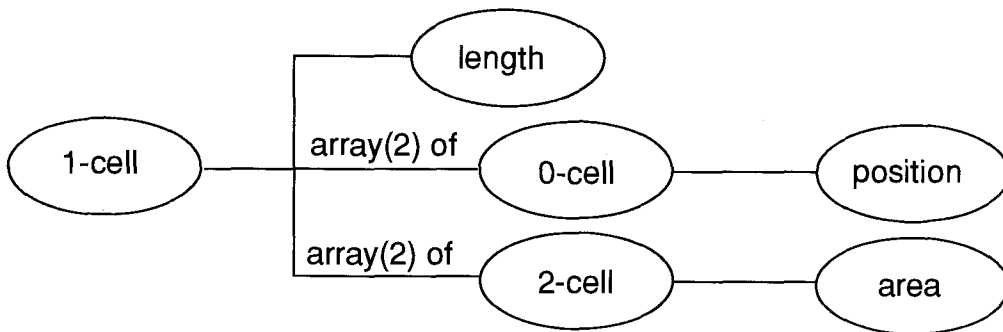
The physical space occupied by a feature can be encoded using a vector or raster structure. A vector structure comprises a network of vector objects, while a raster structure, raster objects.





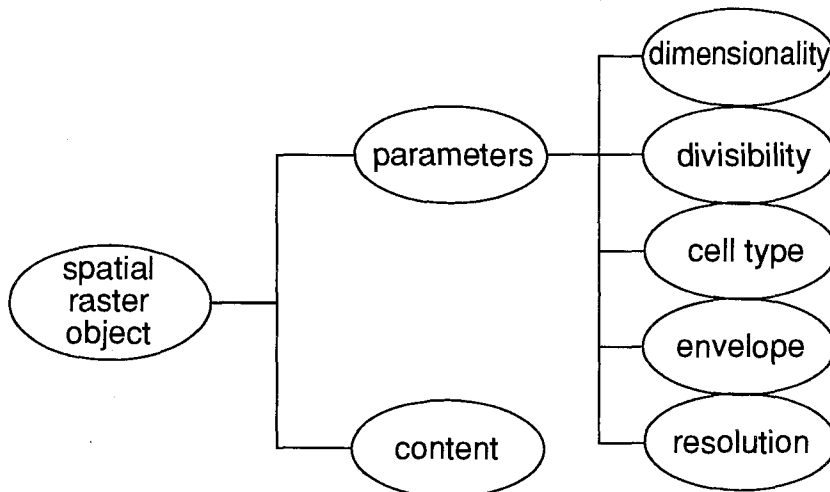
**Figure 5.16 Spatial Vector Object Types**

The node, arc, polygon, and polyhedron are types of spatial vector objects.



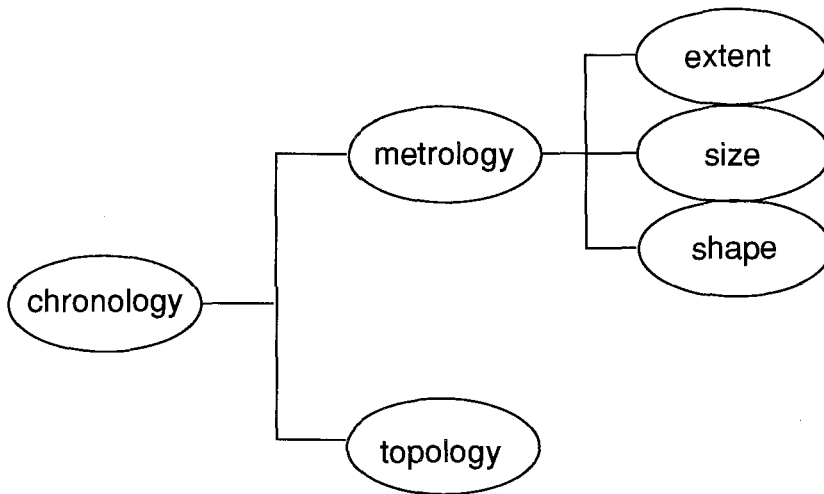
**Figure 5.17 Vector Network**

A vector network could be structured such that each arc has a length, a pair of bounding nodes, and a pair of bounding polygons; each node has coordinates; and each polygon an area.



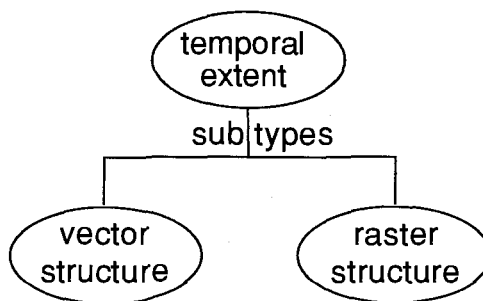
**Figure 5.18 Spatial Raster Object Components**

A raster object consists of the tessellation itself and parameters defining its type.



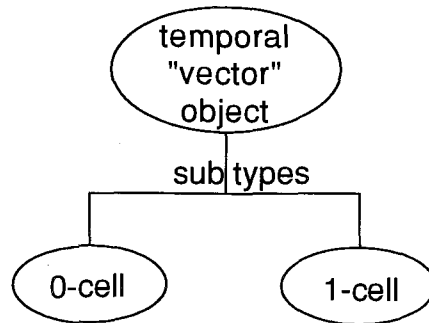
**Figure 5.19 Temporal Attribute Components**

A 'chronology,' describing a feature's temporal nature, comprises a 'metrology' and 'topology.' The 'extent' of a feature's history details when the feature occurred, 'size' indicates its duration, and 'shape' whether it is a period or an instant. Topology may provide temporal relations.

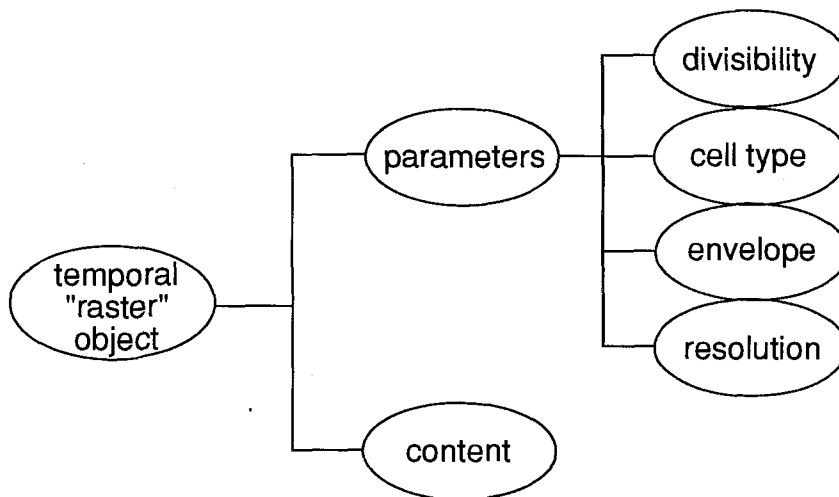


**Figure 5.20 Temporal Extent Types**

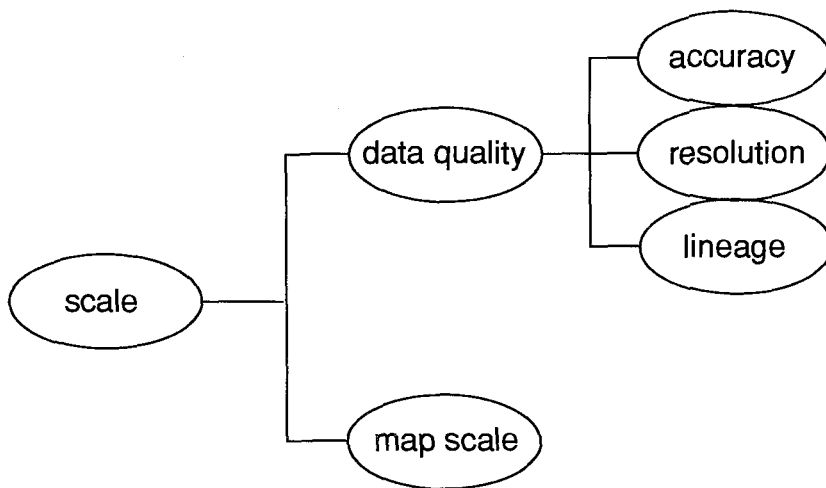
The exact time occupied by a feature can be encoded in temporal equivalents to the vector and raster structures. A temporal vector structure comprises a sequence of temporal vector objects, while a temporal raster structure, temporal raster objects.



**Figure 5.21 Temporal Vector Object Types**  
 A temporal vector object could be an instant or a period.

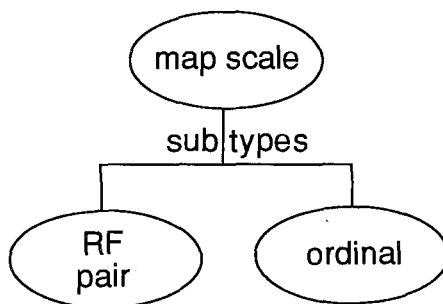


**Figure 5.22 Temporal Raster Object Components**  
 A temporal raster object will have 'parameters' defining its type and the actual array of time cells.



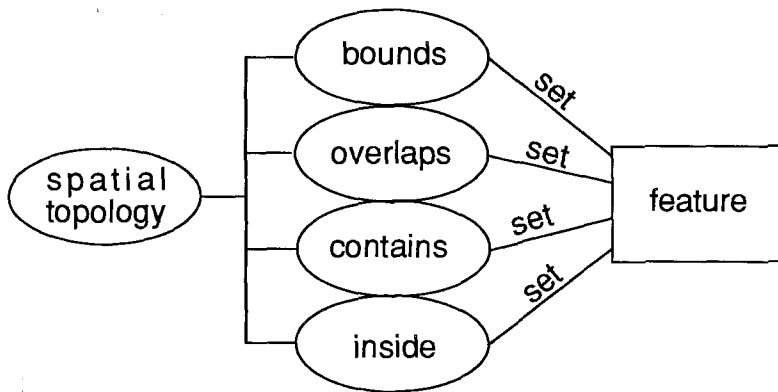
**Figure 5.23 Generalization Attribute Components**

The 'scale' of a feature defines the accuracy, resolution, and history of the other attributes of the feature, as well as the level of analysis or display appropriate for the feature.



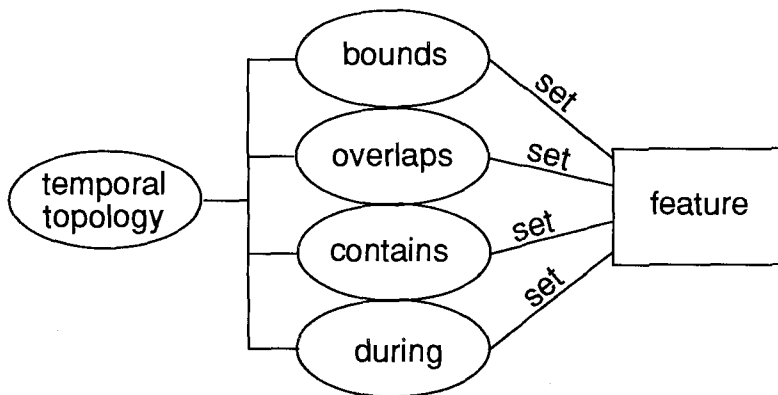
**Figure 5.24 Map Scale Attribute Types**

A map scale may be expressed as a range of representative fractions or simply qualitatively, such as 'medium' or 'engineering.'



**Figure 5.25 Spatial Topology Attribute Components**

A feature may spatially bound, overlap, contain, or lie inside any other number of features. (These would be in addition to the spatial components of a compound feature.)



**Figure 5.26 Temporal Topology Attribute Components**

A feature may temporally bound, overlap, contain, or occur during any other number of features. (These would be in addition to the temporal components of a compound feature.)

## Chapter Six

# Discussion

The purpose of this chapter is to compare the proposed geographic semantic database model (GSDM) with those models reviewed in Chapter 2. The main differences between my model and the general database and geographic data models are given below. Also, a point-by-point comparison of the models is provided in the tables. Because each model is described using rather different terminology and to varying degrees of detail, it is difficult to provide a rigorous analysis, so the tables only represent a broad summary. Note that the models with which mine are compared represent a sample of those in existence and there may be individual models not mentioned that compare more favorably in certain respects.

### 6.1 GSDM v. General Database Models

The proposed model largely incorporates general high-level database model principles. However, the GSDM is specifically designed for geographic applications, rather than general applications. That is the main difference between it and the first group of models, i.e., those reviewed in §2.1. In terms of level-of-abstraction, away from the computer and toward the real world, I would place the relational model [Bradley 1981; Brodie 1986; Codd 1970; Date 1985; Korth & Silberschatz 1991; Martin 1976; Ullman 1988] at a moderate level; the functional model [King & McLeod 1985; Norris-Sherborn 1984; Shipman 1981], the entity-relationship (E-R) model [Chen 1977; 1980; 1983; 1985], and the Hypergraph-based Data Structure (HBDS) [Bouillé 1978; 1984; 1986; Bouillé & Rugg 1983; Satharanond 1981] at a high level; while PROBE [Dayal & Smith 1986; Orenstein 1986; Orenstein & Manola 1988] and OSAM\* (Object-oriented Semantic Association Model) [Su 1986; Su *et al.* 1988; Alashqur *et al.* 1988] are at the

same high level as mine. The other big difference is the degree to which the models have been formally defined and implemented as a DBMS. The relational and functional models, being based almost entirely on mathematical principles, are very formally defined, and the relational DBMS is perhaps the most common kind in existence. HBDS, PROBE, and OSAM\* are all fairly formally defined and have been implemented, at least experimentally. My model, while certainly not as widely used, shares the E-R model's less formal nature.

That none of the general database models is specifically geographic is why none of them explicitly recognizes the spatial or temporal characterization domains present in my model. (PROBE does, however, *allow* such domains to be defined. ) Also, none of the general database models explicitly recognizes the scale and realization domains present in my model. Thus none of the geographic entities, the spatial, temporal, and generalization attributes, the topological and abstraction relationships, and the spatial and temporal operations of the GSDM, is explicitly included in any of the general models. Since I tend to equate topical data with general data, all models are deemed to support the topical domain.

The level-of-abstraction issue does not separate the GSDM from the other data models quite as much as purpose does. With the exception of the relational model, all models appear to support abstract entities, i.e., entities that do not require user-defined IDs. While I'm uncertain about HBDS and PROBE, the only model not allowing attributes to have multiple values is again the relational one. As far as supporting dependent, or computed attributes is concerned, the relational and E-R models appear to be the only ones not doing so. (Extended relational models, however, do support them.) Although most of the general database models allow sub-typing of entities, only OSAM\* and my model allow multi-parent relationships, i.e., multiple super-typing of entities. While all models allow the description of entities by attributes, only OSAM\* and GSDM explicitly support the composition of entities by components. In terms of database dynamics, the E-R model is the only one without such a component.

The GSDM builds, mostly in terms of geographic application, on the capabilities of

other database models, especially OSAM\*. Because it is new, its description is less formal and extensive than the other models, and it has not yet been implemented. In fact, the GSDM could possibly be implemented *using* one of the other database models.

## 6.2 GSDM v. Geographic Data Models

Not all the geographic data models reviewed in §2.2 were designed as database models. Goodchild's model [Goodchild 1987a; 1987b] primarily concerns geo-data analysis; ATKIS (Authoritative Topographic-Cartographic Information System) [Hesse & Leahy 1991] and SAIF (Spatial Archive and Interchange Format) [SAIF 1991a; 1991b; Sondheim 1991] are mainly data transfer models; while the KBGIS (Knowledge-based Geographic Information System) model [Menon & Smith 1989; Peuquet 1984; Smith & Pazner 1984] and Alves' model [Alves 1990] are deemed general-purpose GIS models. Only Nyerges' model [Nyerges 1980] and the GSDM are primarily database models. As for formal definition and implementation, Nyerges' model, KBGIS, ATKIS, and SAIF have been, at least experimentally so, while implementation of Goodchild's, Alves', and my models remains to be done.

None of the other models, except for SAIF, explicitly accommodates the temporal characterization domain to the extent that mine does. That is, they do not directly have temporal attributes or relationships. Also, not all of the geographic data models accommodate the three abstraction domains. KBGIS does not provide for different realizations (i.e., graphic and analytic), SAIF does not provide for multiple scales of representation, and Goodchild's model does not support multiple scales or realizations.

Although geographic entities such as features and layers are provided in all models, neither the profile nor the composite, present in my model, are available in any other model. Most of the geographic data models do not appear to allow the dependent, or computed, attribute type present in my model. With the exception of KBGIS, no model other than mine allows multi-parent relationships. All models have the described-by relationship and some topologic relationships.



While all of the models support operations on geo-data, few of them have operations encapsulated within object definitions.

To conclude, the GSDM has a more comprehensive view of geographical information than does any of the other geo-data models covered here. However, it must be repeated that some of the models are more fully defined than mine.

**Table 6.1 General Database Models: a comparison of other models with the GSDM**

Model	Relational	Functional	E-R	HBDS	PROBE	OSAM*	GSDM
<b>Level</b>	medium	high	high	high	very high	very high	very high
<b>Purpose</b>	general	general	general	general	general	general	geography
<b>Formal</b>	yes	yes	no	yes	yes	yes	no
<b>Implemented</b>	yes, much	yes	no	yes	yes	yes	no
<b>Characterization Domains</b>							
topical	yes	yes	yes	yes	yes	yes	yes
spatial	no	no	no	no	yes~	no	yes
temporal	no	no	no	no	yes~	no	yes
<b>Abstraction Domains</b>							
scale	no	no	no	no	no	no	yes
realization	no	no	no	no	no	no	yes
construction	yes	yes	yes	yes	yes	yes	yes
<b>Entities</b>							
general	yes	yes	yes	yes	yes	yes	no
geographic	no	no	no	no	no	no	yes
abstract	no	yes	yes	yes	?	yes	yes
<b>Attributes</b>							
general	yes	yes	yes	yes	yes	yes	yes
geographic	no	no	no	no	no	no	yes
multi-valued	no	yes	yes	?	?	yes	yes
dependent	no"	yes	no	yes	?	yes	yes
<b>Relationships</b>							
general	yes	yes	yes	yes	yes	yes	no
has-subtype	no'	yes	no'	yes	yes	yes	yes
multi-parent	no	no	no	no	no	yes	yes
described-by	yes	yes	yes	yes	yes	yes	yes
composed-of	no=	no=	no=	no=	no=	yes	yes
topologic	no	no	no	no	no	no	yes
abstraction	no#	no#	no#	no#	no#	no#	yes
<b>Operations</b>							
generic	yes	yes	no	yes	yes	yes	yes
geographic	no	no	no	no	no	no	yes
encapsulated	no	no	no	?	?	yes	yes

~ PROBE has no spatio-temporal domains explicitly, but allows them to be defined.

' Extensions of the relational and E-R models allow subtypes.

" Extensions of the relational model allows dependent attributes.

= However, composed-of can be defined using general relationships.

# However, implemented-as is inherent in all models.

**Table 6.2 Geographic Data Models: a comparison of other models with the GSDM**

<b>Model</b>	<b>Nyerges</b>	<b>Goodchild</b>	<b>KBGIS</b>	<b>ATKIS</b>	<b>Alves</b>	<b>SAIF</b>	<b>GSDM</b>
<b>Purpose</b>	database	analysis	general	transfer	general	transfer	database
<b>Formal</b>	yes	no	yes	yes	no	yes	no
<b>Implemented</b>	yes	no	yes	yes	no	yes	no
<b>Characterization Domains</b>							
topical	yes	yes	yes	yes	yes	yes	yes
spatial	yes	yes	yes	yes	yes	yes	yes
temporal	no	no	no	no	no	yes	yes
<b>Abstraction Domains</b>							
scale	yes	no	yes	yes	yes	no	yes
realization	yes	no	no	yes	yes	yes	yes
construction	yes	yes	yes	yes	yes	yes	yes
<b>Entities</b>							
feature	yes	yes	yes	yes	yes	yes	yes
profile	no	no	no	no	no	no	yes
layer	yes	yes	yes	yes	yes	yes	yes
composite	no	?	no	no	no	no	yes
<b>Attributes</b>							
taxonomy	yes	yes	yes	yes	yes	yes	yes
geometry	yes	yes	yes	yes	yes	yes	yes
chronology	no	no	no	no	no	yes	yes
multi-valued	?	yes	?	?	yes	yes	yes
dependent	no	no	?	no	?	no	yes
<b>Relationships</b>							
has-subtype	no	no	yes	yes	yes	yes	yes
multi-parent	no	no	yes	no	no	no	yes
described-by	yes	yes	yes	yes	yes	yes	yes
composed-of	no	yes	no	no	no	yes	yes
topologic	yes	yes	yes	yes	yes	yes	yes
abstraction	yes	no	some	yes	yes	yes	yes
<b>Operations</b>							
generic	yes	yes	yes	yes	yes	yes	yes
geographic	yes	yes	yes	yes	yes	yes	yes
encapsulated	no	no	no	no	yes	yes	yes

## Chapter Seven

# Conclusion

### 7.1 Summary of Research

The purpose of the thesis was to propose an outline of a geographic semantic database model — a conceptual framework for the design, construction, and use of geographic databases. The work involved producing syntheses of both general semantic database concepts and specific geographic information concepts and combining them into a comprehensive, geographic semantic database model (GSDM). It was assumed that semantic databases are more appropriate for building sophisticated information systems than are syntactic ones, and that geographic information has special characteristics and is more complex than ordinary information.

A review of related models was produced (§2). It included a few general database models that vary in their level of abstraction and degree of implementation, and a few geographic data models that vary in their intended purpose within a GIS.

A synthesis of logical database models was produced (§3). A logical database model incorporates notions of the structural and behavioral aspects of stored information. Structurally, a database contains entities, relations, and attributes. Behaviorally, a database has queries and transactions. Database models are evolving from syntactic to semantic forms, representing greater ability to directly and easily model reality.

A synthesis of geographic data processing concepts was also produced (§4). Any things or events of interest in geographic data processing can be called geographic phenomena. A geographic phenomenon is considered to exhibit three primary kinds of

characteristics: topical, spatial, and temporal. That is, it has some identification and classification, some position and extent, and happens or exists at some time. Information on such phenomena thus exists within what I call characterization domains. I also consider it to exist within three abstraction domains: generalization, realization, and construction. That is, geographic phenomena or information are considered at some degree of accuracy and resolution, take on some form between reality and concept, and have a certain level of meaning or applicability. From a database point-of-view, the characterization domains and abstraction domains are the particularly geographic ways for logically partitioning a collection of geographic data.

The above concepts were combined into an informal conceptual database model for geographic information systems (§5). The proposed model, the GSDM, contains entities, such as features, profiles, layers, and composites, which represent geographic phenomena. The entities are characterized by topical, spatial, temporal, and scale attributes, and by semantic, topologic, and abstraction relations to other entities. The entities may be accessed, processed, and updated by database manipulation operations involving selections and actions. The entities also exist at different levels of abstraction. They exist at different scales, appropriate for different levels of investigation. They are in analytic or graphic form, depending on whether they are to be used for machine or visual processing. And they exist as applied, basic, or primal constructs, appropriate for different kinds of database user.

The GSDM was compared with the other models reviewed (§6). Not surprisingly, it was found to be more directly suitable to geography than were the general-purpose database models. It was also found to be more comprehensive in scope than the geographic data models.

## **7.2 Implications of Proposed Model**

Three possible categories of use to which the proposed model may be put are envisaged. The GSDM may be used as a geographic database specification, analysis, and design

tool; as the basis of an actual geographic DBMS; or as a broader geographic conceptual modelling tool.

### **7.2.1 Geo-database Specification, Analysis, and Design**

Geographic database designers can use the GSDM as a tool for the specification, analysis, and design of their databases — even databases that are implemented according to a lower-level model. Since it is a high-level model, thoughts and ideas about what pieces of geographic information and how they are related and processed can be translated easily and semi-formally onto paper, without being encumbered by the specifics of lower-level database models. Particular entities of interest to a specific application can be enumerated, their attributes and relations determined, and their basic geographic form defined. Once such a high-level database schema has been specified it can be translated into the form to be used in the final model, i.e., into a form understood by the computer system.

### **7.2.2 Geo-DBMS**

The proposed model can be used as the guiding framework for developing an actual geographic DBMS. This would of course require a considerable implementation effort. Such a DBMS would not only allow the specification, analysis, and design of geographic databases, but also their implementation and use, without translation into another model's form.

### **7.2.3 Geo-Conceptual Modelling**

The GSDM could be used not only as a database model, for storage and retrieval of geographic information, but as a more general theory for geographic information processing, including geo-data manipulation, analysis, visualization, and so on. This would require considerable expansion of the model in new directions, particularly manipulation and analysis. In such a broader situation the distinction between the temporary holding of data in memory during processing and the persistent storage of

data on disk between processing sessions is ignored. Instead, components of the model will directly represent all kinds of geographic phenomena — their states, interrelations and processes — not just those stored and retrieved in databases.

## **7.3 Possible Future Projects**

Four broad categories of possible future research areas related to this project are envisaged. The model could be developed further, it could be tested, it could be further compared with other models, or it could be implemented.

### **7.3.1 Model Development**

There is always room for further development of any proposed model. The GSDM may be specified in a more formal manner, or it may be expanded to give it more breadth or depth.

Formalizing the model could be done in a number of ways:

- on its own terms, independently of other database models;
- using another, existing, formal database model, e.g., OSAM\*; or
- mathematically, using a rather abstract but rigorously analytical language.

Expanding the model's scope could be done so as to include a number of new things:

- More detailed or rigorous coverage could be given of certain existing aspects, such as topical taxonomies, 3D entities and relations, logical and physical time, uncertainty management, scale attributes and relations, graphic map entities, primal geometric objects, and encapsulation of operations.
- General database objects could be included, additional to the geographic ones, to give more flexibility.
- New logical database aspects that have not been covered, such as expert rules, and sound and motion objects, could be investigated.
- Other geo-processing activities, such as data capture, statistical and spatial analysis, map display, dynamic geographic phenomena modelling and

prediction, and so on, could be brought under the models realm. This would be necessary for geographic conceptual modelling.

- The model could be interfaced with physical database models or could be expanded to incorporate things such as indexing or hashing, proximity storage, transaction logging, and data distribution.

During the writing of this thesis, there have been a number of developments on or related to geographic data modelling. Further work on my model could take recent literature on the temporal domain [Frank *et al.* 1992; Hazelton 1990; Langran 1992; Snodgrass 1990], the 3D spatial domain [Raper 1989], the generalization domain [Buttenfield & McMaster 1991; Goodchild & Gopal 1989; Guptill 1990; Medyckyj-Scott *et al.* 1991], geo-data transfer standards [Latour 1991; Moellering 1991], geodatabases [Buchmann *et al.* 1989; Gambosi *et al.* 1992; Günther 1988; Günther & Schek 1991; Ooi 1990; van Oosterom 1990], and object-oriented modelling [Gray *et al.* 1992; Hughes 1991; Khoshafian & Abnous 1990] into consideration.

### **7.3.2 Model Testing**

One way of testing the model, short of implementation, would be to set up a hypothetical, or actual, application problem — topographic or cadastral, for example. An appropriate conceptual database schema could be built and various queries and transactions could be composed, to see how well the model copes.

### **7.3.3 Model Comparison**

In order that the proposed model be understood by a wider audience it could be thoroughly compared with other models. A more detailed comparison with the other models with which it has already been compared could be made. Or, it could be compared with those models corresponding to existing GIS packages, such as Terrasoft or Arc/Info.



### 7.3.4 Model Implementation

The proposed model could be implemented as a prototype package with one of two objectives: build a geographic database design tool, or build an actual geographic DBMS. A number of possible tools could be used to build such prototypes:

- an object-oriented programming language, e.g., Smalltalk or C++,
- an object-oriented DBMS, e.g., Gemstone,
- a logic programming language, e.g., Prolog,
- a spatial DBMS, e.g., Spatial II or SpatialDBMS,
- a conventional DBMS, e.g., Oracle or dBase,
- an existing GIS package, e.g., Terrasoft or Arc/Info,
- an expert system shell, e.g., VP-expert, and
- a hypermedia system, e.g., Hypercard.

Any number of the above four projects could be combined into an iterative research and development project, with each phase being in a more extensive form than its earlier incarnation.

---

The current research represents a step towards an overall theoretical understanding of what and how geographic information can be effectively managed within the GIS of the future.

## Bibliography

- Abel, D.J. 1988. "Relational Data Management Facilities for Spatial Information Systems." *Proceedings, Third International Symposium on Spatial Data Handling, Sydney, Australia, Aug. 1988*. Columbus, OH: International Geographical Union, Commission on Geographical Data Sensing and Processing, and Department of Geography, Ohio State University, 9-18.
- Abel, D.J. & J.L. Smith. 1986. "A Relational GIS Database Accommodating Independent Partitionings of the Region." *Proceedings, Second International Symposium on Spatial Data Handling, Seattle, WA, July 1986*. Williamsville, NY: International Geographical Union, Commission on Geographical Data Sensing and Processing, 213-224.
- Aho, A.V., J.E. Hopcroft & J.D. Ullman. 1982. *Data Structures and Algorithms*. Reading, MA: Addison-Wesley.
- Alashqur, A.M., S.Y.W. Su, H. Lam, M.S. Guo & E. Barkmeyer. 1988. "OQL - An Object-oriented Query Language." Unpub. paper, Database Systems Research & Development Center, University of Florida, Gainesville, FL, 25pp.
- Alves, D. 1990. "A Data Model for Geographic Information Systems." *Proceedings, Fourth International Symposium on Spatial Data Handling, Zürich, July 1990*. Columbus, OH: International Geographical Union, Commission on GIS, vol. 2, 879-887.
- Aranoff, S. 1989. *Geographic Information Systems: A Management Perspective*. Ottawa: WDL Publications.
- Aronson, P. 1989. "The Geographic Database - Logically Continuous and Physically Discrete." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 452-461.
- Beatty. 1980. *Geographic Referencing*. Report for Land-Related Information Systems Coordination Project, Alberta Treasury, Edmonton, by J.C. Beatty Consulting Ltd., Spruce Grove, Alberta.

- Bédard, Y. & F. Paquette. 1989. "Extending Entity/Relationship Formalism for Spatial Information Systems." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 818-827.
- Berry, J. 1964. "Approaches to Regional Analysis: a Synthesis." *Annals, American Association of Geographers*, **54**, 2-12.
- Borgida, A. 1986. "Conceptual Modeling of Information Systems." *On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies*. M.L. Brodie & J. Mylopoulos, eds., New York: Springer-Verlag, 461-469.
- Bouillé, F. 1978. "Structuring Cartographic Data and Spatial Processes with the HBDS." *Topological Data Structures for Geographic Information Systems, Proceedings, First International Advanced Study Symposium*. G. Dutton, ed., Cambridge, MA: Harvard University, Laboratory for Computer Graphics and Spatial Analysis, vol. 5.
- Bouillé, F. 1984. "Architecture of a Geographic Structured Expert System." *Proceedings, International Symposium on Spatial Data Handling, Zürich, Aug. 1984*. Zürich: Geographisches Institut, vol. II, 520-543.
- Bouillé, F. 1986. "Interfacing Cartographic Knowledge Structures and Robotics." *Auto Carto London, Proceedings, International Conference on the Acquisition, Management and Presentation of Spatial Data, London, Sept. 1986*. M. Blakemore, ed., London: Auto Carto London Ltd., vol. 2, 563-571.
- Bouillé, F. & R.D. Rugg, eds. 1983. *Proceedings, Second International Seminar on Hypergraph-Based Data Structures (HBDS), Richmond, VA, Mar. 1981*. Richmond, VA: Virginia Commonwealth University.
- Bowyer, A. & J. Woodwark. 1983. *A Programmer's Geometry*. London: Butterworths.
- Brodie, M.L. & J. Mylopoulos, eds. 1986. *On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies*. New York: Springer-Verlag.
- Bradley, J. 1981. *File and Data Base Techniques*. New York: Holt, Rinehart & Winston.
- Brodie, M.L. 1984. "On the Development of Data Models." *On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming*

- Languages*. M.L. Brodie, J. Mylopoulos & J.W. Schmidt, eds., New York: Springer-Verlag.
- Brodie, M.L. 1986. "Database Management: A Survey." *On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies*. M.L. Brodie & J. Mylopoulos, eds., New York: Springer-Verlag, 201-218.
- Brodie, M.L., J. Mylopoulos, & J.W. Schmidt, eds. 1984. *On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages*. New York: Springer-Verlag.
- Bruegger, B.P. & A.U. Frank. 1989. "Hierarchies Over Topological Data Structures." *Technical Papers, ASPRS/ACSM Annual Convention, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Society of Photogrammetry and Remote Sensing-American Congress on Surveying and Mapping, vol. 4, 137-145.
- Buchmann, A., O. Günther, T.R. Smith & Y.F Wang, eds. 1989. *SSD'89, Design and Implementation of Large Spatial Databases, Proceedings, First Symposium of Large Spatial Databases*. Berlin: Springer-Verlag.
- Burrough, P.A. 1986. *Principles of Geographical Information Systems*. Oxford: Clarendon Press.
- Burton, W. 1979. "Logical and Physical Data Types in Geographic Information Systems." *Geo-Processing*, Dec., 1:2, 167-181.
- Buttenfield, B.P. & R.B. McMaster. 1991. *Map Generalization: Making Rules for Knowledge Representation*. Wiley.
- Buyong, T.B. & A.U. Frank. 1989. "Measurement-Based MultiPurpose Cadastre." *Technical Papers, ASPRS/ACSM Annual Convention, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Society of Photogrammetry and Remote Sensing-American Congress on Surveying and Mapping, vol. 5, 58-66.
- Canadian Council on Surveying & Mapping. 1984. *National Standards for the Exchange of Digital Topographic Data, Volume I, Data Classification, Quality Evaluation and EDP File Format*. 2nd draft report, Ottawa: Surveys and Mapping Branch, Energy, Mines & Resources Canada.
- Chen, P. 1977. *The Entity-Relationship Approach to Logical Data Base Design*, No. 6 in QED Monograph Series on Data Base Management. Wellesley, MA: QED Information Sciences.

- Chen, P., ed. 1980. *Proceedings, International Conference on Entity-Relationship Approach to Systems Analysis and Design, Los Angeles, Dec 1979*. Amsterdam: North-Holland.
- Chen, P., ed. 1983. *Proceedings, Second International Conference on Entity-Relationship Approach to Information Modeling and Analysis, Washington, DC, Oct 1981*. Amsterdam: North-Holland.
- Chen, P.P.S. 1985. "Database Design Based on Entity and Relationship." *Principles of Database Design, Volume I, Logical Organizations*. S.B. Yao, ed., Englewood-Cliffs, NJ: Prentice-Hall, 174-210.
- Chrisman, N.R. 1981. *Methods of Spatial Analysis Based on Maps of Categorical Coverages*. Ph.D. Thesis, University of Bristol, England.
- Chrisman, N.R. 1989. "Error in Categorical Maps: Testing Versus Simulation." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 521-529.
- Claire, R.W. & S.C. Gupta. 1982. "Spatial Operators for Selected Data Structures." *Proceedings, Auto-Carto V/Commission IV ISPRS Symposium, Arlington, VA, Aug. 1982*, 189-200.
- Clarke, K.C. 1990. *Analytical and Computer Cartography*. Englewood Cliffs, NJ: Prentice Hall.
- Codd, E.F. 1970. "A Relational Model for Large Shared Data Banks." *Communications, Association for Computing Machinery*, June, 13:6, 377-387.
- Codd, E.F. 1979. "Extending the Database Relational Model to Capture More Meaning." *Association for Computing Machinery, Transactions on Database Systems*, Dec., 4:4, 397-434.
- Cook, B.G. 1978. "The Structural and Algorithmic Basis of a Geographic Data Base." *Topological Data Structures for Geographic Information Systems, Proceedings, First International Advanced Study Symposium*. G. Dutton, ed., Cambridge, MA: Harvard University, Laboratory for Computer Graphics and Spatial Analysis, vol. 4.
- Cox, N.J., B.K. Aldred & D.W. Rhind. 1980. "A Relational Database System and a Proposal for a Geographical Data Type." *Geo-Processing*, Nov., 1:3, 217-230.

- Dangermond, J. 1982. "A Classification of Software Components Commonly Used in Geographic Information Systems." *Proceedings, United States / Australia Workshop on Design and Implementation of Computer-Based Geographic Information Systems, Honolulu, Hawaii, 1982*. D. Peuquet & J. O'Callaghan, eds., Amherst, New York: International Geographical Union, Commission on Geographical Data Sensing and Processing, 70-91.
- Date, C.J. 1985. *An Introduction to Database Systems, Volume I*. 4th ed., Reading, MA: Addison-Wesley.
- Davis, W.A. & C.H. Hwang. 1986. "Organizing and Indexing of Spatial Data." *Proceedings, Second International Symposium on Spatial Data Handling, Seattle, WA, July 1986*. Williamsville, NY: International Geographical Union, Commission on Geographical Data Sensing and Processing, 5-14.
- Dayal, U. & J.M. Smith. 1986. "PROBE: A Knowledge-Oriented Database Management System." *On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies*. M.L. Brodie & J. Mylopoulos, eds., New York: Springer-Verlag, 227-257.
- Dent, B.D. 1990. *Cartography: Thematic Map Design*. 2nd ed., Dubuque, IA: W.C. Brown.
- Digital Cartographic Data Standards Task Force. 1988. "The Proposed Standard for Digital Cartographic Data." *American Cartographer*, Jan., 15:1, 11-140.
- Dueker, K.J. & D. Kjerne. 1989. "Multipurpose Cadastre: Terms and Definitions." *Technical Papers, ASPRS/ACSM Annual Convention, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Society of Photogrammetry and Remote Sensing-American Congress on Surveying and Mapping, vol. 5, 94-103.
- Dueker, K.J. 1988. "Urban Applications of Geographic Information Systems Technology: A Grouping Into Three Levels of Resolution." *URISA'88, Mapping the Future, Proceedings, 26th Annual Conference, Los Angeles, CA, Aug. 1988*. Washington, DC: Urban and Regional Information Systems Association, vol. II, 104-109.
- Ebdon, D. 1985. *Statistics in Geography*. 2nd ed., Oxford: Basil Blackwell.
- Edson, D. 1975. "Digital Cartographic Data Base Preliminary Description." *Auto-Carto 2, Proceedings, Second International Symposium on Computer-Assisted Cartography, Sep. 1975*. J. Kavaliunas & F. Broome, eds., Washington, DC: Department of Commerce & Falls Church, VA: American Congress on Surveying and Mapping, 523-538.

- Egonhofer, M.J. & A.U. Frank. 1989. "Object-Oriented Modeling in GIS: Inheritance and Propagation." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 588-598.
- Feuchtwanger, M. 1985. *An Investigation of Efficient Computer Techniques for the Storage and Retrieval of Land-related Information*. M.Sc. Thesis, Division of Surveying Engineering, University of Calgary, Alta., pub. no. 20010.
- Feuchtwanger, M. 1989a. "Geographic Logical Database Model Requirements." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 599-609.
- Feuchtwanger, M. 1989b. "An Object-oriented Semantic Association Model for GIS." *Challenge for the 1990s, Proceedings, National Conference on Geographic Information Systems, Ottawa, Feb.-Mar.* Ottawa: Canadian Institute of Surveying and Mapping, 689-703.
- Feuchtwanger, M. & T.K. Poiker. 1987. "Infological and Datalogical Spatial Models." *Lecture Notes in Digital Mapping and Land Information*. J.A.R. Blais, ed., Ottawa: Canadian Institute of Surveying and Mapping, 146-167.
- Frank, A.U. 1984. "Requirements for Database Systems Suitable to Manage Large Spatial Databases." *Proceedings, International Symposium on Spatial Data Handling, Zürich, Aug. 1984*. Zürich: Geographisches Institut, vol. I, 38-60.
- Frank, A.U. 1988. "Requirements for a Database Management System for a GIS." *Photogrammetric Engineering and Remote Sensing*, Nov., **54**:11, 1557-1564.
- Frank, A.U., I. Campari & U. Formentini, eds. 1992. *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space, Proceedings, International Conference, Pisa, Italy, Sep. 1992*. Lecture Notes in Computer Science, vol. 639, Berlin: Springer-Verlag.
- Gahegan, M.N. & S.A. Roberts. 1988. "An Intelligent, Object-oriented Geographical Information System." *International Journal of Geographical Information Systems*, Apr.-June, **2**:2, 101-110.
- Gambosi, G., M. Scholl & H.W. Six, eds. 1992. *Geographic Database Management Systems, Workshop Proceedings, Capri, Italy, May 1991*. Berlin: Springer-Verlag.

- Glen, R. 1992. "The Great New Database Debate." *Canadian Datasystems*, Apr., 23-26.
- Gold, C.M. 1988. "PAN Graphs. An Aid to GIS Analysis." *International Journal of Geographical Information Systems*, Jan.-Mar., 2:1, 29-42.
- Gold, C.M. 1989. "Spatial Adjacency - A General Approach." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 298-312.
- Goodchild, M.F. 1987a. "A Spatial Analytical Perspective on Geographical Information Systems." *International Journal of Geographical Information Systems*, Oct.-Dec., 1:4, 327-334.
- Goodchild, M.F. 1987b. "Towards an Enumeration and Classification of GIS Functions." *Proceedings, International GIS Symposium, Arlington, VA, Nov. 1987*, vol. 2, 67-77.
- Goodchild, M.F. & S. Gopal, eds. 1989. *The Accuracy of Spatial Databases*. London: Taylor & Francis.
- Gray, P.M.D., K.G. Kulkarni & N.W. Paton. 1992. *Object-Oriented Databases: A Semantic Data Model Approach*. New York: Prentice-Hall.
- Guptill, S.C. 1990. "Multiple Representations of Geographic Entities Through Space and Time." *Proceedings, Fourth International Symposium on Spatial Data Handling, Zürich, July 1990*. Columbus, OH: International Geographical Union, Commission on GIS, vol. 2, 859-868.
- Günther, O. 1988. *Efficient structures for Geometric Data Management*. Lecture Notes in Computer Science, vol. 337, Berlin: Springer-Verlag.
- Günther, O. & H.J. Schek, eds. 1991. *Advances in Spatial Databases, Proceedings, 2nd Symposium of Large Spatial Databases, Zürich, Aug. 1991*, Lecture Notes in Computer Science, vol. 525, Berlin: Springer-Verlag.
- Hagan, P.J. 1981. "The Evaluation of a Network Data Structure for Cartographic Features." *Technical Papers, ACSM Fall Technical Meeting, San Francisco & Honolulu, Sept. 1981*. Falls Church, VA: American Congress on Surveying and Mapping, 16-27.



- Hammer, M.M. & D.J. McLeod. 1975. "Semantic Integrity in a Relational Database System." *Proceedings, International Conference, Very Large Data Bases, Framingham, MA, Sep. 1975.*
- Hammond, R. & P.S. McCullagh. 1978. *Quantitative Techniques in Geography, an Introduction.* 2nd ed., Oxford: Clarendon Press.
- Haralick, R.M. 1980. "A Spatial Data Structure for Geographic Information Systems." *Map Data Processing.* Freeman & Pieroni, eds., New York: Academic Press.
- Hazelton, B. 1990. "On the Design of Temporally-Referenced 3-D GIS: Towards 4-D GIS." *GIS/LIS'90 Proceedings.* Falls Church, VA: American Society of Photogrammetry and Remote Sensing.
- Hesse, W. & F.J. Leahy. 1991. "Authoritative Topographic Cartographic Information System (ATKIS)." Unpub. paper, Department of Surveying and Land Information, University of Melbourne, Australia.
- Hevner, A.R. & S.B. Yao. 1985. "Network Database Design Methods." *Principles of Database Design, Volume I, Logical Organizations.* S.B. Yao, ed., Englewood-Cliffs, NJ: Prentice-Hall, 294-324.
- Hickin, E.J. 1983. "River Channel Changes: Retrospect and Prospect." *Special Pubs: International Association of Sedimentologists*, 6, 61-83.
- Hughes, J.G. 1991. *Object-Oriented Databases.* New York: Prentice-Hall.
- Jackson, J.E. 1980. *Sphere, Spheroid and Projections for Surveyors.* New York: Halstead Press.
- Jones, C.B. & I.M. Abraham. 1987. "Line Generalization in a Global Cartographic Database." *Cartographica*, Autumn, 24:3,32-45.
- Kainz, W. 1988. "Application of Lattice Theory to Geography." *Proceedings, Third International Symposium on Spatial Data Handling, Sydney, Australia, Aug. 1988.* Columbus, OH: International Geographical Union, Commission on Geographical Data Sensing and Processing, and Department of Geography, Ohio State University, 135-142.
- Khoshafian, S. & R. Abnous. 1990. *Object Orientation: Concepts, Languages, Databases, User Interfaces.* New York: Wiley.

- King, R. & D. McLeod. 1985. "Semantic Data Models." *Principles of Database Design, Volume I, Logical Organizations*. S.B. Yao, ed., Englewood-Cliffs, NJ: Prentice-Hall, 115-150.
- Kleiner, A. 1989. "Storage Methods for Fast Access to Large Cartographic Data Collections - An Empirical Study." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 416-425.
- Koffman, E.B. 1988. *Problem Solving and Structured Programming in Modula-2*. Reading, MA: Addison-Wesley.
- Korth, H.F. & A. Silberschatz. 1991. *Database System Concepts*. 2nd ed., New York: McGraw-Hill.
- Langran, G. 1989. "Accessing SpatioTemporal Data in a Temporal GIS." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 191-198.
- Langran, G. 1992. *Time in Geographic Information Systems*. London: Taylor & Francis.
- Langran, G. & N.R. Chrisman. 1988. "A Framework for Temporal Geographic Information." *Cartographica*, Winter, 25:3, 1-14.
- Latour, S. 1991. *Evaluation of Interchange Standards Technical Reports*. Final draft, Technical report, Canadian General Standard Board Committee on Geomatics, Canada Centre for Geomatics, EMR.
- Laurini, R. & F. Milleret. 1989. "Solving Spatial Queries by Relational Algebra." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 426-435.
- Lillesand, T.M. & R.W. Kiefer. 1987. *Remote Sensing and Image Interpretation*. 2nd ed., New York: Wiley.
- Loitsch, J. 1991. "A Relational Database Management System with Topological Elements and Topological Operators." Unpub. paper.

- Lorie, R.A. & A. Meier. 1984 "Using a Relational DBMS for Geographic Information Systems." *Geo-Processing*, Oct., 2:3, 243-258.
- Lukatela, H. 1989. "GIS Future: Automated Cartography or Geo-Relational Solid Modelling." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 341-347.
- Maguire, D.J., M.F. Goodchild & D.W. Rhind, eds. 1991. *Geographical Information Systems: Principles and Applications*. New York: Wiley.
- Maling, D.H. 1992. *Coordinate Systems and Map Projections*. 2nd ed., Oxford: Pergamon Press.
- Mark, D.M. & J.P. Lauzon. 1984. "Linear Quadrees for Geographic Information Systems." *Proceedings, International Symposium on Spatial Data Handling, Zürich, Aug. 1984*. Zürich: Geographisches Institut, 412-431.
- Martin, J. 1976. *Principles of Data-Base Management*. Englewood-Cliffs, New Jersey: Prentice-Hall.
- Matsuyama, T., L.V. Hao & M. Nagao. 1984. "A File Organization for Geographic Information Systems Based on Spatial Proximity." *Computer Vision, Graphics and Image Processing*, 26, 303-318.
- Medyckyj-Scott, D., I. Newman & D. Walker, eds. 1991. *Metadata in the Geosciences*. Loughborough, England: Group D.
- Menon, S. & T.R. Smith. 1989. "A Declarative Spatial Query Processor for Geographic Information Systems." *Photogrammetric Engineering and Remote Sensing*, Nov. 55:11, 1593-1600.
- Mikhail, E.M. 1976. *Observations and Least Squares*. New York: Harper & Row.
- Miller, R., H.A. Karimi & M. Feuchtwanger. 1989. "Uncertainty and its Management in Geographic Information Systems." *Challenge for the 1990s, Proceedings, National Conference on Geographic Information Systems, Ottawa, Feb.-Mar. 1989*. Ottawa: Canadian Institute of Surveying and Mapping, 252-260.
- Moellering, H. 1980. "Strategies of Real-Time Cartography." *Cartographic Journal*, 17:1, 12-15.

- Moellering, H. ed. 1991. *Spatial Database Transfer Standards: Current International Status*. London: Elsevier Applied Science.
- Morehouse, S. 1985. "Arc/Info - A Geo-Relational Model for Spatial Information." *Auto-Carto 7, Digital Representations of Spatial Knowledge, Proceedings, Seventh International Symposium on Computer-Assisted Cartography, Washington, DC, Mar. 1985*. Falls Church, VA: ASP & American Congress on Surveying and Mapping, 388-397.
- Muehrcke, P.C. 1990. "Cartography and Geographic Information Systems." *Cartography and Geographic Information Systems*, Jan., 17:1, 7-15.
- NCGIA. 1989. "The Research Plan of the National Centre for Geographic Information and Analysis." *International Journal of Geographical Information Systems*, Apr.-June, 3:2, 117-136.
- Norris-Sherborn, A. 1984. *A Geographic Data Modelling and Query Language Facility*. Ph.D. Thesis, School of Computing Science, University of East Anglia, England.
- Nyerges, T.L. 1980. *Modeling the Structure of Cartographic Information for Query Processing*. Ph.D. Thesis, Ohio State University, OH, Ann Arbor, MI: University Microfilms International.
- Nyerges, T.L. & C.S. Smyth. 1983. "IGDMS an Integrated Geographic Data Management System." *Auto-Carto 6, Sixth International Symposium on Automated Cartography, Ottawa, Oct. 1983*, 9pp.
- Ooi, B.C. 1990. *Efficient Query Processing in Geographic Information Systems*. Lecture Notes in Computer Science, vol. 471, Berlin: Springer-Verlag.
- Orenstein, J.A. 1986. "Spatial Query Processing in an Object-Oriented Database System." *Proceedings, International Conference on Management of Data*. New York: Association for Computing Machinery, Special Interest Group on Management of Data, 326-336.
- Orenstein, J.A. & F.A. Manola. 1988. "PROBE: Spatial Data Modeling and Query Processing in an Image Database Application." *IEEE Transactions on Software Engineering*, May, 14:5, 611-629.
- Ormeling, F.J. ed. 1984 *Basic Cartography for Students and Technicians, Volume 1*. London: Elsevier Applied Science.
- Peucker, T.K. & N.R. Chrisman. 1975. "Cartographic Data Structures." *American Cartographer*, 2:1, 56-69.

- Peucker, T.K., R.J. Fowler, J.J Little & D.M. Mark. 1976. *Digital Representation of Three-Dimensional Surfaces by Triangulated Irregular Networks (TIN)*. Technical Report 10, Department of Geography, Simon Fraser University, Burnaby, BC.
- Peuquet, D.J. 1978. "Raster Data Handling in Geographic Information Systems." *Topological Data Structures for Geographic Information Systems, Proceedings, First International Advanced Study Symposium*. G. Dutton, ed., Cambridge, MA: Harvard University, Laboratory for Computer Graphics and Spatial Analysis, vol. 2.
- Peuquet, D.J. 1984. "Data Structures for a Knowledge-based Geographic Information System." *Proceedings, International Symposium on Spatial Data Handling, Zürich, Aug. 1984*. Zürich: Geographisches Institut, vol. II, 372-391.
- Peuquet, D.J. 1988. "Toward the Definition and Use of Complex Spatial Relationships." *Proceedings, Third International Symposium on Spatial Data Handling, Sydney, Australia, Aug. 1988*. Columbus, OH: International Geographical Union, Commission on Geographical Data Sensing and Processing, and Department of Geography, Ohio State University, 211-223.
- Pullar, D. 1989. "Data Organization of a Digital Cadastral Data Base." *Technical Papers, ASPRS/ACSM Annual Convention, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Society of Photogrammetry and Remote Sensing-American Congress on Surveying and Mapping, vol. 5, 87-93.
- Pullar, D.V. & M.J. Egenhofer. 1988. "Toward Formal Definitions of Topological Relations Among Spatial Objects." *Proceedings, Third International Symposium on Spatial Data Handling, Sydney, Australia, Aug. 1988*. Columbus, OH: International Geographical Union, Commission on Geographical Data Sensing and Processing, and Department of Geography, Ohio State University, 225-241.
- Raper, J. ed. 1989. *Three Dimensional Applications in Geographical Information Systems*. London: Taylor & Francis, 1989.
- Rhind, D. 1988. "A GIS Research Agenda." *International Journal of Geographical Information Systems*, Jan.-Mar., 2:1, 23-28.
- Richardus, P. & R.K. Adler. 1972. *Map Projections*. New York: American Elsevier.
- Robinson, A.H., R.D. Sale, J.L. Morrison & P.C. Muehrcke. 1984. *Elements of Cartography*. 5th ed., New York: Wiley.

- Rybeck, T. 1990. "DBMS: Next wave could be an easier ride." *Computer World*, 5 Mar., 67-72.
- SAIF. 1991a. *Spatial Archive and Interchange Format: Overview of SAIF*. British Columbia Specifications and Guidelines for Geomatics, Reference Series, vol. 1, Surveys & Resource Mapping Branch, Ministry of Crown Lands, Victoria, BC.
- SAIF. 1991b. *Spatial Archive and Interchange Format: SAIF Interchange Specification*. British Columbia Specifications and Guidelines for Geomatics, Reference Series, vol. 2, Surveys & Resource Mapping Branch, Ministry of Crown Lands, Victoria, BC.
- Salgé, F. & M.N. Schlafer. 1989. "A Geographic Data Model Based on HBDS Concepts: The IGN Cartographic Data Base Model." *Auto Carto 9, Proceedings, Ninth International Symposium on Computer Assisted Cartography, Baltimore, MD, Apr. 1989*. Falls Church, VA: American Congress on Surveying and Mapping / American Society of Photogrammetry and Remote Sensing, 110-117.
- Samet, H. 1984. "The Quadtree and Related Hierarchical Data Structures." *Computing Surveys*, June, 16:2, 187-260.
- Satharanond, V. 1981. "An Application of the Hypergraph-Based Data Structure (HBDS) in an Agricultural Suitability Study." *Technical Papers, ACSM Fall Technical Meeting, San Francisco & Honolulu, Sept. 1981*. Falls Church, VA: American Congress on Surveying and Mapping, 29-38.
- Schek, H.J. & W. Waterfield. 1986. "A Database Kernel System for Geoscientific Applications." *Proceedings, Second International Symposium on Spatial Data Handling, Seattle, WA, July 1986*. Williamsville, NY: International Geographical Union, Commission on Geographical Data Sensing and Processing, 273-288.
- Shamos, M.I. 1978. *Computational Geometry*. Ph.D. Thesis, Yale University, New Haven, CT, Ann Arbor, MI: University Microfilms International.
- Shamos, M.I. & J.L. Bentley. 1978. "Optimal Algorithms for Structuring Geographic Data." G. Dutton, ed. *Topological Data Structures for Geographic Information Systems, Proceedings, First International Advanced Study Symposium*. Cambridge, MA: Harvard University, Laboratory for Computer Graphics and Spatial Analysis, vol. 6.
- Shapiro, L.G. & R.M. Haralick. 1980. "A Spatial Data Structure." *Geo-Processing*, Nov., 1:3, 313-338.

- Shaw, M. 1984. "Abstraction Techniques in Modern Programming Languages." *IEEE Software*.
- Shipman, D. 1981. "The Functional Data Model and the Data Language DAPLEX." *Association for Computing Machinery, Transactions on Database Systems*, Mar., 6:1.
- Smith, T.R. & M. Pazner. 1984. "Knowledge-based Control of Search and Learning in a Large-scale GIS." *Proceedings, International Symposium on Spatial Data Handling, Zürich, Aug. 1984*. Zürich: Geographisches Institut, vol. II, 498-519.
- Snodgrass, R. 1990. "Temporal Databases: Status and Research Directions." *SIGMOD RECORD*, Dec., 19:4, 83-89.
- Snyder, J.P. 1987. *Map Projections - A Working Manual*. USGS, Professional Paper 1395, Washington, DC: USGPO.
- Sondheim, M. 1991. "Canada Considers SAIF Specifications." *GIS World*, June, 4:4, 35-37.
- Star, J. & J. Estes. 1990. *Geographic Information Systems: An Introduction*. Englewood Cliffs, NJ: Prentice Hall.
- Su., S.Y.W. 1986. "Modeling Integrated Manufacturing Data with SAM\*." *Computer*, Jan., 19:1, 34-49.
- Su, S.Y.W., V. Krishnamurthy & H. Lam. 1988. "An Object-oriented Semantic Association Model (OSAM\*)." *AI in Industrial Engineering and Manufacturing: Theoretical Issues and Applications*. S. Kumara, R. Kashyap & A.L. Soyster, eds., American Institute of Industrial Engineers, 56pp.
- Tamminen, M. 1984. "Efficient Geometric Access to a Multirepresentational Geodatabase." *Geo-Processing*, Mar., 2:2, 177-196.
- Tomlin, C.D. 1983. *Digital Cartographic Modelling Techniques in Environmental Management*. Doctoral Dissertation, School of Forestry and Environmental Studies, Yale University, New Haven, CT.
- Tomlin, C.D. 1990. *Geographic Information Systems and Cartographic Modeling*. Englewood Cliffs, NJ: Prentice Hall.
- Tomlinson, R.F. & R. Boyle. 1981. "The State of Development of Systems for Handling Natural Resources Inventory Data." *Cartographica*, Winter, 18:4, 65-95.

- Tuori, M. & G. Moon. 1984. "A Topographic Map Conceptual Data Model." *Proceedings, International Symposium on Spatial Data Handling, Zürich, Aug. 1984*. Zürich: Geographisches Institut, vol. I, 28-37.
- Ullman, J.D. 1988. *Principles of Database and Knowledge-Base Systems, Volume 1*. Rockville, MD: Computer Science Press.
- Van Der Schans, R. 1990. "The WDGM Model, a Formal Systems View on GIS." *International Journal of Geographical Information Systems*, July-Sept., 4:3, 225-239.
- van Oosterom, P. & T. Vijlbrief. 1991. "Building a GIS on top of the Open DBMS 'Postgres.'" *Proceedings, European Conference, Geographic Information Systems, Brussels, Belgium, Apr. 1991*, 775-787.
- van Oosterom, P. 1990. *Reactive Data Structures for Geographic Information Systems*. Ph.D. Thesis, Department of Computer Science, Leiden University, Netherlands.
- Waugh, T.C. & R.G. Healey. 1986. "The GEOVIEW Design: A Relational Database Approach to Geographical Data Handling." *Proceedings, Second International Symposium on Spatial Data Handling, Seattle, WA, July 1986*. Williamsville, NY: International Geographical Union, Commission on Geographical Data Sensing and Processing, 193-212.
- Westwood, K. 1989. "Towards Successful Integration of Relational and Quadtree Structures in a Geographic Information System." *Challenge for the 1990s, Proceedings, National Conference on Geographic Information Systems, Ottawa, Feb.-Mar., 1989*. Ottawa: Canadian Institute of Surveying and Mapping, 1086-1034.
- White, M.S. 1984. "Technical Requirements and Standards for a Multipurpose Geographic Data System." *American Cartographer*, 11:1, 15-26.
- Yao, S.B., ed. 1985. *Principles of Database Design, Volume I, Logical Organizations*. Englewood-Cliffs, NJ: Prentice-Hall.
- Yao, S.B., V. Waddle & B.C. Housel. 1985. "An Interactive System for Database Design and Integration." *Principles of Database Design, Volume I, Logical Organizations*. S.B. Yao, ed., Englewood-Cliffs, NJ: Prentice-Hall, 325-360.
- Zhang, G. 1989. *Modelling Consistent Transactions Within Automated Cadastral Land Information Systems*. M.Sc. Thesis, Department of Surveying Engineering, University of Calgary, Alta.



# Glossary

This glossary provides definitions to most of the technical terms found in the thesis. Terms being defined are in **boldface**. Terms in *italics* within a definition are defined elsewhere in the glossary. Various abbreviations are used within definitions: Ex. = examples, Syn. = synonyms, Ant. = antonyms, and Cf. = compare with. Note that while many of these terms and definitions are consistent with those found in the literature, many are specific to this thesis.

**0-cell** *0D, spatial or temporal element*. Syn. point, node, vertex, instant.

**0D** Zero-dimensional. Syn. punctual, nodal, instantaneous.

**1-cell** *1D, spatial or temporal element*. Syn. line, arc, edge, interval, period.

**1D** One-dimensional. Syn. linear, durational.

**2-cell** *2D, spatial element*. Syn. region, area, polygon, face.

**2D** Two-dimensional. Syn. regional, areal.

**3-cell** *3D, spatial element*. Syn. volume, polyhedron, solid.

**3D** Three-dimensional. Syn. voluminous, solid.

**abstract object** *High-level, database object, corresponding directly to a real-world object and not requiring an external key.*

**abstraction** Concisely representating something complex, showing important aspects and hiding irrelevant details.

**abstraction domain** Fundamental way of logically viewing geo-information. Types: *generalization, realization, construction*. Cf. *characterization domain*.

**abstraction relation** *Geographic relation existing within one of the abstraction domains.* Types: *generalized-as, realized-as, implemented-as*.

**accuracy** Closeness to which a recorded fact reflects the truth. Syn. correctness.

**action** Part of a *data manipulation* that specifies what is to be done when a *selection* has been made.

**adjacent** *Joint relation* where entities are not *incident*, but meet indirectly via a lower dimensional entity. Syn. primary neighbor.

**aggregate object** *Compound object* characterized by different types of other objects. Ex. city characterized by name, location, population.

**aggregation** *Relation* between *aggregate object* and its components. Syn. has-a, record-of, or part-of relation.

**algorithmic** Traditional model of computation, following a set of instructions in a defined order. Cf. *heuristic*.

**analytic map** *Map* as a simulation of the world, and used for analysis, transformation, or archiving of *geo-data*. Central component of a *GIS*. Consists of *geographic objects* and *operations*. May be *real* or *virtual*. Information is derived from the map after transformation into graphic form. Cf. *graphic map*.

**application** Problem area for which a computer system is used.

**application program** Software for solving specific problems, and existing between a user and an underlying computer system.

**applied level** Highest of three levels of the *construction domain*. Portion of a *GIS* seen by an end-user. Consists of things of specific relevance to particular geographic themes or applications. Syn. specific level.

**applied object** *High-level object* in a *geographic database*. Ex. river, forest, hill. Implemented using alternative *basic-level* objects. Syn. specific object.

**attribute** *Database object* describing useful property or characteristic of an *entity*. Ex. name, number, height, weight. Types: *numeric/symbolic*, *stored/computed*, *uni/multivalued*.

**attribute domain** Possible range or set of values that an *attribute* may take.

**attribute value** Particular instance of an *attribute*.

**attribute-mapping dependency** *Dependency* concerning the mapping between *attributes*, i.e., the number of values of one attribute that are associated with another attribute value at any instant. Types: *univalued*, *multivalued*.

**basic level** Intermediate level of the *construction domain*. Seen by an application developer, i.e., one who customizes a *geographic database* for a particular class of usage. Contains *basic objects* and *operations*. Syn. generic level.

- basic object** Intermediate level *geographic database object*. Ex. feature, layer, geometry, containment. Provided by a *geographic DBMS* as building block upon which a variety of *applied objects* may be constructed, and built using *primal objects*. Syn. generic object.
- bitmap** Simple 2D, *raster object*. Each *pixel* is encoded by a single bit.
- BLOB (binary large object)** Primitive data type comprising a long string of bits that must be decoded to obtain useful data.
- boolean** Having only two possible states or values. Ex. yes/no, on/off. Syn. binary, logical.
- buffer** *Spatial element* containing a given entity and whose boundary is a given distance away from the entity.
- cadastre** Public record of information on physical, legal, and fiscal aspects of land parcels.
- calculate** *Primitive operation* deriving new values by processing given values (arguments). Types: arithmetic, geometric, statistical, textual, etc.
- cardinality** Number of instances of each related type allowed to participate in an occurrence of a *relation*. Syn. mapping, connectivity.
- cartographic generalization** Process of providing *geographic data* at one *scale* given more detailed data.
- categorical coverage** *Region network* with *qualitative content*. Cf. *choropleth coverage*.
- centroid** Geometric centre, somehow defined, of an entity.
- characterization domain** Fundamental way of logically viewing *geo-information*. The information can be in the form of *attributes* of or *relations* between *entities*. Types: *topical, spatial, temporal*. Cf. *abstraction domain*.
- choropleth coverage** *Region network* with *quantitative content*. Cf. *categorical coverage*.
- chronology** Temporal characterization of a *feature*. Syn. history, chronicle.
- classification** *Relation* between instances and types of objects (instances are classified into types). Syn. instantiation.

- classification scheme** Structure or system of classes or categories and their more specialized and more generalized classes or categories.
- coincident** *Intersect relation* when two phenomena occur at exactly the same place or time. Syn. equal.
- collateral** *Spatial relation* where two entities are within a common *enclosing* entity.
- comparison** *Calculation* producing a *boolean* result, based on some relation between two arguments. For each type of relation there is a possible comparison operation. Ex. coincidence, inside, overlap, joins.
- composed-of** *Semantic, geographic relation* between *compound feature* and its parts.
- composite entity** *3D, heteroplethic feature* extending both in space and in theme, height, or time. Types: *topical, spatial, temporal*.
- composite function** *Geographic function* representing the integration or overlay of two or more *layer functions*. The two layers may be from different themes and at the same time, or at different times from the same theme.
- composite object** *Compound object* composed of several other *objects* of one type. Ex. province composed of a group of districts. Syn. set object.
- composition** *Relation* between a *composite object* and its components. Syn. set-of or member-of relation, association, grouping, partitioning.
- compound feature** *Feature* made up of a group of other features. Cf. *elemental feature*.
- compound object** Object composed of other objects. Types: *aggregate, composite*. Syn. nonatomic or complex object.
- compound operation** *Operation* seen as a single and complete unit, but made up of a group of other operations. Specified using a *DML*. Types: *queries, transactions*.
- computational dependency** *Dependency* where *database object* is a function of, or derived computationally from, other database objects.
- conceptual database** *Database* at intermediate level of *abstraction*. Integration of all the data sets that the community of users sees. Syn. main or enterprise database.
- concurrency** Ability to allow several *database* users to access the same data at the same time.
- consistency-preserving transaction (CPT)** *Transaction* transforming a *database* from one consistent state to another.

**constraint** Condition on objects and *operations*, restricting the possible instances of a database, used to prevent violations of *integrity*. Types: inherent, explicit, implicit.

**construction domain** *Abstraction domain* concerning the specification and implementation of *geographic database objects*. Levels: *applied, basic, primal*. Syn. specification-implementation domain.

**construction independence** Ability to express geographic concepts independently of the structures or data types used to represent or implement them.

**content** Topical *attribute value*.

**contiguous** *Relation* when two phenomena touch in space or time. Types: *incident, adjacent*. Syn. joint.

**contiguous subdivision** *Hierarchical subdivision* where each component entity is touching at least one other.

**continuous surface** Horizontal *layer* having quantitative *content* that varies throughout horizontal space. Types: smooth, rugged, stepped. Ex. air pressure (smooth), terrain elevation (rugged), population density (stepped). Cf. *discrete surface*.

**control point** Positioned *feature* to which other features' positions are related.

**controlled dependency** Proper management of *computational dependency* in a *database*. Methods: (i) the dependent object is never stored, but it is automatically computed, from the independent objects, whenever necessary, (ii) the dependent object is always stored, but it is recomputed automatically whenever the independent objects change.

**controlled redundancy** Either elimination of duplication in a *database*, or management of database such that inconsistency is prevented.

**coordinate based geo-referencing** Directly describing an *entity's* position using stored coordinates.

**coordinate geometry (COGO)** Family of techniques for computing positions of unknown points and lines using measurements from known points and lines.

**coordinate system** Numerical system of uniquely and unambiguously specifying positions in space, relative to an arbitrary origin. Uses a pair, or triplet, of coordinates. Types: *plane, geodetic*.

**data definition language (DDL)** Subset of a *data language* for specifying *database schemas*. Cf. *DML*.

**data language** Means of specifying both the structure of a *database* and the *operations* performed on the database. Part of a *DBMS*.

**data manipulation language (DML)** Subset of a *data language* for specifying database *operations*. Types: *procedural, declarative*. Syn. query language.

**data quality** Combination of *accuracy, resolution, precision, and lineage*. Syn. uncertainty, reliability.

**data structure** Structural grouping of data elements into a compound item. Ex. set, list, record, array, tree, *network*, etc.

**database** Large, organized collection of related data, held on a mass storage device.

**database abstraction** The same information existing at different levels of complexity, and the details of data viewed at one level being hidden from those at the next higher level. Levels: *internal, conceptual, external*.

**database dynamics** Behavioral or operational aspects of a *database model*. Mechanism for manipulating objects' instances and types. Specified using a *DML*. Cf. *database statics*.

**database management system (DBMS)** Software that manages a *database*. Provides a common and controlled means of accessing the data, and used directly or indirectly by all database users. Concrete implementation of a *database model*.

**database manipulation** *Geographic operation* comprising an *action* and a *selection*.

**database model** Conceptual framework for the design, construction and use of a *database*. Guiding principle that dictates the number and type of different classes of storable data, their relations, and the query and altering operations that are available. Types: *logical, physical*.

**database object** Thing or concept of potential interest or significance and which can be recorded in a *database* as data or information. Types: *entity, attribute, relation*. Syn. database element.

**database schema** Design or specification of a particular *database*. Remains relatively constant. Syn. database intension. Cf. *database view*.

**database statics** Structural aspects of a *database model*, concerning what information can be in a database. Mechanism for defining objects, attributes, and types. Specified using a *DDL*. Cf. *database dynamics*.

**database system** Portion of an *information system* that provides the function of data storage and retrieval. Consists of hardware, a *database*, a *DBMS*, and *application programs*.

**database view** Content of a *database*, i.e., a set of related data. Often varies over time. Syn. database extension. Cf. *database schema*.

**datalogical model** *Low-level, logical database model*, dealing with files, records, fields, and links. Ex. hierarchical, network, relational. Syn. conventional, syntactic, or record-based model. Cf. *infological model*.

**DBMS** *Database management system*.

**DDL** *Data definition language*.

**declarative language** *DML* focussing on what is to be done, more than how, by expressing the database operation in an abstract form, leaving out the details. Syn. non-navigational. Cf. *procedural language*.

**dependency** When the existence or value of one *database element* is determined by or dependent on the existence or value of one or more other elements. Types: *attribute-mapping, computational*.

**described-by** *Semantic, geographic relation* between *entity* and its *attributes*.

**digital map** *Virtual map* held in a computer file or *database*.

**discontiguous subdivision** *Hierarchical subdivision* where the *compound entity* has at least two disjoint parts. Ex. an archipelago.

**discrete surface** Horizontal *layer* having content that is constant within particular localities but changes abruptly across boundaries. May be a horizontal grouping of all features that belong to one theme. Syn. discrete layer, feature network. Cf. *continuous surface*.

**distance** Separation between two points in space. May simply be straight measure between two positions, or measure may be constrained to be along a particular direction or over a particular surface.

**distributed database** One large data set stored or made available at several different sites.

**DML** *Data manipulation language*.

**during** Asymmetrical, temporal *intersect* relation when one phenomenon is completely within the time frame of the other.

- elemental feature** *Feature* not made up of any other features. Syn. atomic feature. Cf. *compound feature*.
- encapsulation** Software engineering technique where *operations* relevant to a data or object type are defined along with its structural definition. Syn. abstract data typing (ADT).
- enclose** Asymmetrical spatial *intersect* relation where one phenomenon's space completely encompasses another's. Syn. cover, include, contain. Ant. within, inside.
- entity** *Database object* having independent significance. Characterized by some descriptive or statistical *attributes* and by some *relations* with other entities.
- entity-based query** *Retrieval* of characteristics of a given *entity*. Cf. *probe-based query*.
- envelope** Minimum and maximum coordinates of the extent of an *entity*. Syn. bounding rectangle.
- equal** Spatial *intersect* relation where the two phenomena share exactly the same space. Syn. coincident.
- error** Difference between data and truth. Cf. *accuracy*.
- exhaustive** Accounting for everything. Syn. total, continuous.
- external database** *Database* at the highest *level of abstraction*. Subset of the main database, seen by individual group of users for particular application. Syn. sub-database.
- external key** User-assigned *key*. Cf. *internal key*.
- feature** *Geographic entity* having a collection of topical, spatial, temporal, and scale characterizations. Types: *elemental/compound, homo/heteroplethic*.
- feature network** *Discrete surface* composed of a *network of features*.
- finite processor** Human or machine limited in its capability to process large masses of data.
- generalization** *Relation* between a general *object type* (a supertype) and more specialized object types (the subtypes). Syn. is-a or subtype relation. Ant. specialization.
- generalization attribute** *Scale* of a *geographic entity*. Comprises *map scale* and *data quality*.



**generalization domain** *Abstraction domain* concerning the level of detail or complexity of *geographic phenomena*, the scope of any geographic investigation, or the *data quality* or uncertainty of *geo-data*. Syn. scale domain.

**generalized-as** *Abstraction relation* between same *feature* at two different *scales*.

**geo-data** *Geographic data*.

**geo-info** *Geographic information*.

**geo-processing** (Discipline concerning the) processing of *geo-data*.

**geo-referencing** Describing the position of *geographic phenomena*. Types: *relational*, *metrical*. Syn. geo-coding.

**geodetic coordinate system** *Coordinate system* positioning points with respect to a spherical or ellipsoidal surface. Commonly uses a pair of angular distances. Syn. global, spherical, or geographic coordinate system.

**geographic attribute** *Attribute* characterizing a *geographic entity*. Belongs to one of the three *characterization domains* or to the *generalization domain*. Types: *taxonomy*, *geometry*, *chronology*, *scale*.

**geographic data (geo-data)** Informaion or data on *geographic phenomena*. Syn. geo-information.

**geographic database** Special kind of *database* holding *geo-data*.

**geographic database management system** Special *DBMS* controlling the nature and content of a *geographic database*.

**geographic entity** *Entity* representing a *geographic phenomenon*. Characterized by *geographic attributes*, *relations*, and *operations*. Ex. *feature*, *layer*, *profile*, *composite*. Types: *analytic*, *graphic*. Levels: *basic*, *applied*. Syn. map entity.

**geographic function** Mapping between *characterization domains* — a variable in one domain depending on that in the other. Syn. field.

**geographic information (geo-info)** Informaion or data on *geographic phenomena*. Syn. geo-data.

**geographic information system (GIS)** Complex but integrated configuration of hardware, software, data, principles, and procedures for collecting, processing, analyzing, archiving, displaying, or disseminating topical, spatial, and temporal information on the natural and human phenomena existing in a region of interest.

**geographic semantic database model (GSDM)** Special *semantic database model* concerning *geo-data*.

**geographic object** Geographic *database object*. Types: *geographic entity*, *attribute*, and *relation*.

**geographic operation** *Operation* retrieving or updating information in a *geographic database*.

**geographic phenomenon** Thing or concept of interest and related to a place in the world.

**geographic relation** *Relation* between *geographic entities*. May be stored in the database or be derived when necessary. Types: *semantic*, *topologic*, *abstraction*.

**geometry** Spatial characterization of an *entity*. Comprises *metrology* and *topology*.

**GIS** *Geographic information system*.

**graphic map** *Map* functioning as a visual representation of the world — used for communication, interpretation, or archiving of *geo-data*, or for decoration. Visualization of a part of an *analytic map* — a subset of themes, within some region, at a particular time, and at a certain scale and projection. May be a single map sheet or file with no intelligence, such as a display or plot file, or may be an intelligent collection of graphic map elements, or symbols. Cf. *analytic map*.

**GSDM** *Geographic semantic database model*.

**has-subtype** *Semantic, geographic relation* between an *entity type* and its subtypes.

**height** *1D* metrical attribute. *Distance* vertically above a reference surface.

**heterogeneous subdivision** *Hierarchical subdivision* where components are of different types. Ex. a road made up of sections and intersections. Cf. *homogeneous subdivision*.

**heteroplethic feature** *Feature* whose *content* varies within. Ex. *layer*, *profile*, *composite*. Cf. *homoplethic feature*.

**heuristic** Computational model attempting to emulate human intelligence, following a set of rules in no predefined order. Cf. *algorithmic*.

**hierarchical geo-referencing** Positioning features on one level of a spatial *hierarchy* in terms of other features that are on a lower level in the hierarchy.

**hierarchical subdivision** *Compound entity* spatially or temporally composed of other smaller or shorter-duration ones. Varies according to exhaustiveness, exclusiveness, homogeneity, and contiguity.

**hierarchy** *Network* having order but no cycles. Generally, each element is linked to one superior and to several subordinates. Syn. *tree*. Cf. *lattice*.

**high-level** Sophisticated or complex *level of abstraction*, relatively close to reality and human understanding. Not showing details. Cf. *low-level*.

**history function** *Geographic function* representing the mapping between the *topical* and *temporal domains* at a given place. Syn. *chronology*, *time-series*, *time line*, *time-table*.

**homogeneous subdivision** *Hierarchical subdivision* where components are of the same dimensional type. Ex. a particular crop composed of fields. Cf. *heterogeneous subdivision*.

**homoplethic feature** *Feature* whose internal *content* is constant. Ex. lake, road, town (at small scale). Cf. *heteroplethic feature*.

**honesty** Principle that *data quality* must be truthfully or realistically presented.

**implemented-as** *Abstraction relation* between *object* and its form at a lower level of *construction*. Ex. lake:feature, price:numeric, terrain:layer, layer:quadtree

**incident** *Joint relation* where phenomena meet directly and are of different dimensional types. Syn. *bounds*, *boundary*. Cf. *adjacent*.

**infological model** *High-level, database model* dealing with *entities, attributes, relations, and operations*. Syn. *semantic or object-based model*. Cf. *datalogical model*.

**information system** Working model of aspects of some portion of the real world. Built out of a *database system* and a collection of *application programs*.

**inheritance** Property of a *generalization* whereby certain characteristics of the subtypes are in common with those of the supertype and do not need to be redefined for the subtypes.

**instant** *0D, temporal element*. Syn. *0-cell, point, moment, epoch, date*.

**integrity** Completeness, correctness, and consistency of a *database*. Occurs when there is protection against illegal or inappropriate operations being performed on certain data.

**interact** *Primitive operation* getting data from a user (input) or sending retrieved data to a user (output).

**interaction** Action, mapping, or other *relation* between two or more objects. Ex. land ownership between land parcel and land owner. Syn. relation.

**internal database** *Database at lowest of three levels of abstraction.* Concrete, in terms of electronics, but imperceptible, in human terms. Syn. physical or storage database.

**internal key** System-assigned *key*, hidden from users. Cf. *external key*.

**intersect** *Relation* where two phenomena share some space or time. Types: *overlap, enclose* or *during, coincide*.

**joint** *Relation* where two phenomena touch one another, but do not *intersect*. Types: *incident, adjacent*. Syn. contiguous, touch, meet. Ant. disjoint, discontinuous.

**key** *Attribute* uniquely identifying an entity. Types: *internal, external*.

**lattice** *Network* having order and cycles. Generally, each element is linked to one or more superiors and subordinates.

**layer entity** *2D, heteroplethic feature* extending horizontally or vertically throughout a study area, at a particular time, for a particular topic. Syn. (horizontal) coverage, surface, (vertical) section.

**layer function** *2D, geographic function* representing the mapping between *topical* and *spatial domains* at a given time.

**layer operation** *Operation* working at the level of the map *layer*. Ex. digitize map, generalize map, join maps, visualize map, overlay themes, separate themes.

**level of abstraction** Degree of complexity or tangibility.

**line** *1D, spatial element*. Syn. *1-cell, arc*.

**line network** *Discrete surface* containing *linear features*. Ex. road network.

**lineage** Background or history of data. Part of *data quality*.

**list** *Structural relation* of a sequence of similar elements.

**location** *2D, geometric attribute*. Unique specification of a point on a horizontal reference surface

**locus** Exactly determined place. Syn. point.

**logical database model** *Database model* oriented towards human understanding, helping us specify what is going on, what information or data are stored, and what can happen to the data. Types: *infological, datalogical*. Cf. *physical database model*.

**low-level** Elementary, simple, or detailed *level of abstraction*, relatively close to electronics and computer understanding. Cf. *high-level*.

**map** Static, scaled, geographic *model* of part of the world. Types: *analytic/graphic, real/virtual*.

**map projection** Mathematical means by which features positioned in *geodetic* space can be mapped in *plane* space. Types: conformal, equivalent, equidistant, azimuthal. Ex. Mercator, Mollweide, sinusoidal, Lambert, stereographic, etc.

**map scale** Level of abstraction of information in a *map*. Large scale means a low level of abstraction, and implies high quality, more detailed data. Small scale means a high level of abstraction, and implies low quality, summarized data. Syn. *RF*.

**measurement based geo-referencing** Indirectly describing a *feature's* position using a network of measurements and *control points*.

**measurement scale** Data, variable, or attribute type. Types: nominal, ordinal, interval, ratio.

**mental map** *Virtual map* held in the mind. Syn. conceptual map.

**meta attribute** *Attribute* concerning other attributes. Ex. *scale, map projection*.

**metadata** Data about data. Ex. *schema, meta attributes*.

**metrical geo-referencing** Positioning an entity using some metrics, or quantities. Types: *coordinate-based, measurement-based*. Cf. *relational geo-referencing*.

**metrology** Part of *geometry* concerned with spatial measurements, such as distance and direction. Syn. metrics. Cf. *topology*.

**metrology attribute** Part of *geometry* or *chronology attribute*. Comprises spatial or temporal extent, location, size, and shape. Cf. *topology attribute*.

**model** Generalized or simplified conceptualization, representation, or simulation of something complicated. Ex. *map, database model*.

**motion function** *Geographic function*, representing the mapping between the *spatial* and *temporal domains* for a given topic.

**multi-scale database** *Geographic database* having no fixed scale. Syn. scaleless.

**multi-typed network** *Feature network* having a combination of *feature types*.

**multiple generalization** When data for the same objects exists at different *scales* for different purposes. Syn. multiple representation, multi-scale, variable scale.

**multivalued attribute** *Attribute* taking a group of *values* of a similar type.

**multivalued dependency** *Dependency* when zero or more values of *attribute Y* are associated with a given value of attribute X, when there is a 1:m or m:m *relation* between attributes X and Y.

**mutually exclusive** Non-overlapping.

**nearest neighbor query** *Retrieval* of nearest *feature* to a point *probe*.

**neighbor** *Spatial relation* where two phenomena are "topologically close." Types: *adjacent, proximal*.

**network** Structure or system of elements comprising nodes and links.

**numeric** *Quantitative*.

**object instance** Particular occurrence of an *object*. Syn. element, item.

**object type** Set of all possible *objects* that may exist and have similar characteristics. Syn. class.

**operation** Action of interest involving *database objects*, querying or altering the database. Types: *primitive, compound*. Syn. activity, event, process.

**orthogonal** Perpendicular, complementary, and independent.

**overlap** *Intersect relation* where some space or time is common to at least part of each phenomenon.

**partial subdivision** *Hierarchical subdivision* where part of the *compound entity* is unaccounted for by its components. Ex. cliffs, tracks, and buildings making up part of a quarry. Syn. non-exhaustive subdivision.

**period** *ID, temporal element*. Syn. *1-cell*, interval, duration.

**physical database model** *Database model oriented towards machine concepts and specifying how data are stored and retrieved at the computer system level. Cf. logical database model.*

**pixel** *Picture element or cell. Small, usually square, element of a raster.*

**planar map** *Exhaustive partitioning of space into mutually exclusive regions.*

**plane coordinate system** *Coordinate system positioning points with respect to a flat surface. Types: rectangular, polar. Cf. geodetic coordinate system.*

**planimetric map** *Map obtained from an orthogonal view from vertically above the land. Syn. orthographic.*

**point** *0D, spatial or temporal element. Syn. 0-cell, node, epoch, instant.*

**point network** *Feature network containing point features.*

**point-in-polygon-network query** *Retrieval of region containing a point probe.*

**position** *3D geometric attribute, sometimes made up of a location and a height.*

**precision** *Central tendency of a set of measurements. Syn. repeatability, variance.*

**predicate** *Clause restricting the result of a selection. Consists of a set of conditions that must hold true for the specified subset of the database to exist.*

**primal level** *Lowest of three levels of the construction domain. Consists of primal objects. Syn. atomic or elemental level.*

**primal object** *Low-level, geographic database construct, used by a geographic DBMS developer to build basic objects. Ex. line, polygon, bitmap, region, quadtree. Syn. atomic or elemental object.*

**primary neighbor** *Adjacent.*

**primitive object** *Database object not composed of any other object. Ex. integer, real, BLOB. Syn. atomic or elementary object.*

**primitive operation** *Single operation that operates on a set of primitive objects and is not composed of any others. Provided as a basic part of a DBMS. Syn. atomic, simple or basic operation.*

**probe** *Item used in specification of predicates during database operations. May be of arbitrary topical, spatial, temporal, and scale extents. During interaction with the*

database a user specifies a probe and asks for those database entities related to the probe. Syn. interaction or query object.

**probe-based query** *Retrieval of entities related to a given probe. Ex. range, point-in-polygon-network, and nearest neighbor queries. Cf. entity-based query.*

**procedural language** *DML for expressing database operations as a series of basic operations, in which case a high-level operation must be formulated using many steps. Syn. navigational. Cf. declaritive language.*

**profile entity** *1D, heteroplethic feature with content depending on theme, location, or time. Types: topical, spatial, temporal. Conceptually, passes through a series of layers.*

**profile function** *1D, geographic function relating content to topic, distance, or time.*

**program-data independence** *When application programs are independent of the way database data are stored, and data are independent of the way programs are implemented. Changes to one will not affect the functionality of the other.*

**propagation** *That some characteristics of a compound object depend on those of constituent objects.*

**proximal** *Disjoint, or discontinuous, relation where the entities are connected via a common joint entity. Syn. secondary neighbor.*

**quadtree** *Hieararchical, 2D, raster object. Each pixel is recursively subdivided into four sub-pixels until the desired resolution for that pixel is reached.*

**qualitative data** *Data involving descriptive classes or categories. Types: nominal, ordinal. Syn. symbolic, discrete. Cf. quantitative data.*

**quantitative data** *Data involving metrical or statistical values. Types: interval, ratio. Syn. numeric, continuous. Cf. qualitative data.*

**query** *Compound operation that peruses, or retrieves information from, a database. Does not change the database content. Cf. transaction.*

**range query** *Retrieval of objects intersecting a rectangular probe.*

**raster encoding** *Method of encoding an entity's spatial or temporal extent. Space or time is considered to consist of a fine array of pixels and an entity is represented by the set of pixels which it covers. Syn. space-based, image-based, gridded, tessellation, cellular, space-filling. Cf. vector encoding.*

**raster object** *Raster-encoded object. Types vary according to cell attribute type and cell divisibility. Ex. bitmap, quadtree.*



- raster spatial extent** *Spatial or temporal extent composed of network of raster objects.*
- real map** Directly viewable and permanently tangible *graphic map*, such as on a sheet of paper or drafting film. Syn. physical map. Cf. *virtual map*.
- real world** *Geographic phenomena and their interactions.* Syn. reality.
- realization domain** *Abstraction domain concerning the different forms geographic phenomena or information exhibit, from the concreteness of the real world to the abstractness of mental maps, with the analytic and graphic maps in between.*
- realized-as** *Abstraction relation between analytic entity and its graphic forms.* Ex. road realized-as red line, surface as contour network, composite as plot file.
- recursive** Defined in terms of itself.
- redundancy** When any fact is recorded more than once in a *database*. Possibly resulting in wasted space and inconsistency.
- region** 2D element of horizontal space. Syn. 2-cell, zone, area, polygon.
- region network** *Feature network consisting of regions.*
- relational geo-referencing** Describing the position of *features* in terms of other features. Types: *topological, hierarchical*. Cf. *metrical geo-referencing*.
- relation** Meaningful linking, connection, association, or grouping of *database objects*. Ex. marriage, membership, ownership, adjacency, subdivision. Levels: *structural, semantic*. Types: stored, derived.
- relativism** Idea that the same piece of information can be seen in alternate ways by different users.
- representative fraction (RF)** *Quantitative expression of map scale.* Ratio of distance on a *graphic map* to the corresponding distance on the ground.
- resolution** Smallest item or value that can be identified, distinguished or recorded. Syn. granularity, detail, least count.
- retrieval** Getting information from a *database*.
- retrieve** *Primitive operation* getting a specified part of a *database* and assigning it to some temporary variable.
- RF** *Representative fraction.*

- scale** *Geographic attribute* describing *map scale* and *data quality* of a *feature*.
- scale dependence** *Computational dependence* where smaller scaled data depend on larger scaled data, and summary data depend on detailed data.
- scalelessness** Apparent lack of a fixed map scale in a *multi-scale database*. Given a particular investigation, the appropriate sized area and level of detail is automatically presented to a user. Syn. *scale variability*.
- schema evolvability** Ability to let *database schema* evolve with changing application environment without any loss of integrity.
- scope** Extent, or size, of any geographic investigation or study.
- seam** Boundary between *tiles* in a *geographic database*.
- seamless** When a *geographic database* hides the details of the data collection or storage boundaries from the users.
- security** Protection of *database* against unauthorized access and against loss due to system failure.
- selection** Mechanism for specifying a subset of a *database*. Used for defining both conceptual database queries and external database schemas. Part of a *database manipulation*. Syn. *retrieval*.
- semantic database model** *High-level, database model*. Allows data to more accurately and directly reflect real-world entities or concepts. Defines both the structural and behavioral characteristics of stored data — structure being defined in terms of *entities, attributes, and relations*, and behaviour in terms of *queries and transactions*. Syn. *infological model*.
- semantic relation** Meaningful structural grouping of objects. Has defined characteristics. Types: *generalization* between a type and sub-types, *aggregation* of dissimilar types into a compound type, *composition* of homogeneous types into a compound type, and *interaction* between otherwise independent types.
- separation** Shortest distance, somehow defined, separating two entities. Syn. *shortest path*.
- set operation** *Operation* involving sets. Types: intersection, union, difference.
- size** Length, area, or volume of an *entity*.
- skeleton** Centre line, somehow defined, of an *entity*.

**software abstraction** Software engineering technique where a software module's specification, or logical behaviour, and its implementation, or the details of how it works, are both clearly defined and separate. A module's specification is known to its users, but its implementation is hidden.

**spatial attribute** *Geometry of a geographic entity. Components: metrology, topology.*

**spatial calculation** *Primitive, spatial operation, used together with retrieve and update to effect queries and transactions. Types vary according to number and types of arguments and results. Ex. envelope, contains, centroid, smooth, distance, union, etc.*

**spatial domain** *3D, characterization domain concerning geometry of geographic phenomena. Syn. geometric domain.*

**spatial element** *Subdivision of space. Types: 0-cell, 1-cell, 2-cell, 3-cell.*

**spatial extent** *Part of a spatial metrology attribute describing space occupied by entity. Types: vector, raster.*

**spatial operation** *Geographic database operation performed on spatial data.*

**spatial profile** *Profile entity having a consecutive sequence of values for a given topic and time. Types: horizontal/vertical, discrete/continuous. Ex. elevation series, geologic core.*

**spatial relation** *Relation between two entities based on their spatial characteristics. Types: topological, metrical.*

**spatial retrieval** *Getting objects from a geographic database according to spatial criteria. Types: entity-based, probe-based.*

**spatial update** *Operation creating, deleting, or changing the spatial content of a geographic database. Ex. adding or removing individual points, lines, or regions, and joining or dividing such features.*

**structural relation** *Grouping of data into a compound item according to some pattern. Ex. set, list, array, tree, network, and record. Syn. data structure, data type constructor.*

**subject** *Part of a selection indicating what is to be operated on.*

**subschema** *Subset of database schema. Establishes the context of a selection.*

**summary** *Characterization of a group of objects by a single attribute, such as the total, mean, or variance.*

**surveying** Collecting *geo-data* from the field, i.e., any of land, sea, air, or space surveying.

**symbolic** *Qualitative*.

**taxonomy** Topical characterization of an *entity*. Syn. *thematic profile*.

**temporal attribute** *Chronology* of a *geographic entity*. Comprises temporal *metrology* and *topology*.

**temporal calculation** *Calculation* involving temporal data. Ex. separation, before/after, during.

**temporal composite** *Composite entity* conceptually grouping *layers*, for a given theme, over time.

**temporal domain** *ID, characterization domain* concerning history or chronology of *geographic phenomena*. Syn. historical or chronological domain.

**temporal element** Subdivision of time. Types: *0-cell, 1-cell*.

**temporal extent** Part of a temporal *metrology attribute*, stating when something was. Types: *vector, raster*.

**temporal operation** *Operation* performed on temporal data. Types: *calculation, retrieval, update*.

**temporal profile** *Profile entity* having a consecutive sequence of *values* for a given *topic* and *location*. Types: discrete, continuous. Syn. time series, time line.

**temporal relation** *Relation* between two entities based on their temporal attributes. Types: consecutive, *intersect*.

**temporal retrieval** *Retrieval* based on temporal criteria, such as when did an entity exist, or what entities existed before a particular date.

**temporal update** Creating, deleting, or changing temporal data.

**thematic composite** *Composite entity* conceptually grouping *layers* of different themes at a given time.

**theme** Grouping of similar or related phenomena or data. Ex. topography, geology, vegetation, population, economics, engineering, utilities. Syn. topic, subject.

**tile** Arbitrary *region* within which *geo-data* are collected and stored.

**TIN (triangulated irregular network)** *Network* of irregular triangles or their vertices and sides.

**topical attribute** General, non-spatial, non-temporal *attribute* describing what something is.

**topical domain** Multi-dimensional *characterization domain* concerning (non-spatial, non-temporal) categorical or statistical nature of geographic phenomena. Syn. taxonomic, thematic, or attribute domain.

**topical operation** *Operation* performed on topical data.

**topical profile** *Profile entity* having *contents* from a set of themes at a given place and time. Syn. thematic core or profile.

**topical relation** General, non-spatial, non-temporal linking of *geographic entities*.

**topography** Natural and man-made surface features and cultural entities of a region.

**topological geo-referencing** Positioning *entities* of a *planar map* in terms of other entities in the same planar map.

**topology** Part of *geometry* concerned with *spatial relations*, such as contiguity, intersection, and neighbor, which are unaffected by continuous deformation. Cf. *metrology*.

**topology attribute** Part of *geometry* or *chronology attribute* comprising bounds, overlaps, contains, and inside/during *relations*. Cf. *metrology attribute*.

**total subdivision** *Hierarchical subdivision* where the entire *compound entity* is accounted for by its components. Syn. exhaustive subdivision. Cf. partial subdivision.

**transaction** *Compound operation* altering the content of the *database*. Often represents an actual event in the real world. Ex. change in land use, growth in population. Built from other operations — a series of other transactions, *primitive operations*, or programming language constructs. Syn. application operation.

**transaction logging** Keeping a record of all *transactions* that have happened. Important for both lineage and security purposes.

**univalued attribute** *Attribute* taking a single *value*.

**univalued dependency** *Dependency* when one *attribute X* uniquely determines another attribute *Y*, when there is a 1:1 or m:1 *relation* between attributes *X* and *Y*. Syn. functional dependency

- update** *Primitive operation* modifying the content of the *database*. Types: create, delete, change.
- user** Someone or something that designs or access a *database* via a *DBMS*. Ex. end-user, database administrator, programmer, program, device.
- variable integration** Ability to totally, partially or not store the *relations* between elements of different *layers*.
- vector** Quantity having magnitude and direction. Has two or more fixed number of elements. Cf. scaler.
- vector encoding** Method of encoding an object's spatial or temporal extent. Entities are delineated using ordered sets of coordinates that form the outline or centre line. Syn. coordinate-based, object-based, calligraphic, boundary representation. Cf. *raster encoding*.
- vector object** *Vector-encoded* object. Types: node/point, arc/interval, polygon, solid.
- vector spatial extent** *Spatial or temporal extent* composed of *network* of *vector objects*.
- virtual map** *Map* that is not *real*. Types: (i) not permanently tangible, e.g., computer screen image or projected image, (ii) not directly viewable, e.g., photographic film or slide, or (iii) neither of the two, e.g., digital file or database, or mental image. Cf. *real map*.
- visualization** Cartographic process of designing and constructing a *graphic map* from an *analytic map*. Entails selection, projection, symbolization, and rendering.
- voxel** Volume element or cell. Small, usually cubic, element of a 3D *raster*.
- window** Region of a computer screen within which information, such as a map, is viewed. Syn. viewport.