



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

**INPE-11454-RPE/770**

## **ANATOMIA DE SISTEMAS DE INFORMAÇÃO GEOGRÁFICA**

Câmara, Gilberto  
Casanova, Marco A.  
Hemerly, Andrea S.  
Magalhães, Geovane C.  
Medeiros, Claudia M. B.

# Anatomia de Sistemas de Informação Geográfica

por

Gilberto Câmara<sup>1</sup> Marco A. Casanova<sup>2</sup> Andrea S. Hemerly<sup>2</sup>

Geovane C. Magalhães<sup>3,4</sup> Claudia M. B. Medeiros<sup>4</sup>

<sup>1</sup>Divisão de Processamento de Imagens  
INPE

<sup>2</sup>Centro Latino Americano de Soluções  
para Educação Superior e Pesquisa  
IBM Brasil

<sup>3</sup>Departamento de Sistemas de Operação  
CPqD/TELEBRÁS

<sup>4</sup>Instituto de Computação  
UNICAMP



# Prefácio

*Sistemas de Informação Geográfica*, ou abreviadamente SIGs, são sistemas de informação construídos especialmente para armazenar, analisar e manipular dados geográficos, ou seja, dados que representam objetos e fenômenos em que a localização geográfica é uma característica inerente e indispensável para tratá-los. Dados geográficos são coletados a partir de diversas fontes e armazenados via de regra nos chamados *bancos de dados geográficos*.

Este texto trata da anatomia de um SIG segundo três pontos de vista – usuário final, projetista de aplicações e projetista de bancos de dados geográficos. Em função destes múltiplos enfoques, o texto está dividido em quatro partes, abordando respectivamente: aspectos conceituais gerais; problemas de especificação de dados e de consultas; questões de processamento e armazenamento de dados; e dois grandes projetos brasileiros envolvendo tecnologia de Geoprocessamento.

Em detalhe, a parte I, dividida em três capítulos, apresenta os principais conceitos necessários ao entendimento do resto do texto. Assim, o capítulo 1 contém um breve apanhado de questões relativas a cartografia e sensoriamento remoto. O capítulo 2 apresenta conceitos gerais de SIG, resumindo suas principais características funcionais e incluindo questões de modelagem de processos e de dados do mundo real. O capítulo 3 introduz as principais classes de dados geográficos.

A parte II é composta por quatro capítulos. O capítulo 4 concentra-se na modelagem de dados para bancos de dados geográficos, propondo um modelo geral. O capítulo 5 discorre sobre as principais operações primitivas de manipulação de dados encontradas em um SIG. O capítulo 6 trata de linguagens de consulta. O capítulo 7 discute a questão de interfaces para SIG, considerando os problemas de visualização do resultado de consultas e de arquitetura de interfaces.

A parte III compreende três capítulos. O capítulo 8 analisa a arquitetura de um SIG, a partir do modelo proposto na parte II. O capítulo 9 discute as questões de armazenamento e indexação de dados geográficos. O capítulo 10 aborda gerenciamento de transações e otimização de consultas, dois aspectos da implementação de um SIG.

A parte IV discorre sobre dois projetos desenvolvidos no Brasil e sobre tendências de armazenamento distribuído de dados geográficos. O capítulo 11 descreve o sistema SPRING, um SIG desenvolvido pelo INPE, cujo modelo e linguagem de consulta baseiam-se naqueles introduzidos na parte II do livro. O capítulo 12 apresenta o projeto SAGRE, desenvolvido no

CPqD/TELEBRÁS, para gerenciamento da rede externa de telefonia. Encerrando o texto, o capítulo 13 aborda a questão de centros de dados geográficos.

Este texto foi escrito como parte das atividades do Projeto GEOTEC, um projeto multi-institucional dentro do Programa Temático em Ciência da Computação e voltado para o desenvolvimento de ferramentas e tecnologias para Sistemas de Informação Geográfica. Os autores gostariam de agradecer ao CNPq/MCT o apoio recebido para realização deste projeto.

Os agradecimentos se estendem também a Claudia Tocantins, Antonio Machado e Silva, Juliano Oliveira, Cristina Ciferri e Mariano Cilia pelas inúmeras contribuições diretas e indiretas à preparação do texto. A Maurício Mediano cabe um agradecimento especial pela cuidadosa leitura da versão final do texto.

Rio de Janeiro, abril de 1996

A.S.H.

C.M.B.M.

G.C.

G.C.M.

M.A.C.

# Sumário

<b>PARTE I</b>	<b>5</b>
<b>1 Conceitos Básicos</b>	<b>5</b>
1.1 Introdução . . . . .	5
1.2 Conceitos de Cartografia . . . . .	5
1.2.1 Definições Básicas . . . . .	5
1.2.2 Sistemas de Coordenadas . . . . .	6
1.2.3 Projeções Cartográficas . . . . .	7
1.2.4 Mapas e Cartas . . . . .	14
1.3 Conceitos de Sensoriamento Remoto . . . . .	15
1.3.1 Princípios Físicos . . . . .	16
1.3.2 Sensores a Bordo de Satélites . . . . .	17
1.3.3 Câmaras Fotográficas Aero-Transportadas . . . . .	20
<b>2 Sistemas de Informação Geográfica</b>	<b>21</b>
2.1 Introdução . . . . .	21
2.2 Conceitos Básicos . . . . .	22
2.2.1 Caracterização e Componentes . . . . .	22
2.2.2 Processo de Implantação . . . . .	24
2.2.3 Breve Histórico . . . . .	25
2.3 Aplicações . . . . .	27
2.3.1 Classificação das Aplicações . . . . .	27
2.3.2 Aplicações Sócio-Econômicas . . . . .	28
2.3.3 Aplicações Ambientais . . . . .	31
2.3.4 Aplicações de Gerenciamento . . . . .	34

<b>3</b>	<b>Caracterização de Dados Geográficos</b>	<b>37</b>
3.1	Introdução . . . . .	37
3.2	Características Gerais de Dados Geográficos . . . . .	37
3.3	Campos e Objetos Geográficos . . . . .	39
3.3.1	Modelos de Campos e de Objetos . . . . .	39
3.3.2	Representações para Campos e Objetos . . . . .	40
3.4	Principais Classes de Dados Geográficos . . . . .	41
3.4.1	Mapas Temáticos e Mapas Cadastrais . . . . .	41
3.4.2	Redes . . . . .	42
3.4.3	Modelos Numéricos de Terreno . . . . .	43
3.4.4	Imagens . . . . .	44
<b>PARTE II</b>		<b>49</b>
<b>4</b>	<b>Modelagem de Dados Geográficos</b>	<b>49</b>
4.1	Introdução . . . . .	49
4.2	Um Modelo de Dados Geográficos . . . . .	50
4.2.1	Níveis de Especificação . . . . .	50
4.2.2	Nível Conceitual . . . . .	52
4.2.3	Nível de Representação . . . . .	54
4.2.4	Extensões ao Modelo . . . . .	56
4.3	Padrões para SIG . . . . .	57
4.3.1	Visão Geral . . . . .	58
4.3.2	O Padrão SAIF . . . . .	59
4.3.3	O Padrão SDTS . . . . .	62
4.3.4	O Padrão OGIS . . . . .	63
<b>5</b>	<b>Operações sobre Dados Geográficos</b>	<b>65</b>
5.1	Introdução . . . . .	65
5.2	Análise das Operações sobre Geo-Objetos . . . . .	66
5.2.1	Papel das Representações . . . . .	66
5.2.2	Relacionamentos Topológicos . . . . .	67

5.3	Análise das Operações sobre Geo-Campos . . . . .	74
5.3.1	Operações Pontuais . . . . .	74
5.3.2	Operações de Vizinhança . . . . .	76
5.4	Operações Mistas entre Geo-Campos e Geo-Objetos . . . . .	78
<b>6</b>	<b>Recuperação de Dados Geográficos</b>	<b>81</b>
6.1	Introdução . . . . .	81
6.2	Visão Geral de Linguagens de Consulta . . . . .	81
6.2.1	Linguagens Textuais . . . . .	81
6.2.2	Linguagens Visuais . . . . .	82
6.2.3	Ambientes Multimodais . . . . .	83
6.2.4	Linguagens de Consulta sobre Imagens . . . . .	83
6.3	A Linguagem LEGAL . . . . .	84
6.3.1	Definição de Esquemas Conceituais . . . . .	84
6.3.2	Criação de Geo-Objetos e Geo-Campos . . . . .	85
6.3.3	Consultas . . . . .	86
6.3.4	Manipulação de Objetos e Coleções . . . . .	87
<b>7</b>	<b>Apresentação de Dados Geográficos</b>	<b>91</b>
7.1	Introdução . . . . .	91
7.2	Visualização de Dados Geográficos . . . . .	91
7.2.1	Considerações Gerais . . . . .	91
7.2.2	Produção Cartográfica . . . . .	94
7.2.3	Apresentação de Dados em LEGAL . . . . .	98
7.3	Projeto de Interfaces . . . . .	99
7.3.1	Arquitetura . . . . .	101
7.3.2	Linguagens . . . . .	102
7.3.3	Modelo do Usuário . . . . .	103



<b>PARTE III</b>	<b>107</b>
<b>8 Arquiteturas para SIG</b>	<b>107</b>
8.1 Introdução . . . . .	107
8.2 Uma Arquitetura em Camadas . . . . .	107
8.3 Estratégias de Implementação . . . . .	109
8.4 Arquiteturas Distribuídas . . . . .	112
8.4.1 Arquiteturas Cliente-Multiservidor . . . . .	112
8.4.2 Arquiteturas Orientadas-a-Serviços . . . . .	113
<b>9 Armazenamento de Dados Geográficos</b>	<b>117</b>
9.1 Introdução . . . . .	117
9.2 Estratégia Elementar de Armazenamento . . . . .	117
9.3 Armazenamento de Representações Matriciais . . . . .	119
9.4 Armazenamento e Indexação de Representações Vetoriais . . . . .	120
9.4.1 Pontos . . . . .	120
9.4.2 Retângulos . . . . .	121
9.4.3 Linhas Poligonais . . . . .	125
9.4.4 Polígonos . . . . .	126
9.4.5 Representações Vetoriais Complexas . . . . .	127
<b>10 Processamento de Transações e Consultas</b>	<b>129</b>
10.1 Introdução . . . . .	129
10.2 Gerência de Transações . . . . .	130
10.2.1 Transações em SIG . . . . .	130
10.2.2 Mecanismos para Implementação de Transações . . . . .	131
10.2.3 Exemplo . . . . .	133
10.3 Processamento de Consultas . . . . .	134
10.3.1 Tipos de Consultas . . . . .	134
10.3.2 Estratégia para Processamento de Consultas . . . . .	135
10.3.3 Exemplos de Otimização de Consultas . . . . .	137

<b>PARTE IV</b>	<b>145</b>
<b>11 O Sistema SPRING</b>	<b>145</b>
11.1 Introdução . . . . .	145
11.2 O Modelo de Dados do SPRING . . . . .	146
11.2.1 Nível Conceitual . . . . .	146
11.2.2 Nível de Representação . . . . .	148
11.3 Interface Interativa e Linguagem de Consulta . . . . .	149
11.4 Funcionalidade . . . . .	150
<b>12 O Projeto SAGRE</b>	<b>153</b>
12.1 Introdução . . . . .	153
12.2 Funcionalidade . . . . .	153
12.3 Componentes e Arquitetura . . . . .	155
12.3.1 Principais Componentes . . . . .	155
12.3.2 Arquitetura do SAGRE . . . . .	156
12.4 Elementos Armazenados e Modelo de Dados . . . . .	157
12.5 Características da Manipulação de Dados . . . . .	158
12.6 Interface . . . . .	160
12.7 Mecanismos Internos . . . . .	164
12.7.1 Conversão dos Dados para Entrada no Sistema . . . . .	164
12.7.2 Controle de Versões . . . . .	165
<b>13 Centros de Dados Geográficos</b>	<b>167</b>
13.1 Introdução . . . . .	167
13.2 Cooperativa de Dados e Centro de Dados . . . . .	167
13.3 Funções de um Centro de Dados . . . . .	169
13.4 Exemplos . . . . .	172



# Lista de Figuras

1.1	Projeção estereográfica polar . . . . .	10
1.2	Projeção cônica de Lambert . . . . .	10
1.3	Projeção de Mercator . . . . .	11
1.4	Projeção plana . . . . .	11
1.5	Projeção cônica . . . . .	12
1.6	Projeção cilíndrica . . . . .	12
1.7	Cartas 1:1.000.000 - UTM . . . . .	14
2.1	Arquitetura de Sistemas de Informação Geográfica . . . . .	24
4.1	Níveis de especificação para troca de dados . . . . .	59
4.2	O modelo de dados (parcial) do SAIF . . . . .	62
5.1	Exemplo da matriz de 4 interseções. . . . .	69
5.2	Topologias linha-região simples com a mesma matriz de 4 interseções. . . . .	69
5.3	Exemplos de relacionamentos topológicos . . . . .	73
5.4	Exemplo de operação de ponderação. . . . .	75
5.5	Exemplo de cálculo de índice de diversidade. . . . .	76
5.6	Exemplo de cálculo de máximo zonal. . . . .	77
5.7	Exemplo de cálculo de interseção espacial. . . . .	79
5.8	Exemplo de reclassificação por atributo. . . . .	79
7.1	Apresentações de uma mesma consulta em resoluções diferentes . . . . .	93
7.2	Apresentações distintas de uma mesma representação . . . . .	94
7.3	Exemplos de generalização . . . . .	97
7.4	Exemplo de arquitetura de uma interface aberta para SIG . . . . .	100
7.5	Enfoques no projeto de interface . . . . .	102

8.1	Arquitetura genérica em camadas para um SIG. . . . .	108
8.2	Arquitetura orientada-a-serviços para SIG. . . . .	114
9.1	Estrutura de um arquivo em grade. . . . .	120
9.2	Particionamento do espaço determinado por uma árvore kd . . . . .	121
9.3	Exemplo de uma árvore R . . . . .	122
9.4	Exemplo de uma árvore R+ . . . . .	124
9.5	Exemplo de z-ordem . . . . .	127
9.6	Exemplo da decomposição de um polígono em células. . . . .	127
11.1	Nível conceitual do modelo de dados . . . . .	146
11.2	Exemplo de Atributos para LOTES . . . . .	147
11.3	Exemplo de Atributos para MAPALOTES . . . . .	147
11.4	Associação de geo-objeto a mapa cadastral . . . . .	147
11.5	Ligação entre geo-objeto e objeto não espacial . . . . .	148
11.6	Exemplo de definição de esquema conceitual no spring . . . . .	150
11.7	Exemplo de implementação em LEGAL . . . . .	151
12.1	Arquitetura do SAGRE . . . . .	156
12.2	Exemplo de tela base do SAGRE . . . . .	159
12.3	Interface de atributos do SAGRE . . . . .	163
13.1	Cooperativa de dados . . . . .	168

# Lista de Tabelas

1.1	Análise comparativa das projeções . . . . .	13
1.2	Características de satélites cujos dados são recebidos no Brasil . . . . .	18
2.1	Evolução da tecnologia de SIGs . . . . .	27
3.1	Exemplos de análise espacial . . . . .	39
5.1	Relacionamentos topológicos possíveis entre regiões simples . . . . .	70
5.2	Regras para determinação de aptidão agrícola. . . . .	75
7.1	Comparação entre cartografia tradicional e cartografia usando SIG . . . . .	95
8.1	Comparação entre as estratégias de implementação . . . . .	109
11.1	Relação entre níveis conceitual e de representação . . . . .	149



# PARTE I





# Capítulo 1

## Conceitos Básicos

### 1.1 Introdução

A implementação de SIGs requer integrar conhecimentos de diversas áreas da Ciência da Computação e de disciplinas relacionadas ao processamento de certos tipos específicos de dados. Para melhor compreensão do texto, este capítulo apresenta inicialmente alguns conceitos de básicos de Cartografia. Em seguida, enfoca a área de Sensoriamento Remoto, abordando alguns tópicos relevantes como, por exemplo, aquisição de dados através de sistemas com sensores passivos.

Este capítulo utiliza algumas referências básicas da área [MGR93a, ABC<sup>+</sup>91, Aro89, PM90]. Em particular, a seção 1.2 é parcialmente baseada em [SPR93].

### 1.2 Conceitos de Cartografia

#### 1.2.1 Definições Básicas

Vista do espaço, a Terra assemelha-se a uma esfera com os pólos achatados. Na realidade, sua forma é afetada pela gravidade, força centrífuga de rotação e variações de densidade de suas rochas e componentes minerais [RSM78]. Devido à complexidade de se trabalhar com a forma real da Terra, os cartógrafos aproximam sua superfície para um modelo do *globo terrestre*. Neste processo de aproximação, inicialmente se constrói um geóide, resultante da medição do nível dos oceanos. Em seguida, aproxima-se o geóide por um elipsóide de revolução, mais regular. Um *elipsóide de revolução* é um sólido gerado pela rotação de uma elipse em torno do eixo menor dos pólos. Por fim, pode-se considerar o próprio elipsóide ou transformá-lo em uma esfera com a mesma superfície, gerando então o globo terrestre.

Estudos geodésicos apresentam valores diferentes para os elementos de um elipsóide (raio do equador, raio polar e coeficiente de achatamento). Assim, cada região deve adotar como referência o elipsóide mais indicado. No Brasil, adotou-se o elipsóide de Hayford, cujas dimensões foram consideradas as mais convenientes para a América do Sul. Atualmente, no entanto,

utiliza-se com mais frequência o elipsóide da União Astronômica Internacional, homologado em 1967 pela Associação Internacional de Geodésia, que passou a se chamar *elipsóide de referência*.

Um *datum* é um ponto onde a superfície do elipsóide de referência toca a Terra, sendo caracterizado a partir de uma superfície de referência (datum horizontal) e de uma superfície de nível (datum vertical). Para a definição do datum escolhe-se um ponto mais ou menos central em relação à área de abrangência. No Brasil, os mapas mais antigos adotam o Datum de Córrego Alegre - MG e, mais recentemente, o Datum SAD 69 (Datum Sul Americano de 1969), porém existem mapas feitos em ambos e até mesmo com datum locais.

Apesar do globo terrestre mostrar distâncias, áreas, formas e direções reais, ele não é o mais apropriado para lidar com processamentos sistemáticos. Para permitir tais processamentos, são criadas *projeções* num plano – ou seja, cada ponto do elipsóide ou esfera é projetado em uma superfície plana. Esta superfície – o mapa – pode ser apresentado em diferentes escalas.

*Escala* é a relação entre as dimensões dos elementos representados em um mapa e a grandeza correspondente, medida sobre a superfície da Terra. A escala é uma informação obrigatória para qualquer mapa e geralmente está representada de forma numérica. As *escalas numéricas* ou *fracionárias* são descritas por frações cujos denominadores representam as dimensões naturais e os numeradores, as que lhes correspondem no mapa.

A escala de 1 para 50.000 (notação 1:50.000 ou 1/50.000), por exemplo, indica que uma unidade de medida no mapa equivale a 50.000 unidades da mesma medida sobre o terreno. Assim, 1cm no mapa corresponde a 50.000cm (ou 500m) no terreno.

Os termos “escala grande” ou “escala pequena” podem confundir. Para esclarecê-los basta lembrar que escalas são representadas por números fracionários. Assim, uma escala 1:100.000 é menor que uma escala de 1:20.000. Ainda, um objeto representado em um mapa em uma “escala pequena” é menor que o mesmo representado em um mapa em uma “escala grande”.

Um dos problemas ao se utilizar SIGs reside no fato de que um mapa digital pode ser apresentado ou impresso em qualquer escala, independentemente da escala de origem em que os dados foram armazenados. Porém, caso a escala escolhida seja maior que a original, não há naturalmente ganho de precisão. Por outro lado, se a escala escolhida for menor, detalhes e precisão podem ser perdidos. O processo de produção de cartas em escala menor do que a de armazenamento dos dados é denominado *generalização*, sendo discutido no capítulo 7.

### 1.2.2 Sistemas de Coordenadas

Um objeto geográfico qualquer (como uma cidade, a foz de um rio, o cume de uma montanha) somente poderá ser localizado se puder ser descrito em relação a outros objetos cujas posições sejam previamente conhecidas, ou se tiver sua localização determinada em uma rede coerente de coordenadas. Quando se dispõe de um sistema de coordenadas fixas, pode-se definir a localização de qualquer ponto na superfície terrestre.

Os sistemas de coordenadas dividem-se em dois grandes grupos: sistemas de coordenadas geográficas ou terrestres e sistemas de coordenadas planas ou cartesianas.

No *sistema de coordenadas geográficas* ou *terrestres*, historicamente mais antigo, cada ponto da superfície terrestre é localizado na interseção de um meridiano com um paralelo. *Meridianos* são círculos máximos da esfera cujos planos contem o eixo dos pólos. O *meridiano de origem* (também conhecido como *inicial* ou *fundamental*) é usualmente aquele que passa pelo antigo observatório britânico de Greenwich. Ele é escolhido convencionalmente como a origem das longitudes sobre a superfície terrestre e como base para a contagem dos fusos horários, correspondendo ao meridiano a  $0^\circ$ . (Esta localização porém é dependente do país, sendo que [Mue86] lista em torno de 30 diferentes meridianos de origem). A leste do meridiano de origem, os meridianos são medidos por valores crescentes até  $+180^\circ$ . A oeste, suas medidas são decrescentes até o limite mínimo de  $-180^\circ$ .

*Paralelos* são círculos da esfera cujos planos são perpendiculares ao eixo dos pólos. O *Equador* é o paralelo que divide a Terra em dois hemisférios: Norte e Sul. O paralelo a  $0^\circ$  corresponde ao Equador,  $90^\circ$  ao Pólo Norte e  $-90^\circ$ , ao Pólo Sul. Todos os meridianos se encontram em ambos os pólos e cruzam o equador em ângulo reto. A distância entre meridianos diminui do Equador para os pólos. Os paralelos jamais se cruzam.

Representa-se um ponto na superfície terrestre por um valor de latitude e longitude. *Longitude* é a distância angular entre um ponto qualquer da superfície terrestre e o meridiano de origem. *Latitude* é a distância angular entre um ponto qualquer da superfície terrestre e a linha do Equador. Pontos que não correspondem à medição média dos oceanos podem ter também a altitude como terceiro parâmetro.

Como o sistema de coordenadas geográficas considera desvios angulares a partir do centro da Terra, não é um sistema conveniente para aplicações em que se buscam distâncias ou áreas. Para estes casos, utilizam-se outros sistemas de coordenadas, mais adequados, como, por exemplo, o sistema de coordenadas planas, descrito a seguir.

O *sistema de coordenadas planas*, também conhecido por *sistema de coordenadas cartesianas*, baseia-se na escolha de dois eixos perpendiculares, usualmente denominados eixos horizontal e vertical, cuja interseção é denominada origem, estabelecida como base para a localização de qualquer ponto do plano.

Nesse sistema de coordenadas, um ponto é representado por dois números: um correspondente à projeção sobre o eixo  $x$  (horizontal), associado principalmente à longitude, e outro correspondente à projeção sobre o eixo  $y$  (vertical), associado principalmente à latitude. Estas coordenadas são relacionadas matematicamente às coordenadas geográficas, de maneira que umas podem ser convertidas nas outras.

### 1.2.3 Projeções Cartográficas

#### Definições Básicas

Todos os mapas são representações aproximadas da superfície terrestre, que projetam cada ponto do globo terrestre em uma superfície plana. Para se obter essa correspondência, utilizam-se os *sistemas de projeções cartográficas*.

Há um grande número de projeções cartográficas, uma vez que há uma variedade de modos de projetar em um plano os objetos geográficos que caracterizam a superfície terrestre. No entanto, é impossível se fazer uma cópia plana da superfície do globo terrestre sem desfigurá-la ou alterá-la – o que dá origem à noção de *grau de deformação de uma projeção*. A deformação é nula nos locais onde a superfície toca o globo. Dependendo do que se pretende analisar no mapa, cada tipo de projeção minimiza um certo tipo de deformação, por exemplo, buscando conservação dos ângulos ou uma proporcionalidade das superfícies.

O *paralelo padrão* é aquele onde as deformações são nulas, isto é, a escala é verdadeira, sendo utilizado como linha de controle no cálculo de uma projeção cartográfica. A partir desse paralelo, as deformações vão aumentando progressivamente sobre os paralelos e sobre os meridianos, com valores desiguais.

A *longitude de origem* é representada por uma linha reta, que constitui o eixo de simetria, no sentido vertical. A definição de longitude de origem depende da projeção adotada. A *latitude de origem* refere-se ao paralelo padrão mais próximo à região de interesse. Dependendo da projeção utilizada, define-se ou não a latitude de origem.

Denomina-se *transformação cartográfica projetiva* a operação de converter dados espaciais representados segundo um determinado sistema de projeção cartográfica para outro sistema de projeção.

## Classificação das Projeções

Os sistemas de projeção cartográfica são analisados segundo o tipo de superfície adotada e grau de deformação.

A superfície que toca o globo terrestre pode ser tangente, secante ou complexa. O caso de *superfície tangente*, apesar de mais simples, acarreta mais distorções que os demais. A *superfície secante* toca o globo ao longo de uma linha. Apesar de mais complexo, acarreta menos distorções pois a superfície de contato é maior. Já uma *superfície complexa* é formada pela combinação de superfícies, o que diminui ainda mais as distorções porém aumenta a complexidade.

Quanto ao tipo de superfície de projeção adotada, classificam-se as projeções em: planas ou azimutais, cônicas, cilíndricas e poliédricas, segundo se represente a superfície curva da Terra sobre um plano, cone, um cilindro ou um poliedro tangente ou secante à esfera terrestre. Há também um grupo de projeções, descritas pelo USGS (“U. S. Geological Survey”), que não podem ser facilmente classificadas e que são denominadas miscelâneas (“miscellaneous”) ou espaciais, onde as últimas foram desenvolvidas para acomodar visadas de satélites.

Na *projeção plana ou azimutal*, constrói-se o mapa imaginando-o projetado num plano tangente ou secante à Terra. Na *projeção cônica*, pode-se imaginar que o mapa é inicialmente projetado sobre um cone tangente ou secante à superfície terrestre, que é em seguida “desenrolado” sobre um plano. O mesmo ocorre na *projeção cilíndrica*, onde a projeção é imaginada sobre um cilindro. Em todas as projeções cônicas, os meridianos são retas que convergem em um ponto (que representa o vértice do cone), e os paralelos são circunferências concêntricas a

esse ponto. Em todas as projeções cilíndricas, os meridianos e os paralelos são representados por retas perpendiculares.

Exemplos destes três tipos de projeções são mostrados nas figuras 1.1 (projeção plana este-reográfica polar), 1.2 (projeção cônica de Lambert) e 1.3 (projeção de Mercator, cilíndrica).

Quanto ao aspecto, a superfície pode ser equatorial, polar, oblíqua ou transversa. O *aspecto equatorial* ocorre quando a superfície de projeção é centrada em algum ponto do Equador. O *aspecto polar* resulta quando a superfície de projeção é centrada em um dos pólos; o *aspecto oblíquo ou horizontal* acontece quando a superfície de projeção é alinhada ao longo de qualquer outro ponto do globo. O *aspecto transversa* ocorre quando a superfície de projeção é cilíndrica e a linha central de projeção é alinhada a um par de meridianos, ao invés do Equador. Por fim um aspecto é dito *normal* se ele é o mais simples para uma dada superfície de projeção, ou seja, polar, no caso de azimutal; oblíquo, no caso de cônica; e equatorial, no caso de cilíndrica.

As figuras 1.4, 1.5 e 1.6 apresentam uma comparação da representação de um quarto de hemisfério, segundo os diferentes sistemas de projeção.

Cada método de projeção da superfície terrestre preserva diferentes propriedades espaciais: área, direção, distância e forma. A preservação de uma propriedade implica normalmente na distorção das demais. Assim, quanto ao grau de deformação das superfícies representadas, as projeções são classificadas em conformes ou isogonais, equivalentes ou isométricas e eqüidistantes.

As projeções *conformes ou isogonais* mantêm fidelidade aos ângulos locais observados na superfície representada (por exemplo, Mercator). As projeções *equivalentes ou isométricas* conservam as relações de superfície, não havendo deformação de área (por exemplo, Cônica de Albers, Azimutal de Lambert). Já as projeções *equidistantes* conservam a proporção entre as distâncias, em determinadas direções, na superfície representada (por exemplo, a Cilíndrica Eqüidistante).

A escolha de uma projeção deve se basear na precisão desejada, no impacto sobre o que se pretende analisar e no tipo de dado disponível. A tabela 1.1, adaptada de [SPR93], apresenta uma análise comparativa das projeções.

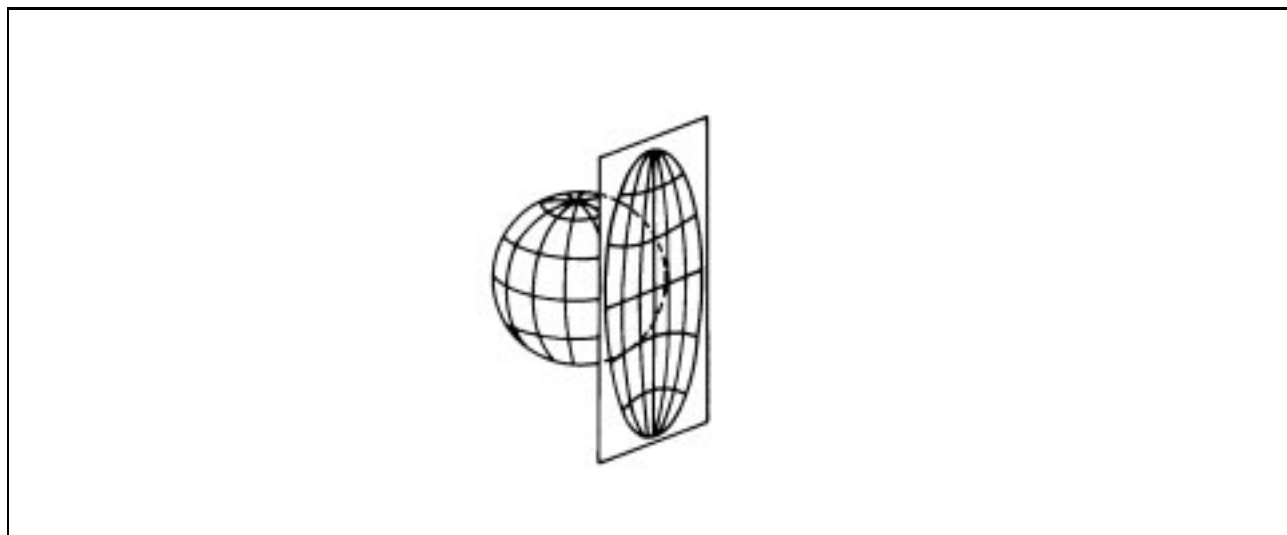


Figura 1.1: Projeção estereográfica polar

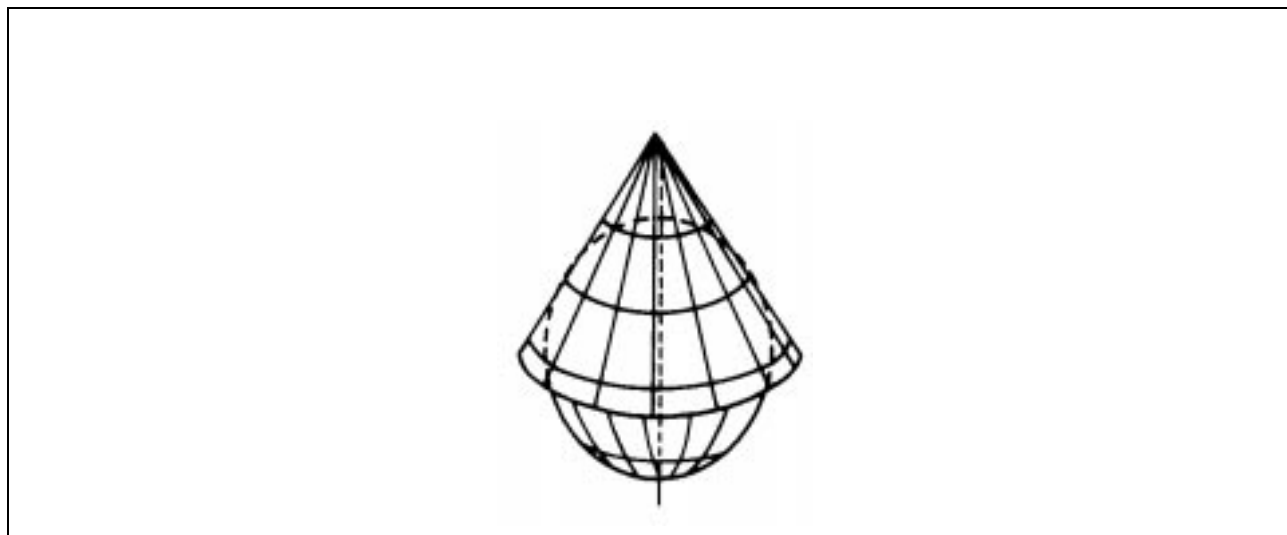


Figura 1.2: Projeção cônica de Lambert

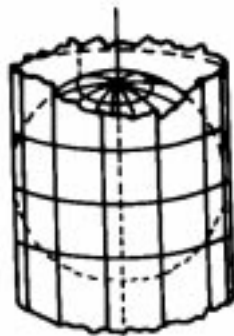


Figura 1.3: Projeção de Mercator

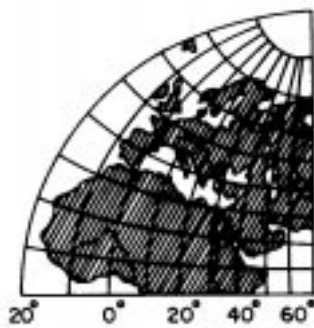


Figura 1.4: Projeção plana



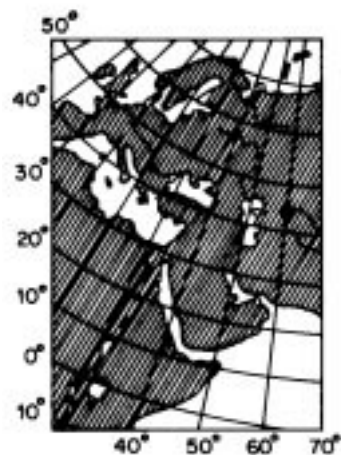


Figura 1.5: Projeção cônica

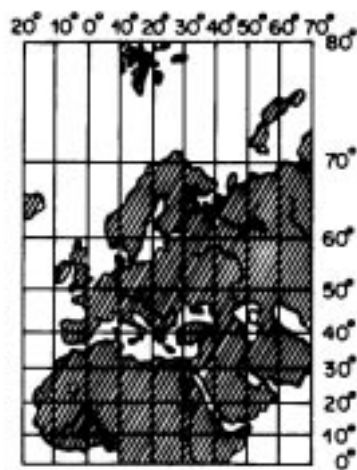


Figura 1.6: Projeção cilíndrica

Projeção	Classificação	Aplicações	Características da projeção
Albers	Cônica Equivalente	- cartas gerais e geográficas	- preserva áreas - garante precisão de escala - substitui com vantagens todas as outras cônicas equivalentes
Bipolar	Cônica Conforme	- indicada para base cartopográfica confiável do continente americano	- preserva ângulos - é uma adaptação da Cônica de Lambert
Cilíndrica Equidistante	Cilíndrica Equidistante	- mapas mundi - mapas em escala pequena - trabalhos computacionais	- altera áreas - altera os ângulos
Gauss	Cilíndrica Conforme	- cartas topográficas - mapeamento básico em escala média e grande	- altera áreas (porém as distorções não ultrapassam 0.5%) - preserva os ângulos - similar à UTM com defasagem de 3° de longitude entre os meridianos centrais
Estereográfica Polar	Plana Conforme	- mapeamento das regiões polares - mapeamento da Lua, Marte e Mercúrio	- preserva ângulos - preserva forma de pequenas áreas - oferece distorção de escalas
Lambert	Cônica Conforme	- cartas gerais e geográficas - cartas militares - cartas aeronáuticas do mundo	- preserva ângulos - mantém a forma de áreas pequenas praticamente inalterada - oferece grande precisão de escala
Lambert Million	Cilíndrica Conforme	- Atlas - cartas ao milionésimo	- preserva ângulos - mantém a forma de áreas pequenas praticamente inalterada - dá grande precisão de escala
Mercator	Cilíndrica Conforme	- cartas náuticas - cartas geológicas/magnéticas - mapas mundi	- preserva os ângulos - mantém a forma de áreas pequenas celestes/meteorológicas
Miller	Cilíndrica Equidistante	- mapas mundi - mapas em escalas pequenas	- altera os ângulos - altera área
Policônica	Cônica Equidistante	- mapeamento temático em escalas pequenas	- preserva distâncias - altera áreas - altera ângulos - substituída por UTM
UTM	Cilíndrica Conforme	- mapeamento básico em escalas médias e grandes - cartas topográficas	- preserva ângulos - altera áreas (porém as distorções não ultrapassam 0.5%)

Tabela 1.1: Análise comparativa das projeções

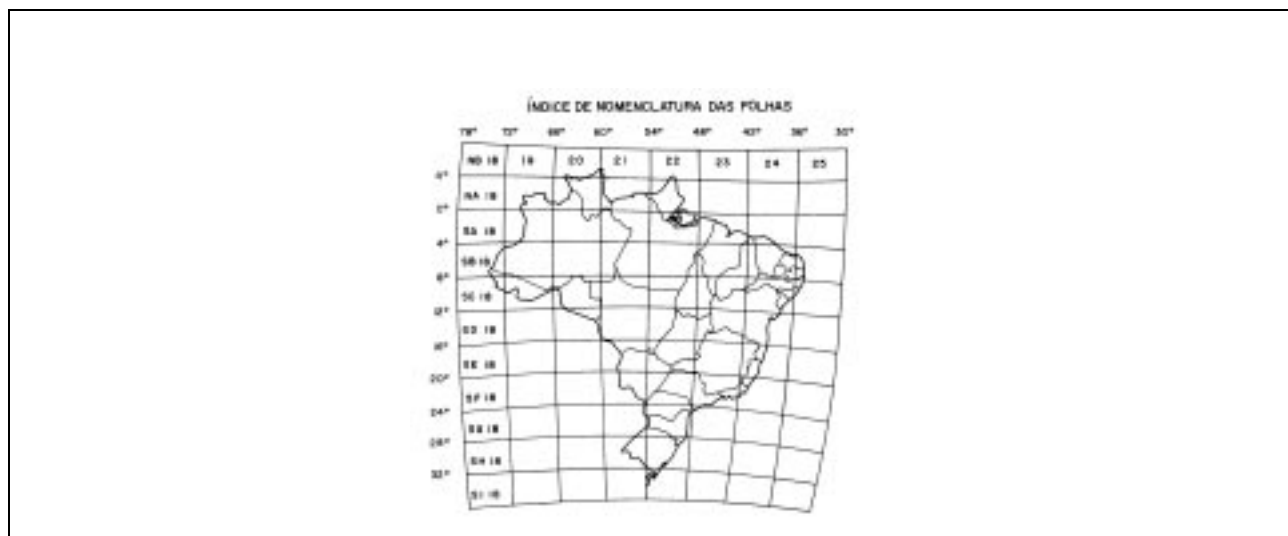


Figura 1.7: Cartas 1:1.000.000 - UTM

## A Projeção UTM

A projeção *UTM* (“Universal Transverse Mercator”) é definida dividindo-se a Terra em 60 fusos de 6° de longitude. Para cada fuso, adota-se como superfície de projeção um cilindro transversal com eixo perpendicular ao seu meridiano central, que assume ainda o papel de longitude de origem. Apesar da característica “universal” desta projeção, enfatiza-se que o elipsóide de referência varia em função da região da superfície terrestre.

O meridiano central de um fuso, o Equador e os meridianos situados a 90° do meridiano central são representados por retas, enquanto que os demais meridianos e os paralelos são representados por curvas complexas. O meridiano central é representado em verdadeira grandeza. A escala aumenta com a distância em relação ao meridiano central e torna-se infinita a 90° deste. Aplica-se ao meridiano central de cada fuso um fator de redução de escala igual a 0,9996, para minimizar as variações de escala dentro do fuso. Como consequência, existem duas linhas aproximadamente retas, uma a leste e outra a oeste, distantes cerca de 1°37' do meridiano central, representadas em verdadeira grandeza.

Há um mapeamento sistemático do Brasil, feito na projeção UTM, nas escalas 1:250.000, 1:100.000 e 1:50.000. A figura 1.7 apresenta a distribuição das cartas 1:1.000.000 para o Brasil. Para saber a longitude de origem, o usuário deve localizar a área de interesse na figura e verificar a que fuso ela pertence. O meridiano central do fuso corresponderá à longitude de origem.

### 1.2.4 Mapas e Cartas

Os conceitos de *mapa* e *carta* não possuem uma distinção rígida [Oli95a]. A palavra “mapa” teve origem na Idade Média e era exclusivamente empregada para designar representações terrestres. Após o século XIV, os mapas marítimos passaram a ser denominados cartas, como por exemplo,

as conhecidas “cartas de marear” dos portugueses. Algumas definições consideram que um “mapa” não tem caráter científico especializado, sendo destinado a fins culturais, ilustrativos ou mesmo comerciais. Já “carta” é a representação dos aspectos naturais ou artificiais da Terra, destinada a fins práticos da atividade humana, permitindo a avaliação precisa de distâncias, direções e a localização geográfica de pontos, áreas e detalhes. Neste texto, os termos “mapa” e “carta” estão utilizados indistintamente, adotando a definição de caráter científico.

Baseado na ABNT (Associação Brasileira de Normas e Técnicas), [Oli95a] classifica cartas como se segue.

- geográficas:
  - topográficas: confeccionadas mediante levantamento topográfico regular, ou compiladas de cartas topográficas existentes, e que incluem os acidentes naturais e artificiais, permitindo facilmente a determinação de altitudes.
  - planimétricas: semelhantes às cartas topográficas, porém não apresentam indicação de altitudes.
- cadastrais e plantas: geralmente em escala grande, usadas para mostrar limites verdadeiros e usos das propriedades.
- aeronáuticas: representam a superfície da Terra com sua cultura e relevo de maneira a satisfazer especificamente as necessidades da navegação aérea.
- náuticas: resultam dos levantamentos dos mares, rios, canais e lagoas navegáveis e se destinam à segurança da navegação.
- especiais, por exemplo:
  - meteorológicas: mostram as classificações climáticas e as que, em serviço contínuo, diário e sistemático, contêm informações meteorológicas, observadas simultaneamente em vários lugares, além das alterações progressivas nas condições do tempo.
  - de solo: identificam e classificam os diversos tipos de solos e sua distribuição geográfica.
  - de vegetação: representam as características e a distribuição da cobertura vegetal.
  - de uso da Terra: representam a classificação e distribuição geográfica dos diversos usos aos quais está sujeita a superfície da Terra.
  - globos: contêm representações da superfície da Terra em outra superfície semelhante.

## 1.3 Conceitos de Sensoriamento Remoto

Esta seção revê brevemente conceitos de sensoriamento remoto, sob o ponto de vista de tecnologia para aquisição de dados para SIGs. Neste sentido, *sensoriamento remoto* é definido como

o conjunto de processos e técnicas usados para medir propriedades eletromagnéticas de uma superfície, ou de um objeto, sem que haja contato entre o objeto e o equipamento sensor.

Existem diversos sistemas de aquisição de dados, tais como câmaras fotográficas aerotransportadas, satélites, sistemas de radar, sonar ou microondas. Os sistemas podem ser *ativos*, como é o caso dos sistemas de microondas, que registram a diferença de frequência entre o sinal emitido por elas e o sinal recebido da superfície (efeito “Doppler”), ou *passivos*, como é o caso de câmaras fotográficas, que registram a reflectância ou emitância de uma superfície.

Esta seção se concentra em dados digitais coletados por câmaras fotográficas aerotransportadas e por sensores a bordo de satélites.

### 1.3.1 Princípios Físicos

*Sistemas de imageamento* são sistemas físicos destinados a produzir ou captar imagens, tais como câmaras fotográficas, câmaras filmadoras, sensores a bordo de satélites e outros [MV89]. A imagem de uma cena é formada através de lentes sobre um plano, onde são colocados sensores que medem a intensidade da luz incidente. No caso da fotografia, por exemplo, o filme fotográfico é colocado no plano dos sensores.

Os sensores respondem proporcionalmente à quantidade de luz incidente, seja através de uma reação física, como no caso do filme fotográfico, ou através de um sinal elétrico (corrente ou tensão), como no caso das células fotoelétricas. A resposta de um sensor depende não só da quantidade de luz como também da frequência da luz. É comum, portanto, descrever a característica de um sensor através de uma curva de resposta espectral, que fornece a intensidade da resposta para cada frequência (ou comprimento de onda). A curva de resposta espectral varia do azul, verde e vermelho na faixa do visível, passando pelo infra-vermelho próximo, médio e termal, e chegando a faixas de microondas.

Existem diversas definições das propriedades básicas de um sensor, levando a diferentes valores. Estas propriedades podem ser resumidas como se segue [DS93], onde algumas são exclusivas para satélites.

- *resolução espectral*, dada pelo número de bandas do espectro eletromagnético capturadas pelos sensores;
- *resolução radiométrica*, dada pela capacidade do sensor de diferenciar níveis de energia do alvo (quantização);
- *resolução espacial*, dada pela área espectralmente confiável da superfície terrestre observada instantaneamente por cada sensor;
- *resolução temporal*, dada pelo intervalo de tempo entre duas tomadas de imagem;
- *campo instantâneo de visão* ou IFOV (“instantaneous field of view”), correspondendo ao ângulo sólido relativo à visada de um elemento de imagem;

- *campo angular de visão*, correspondendo à varredura completa de um sensor;
- *cobertura espectral*, dada pela localização espectral das bandas;
- *função de espalhamento pontual* – PSF (“point spread function”);
- *função de resposta temporal*.

A resolução espacial é função do IFOV, do PSF, de condições atmosféricas, de propriedades da cena (tais como contraste espectral e geometria do objeto observado, geometria do sensor, altitude e processamento do dado). Ela varia também em função do objetivo ao qual o dado se destina, se detecção de um objeto, se identificação ou rotulação de um objeto detectado ou análise de um objeto identificado. Por exemplo, a resolução espacial para análise de um objeto pode ser inferior à resolução visando simplesmente a sua detecção.

A resolução temporal é determinada pela altitude do satélite, pela velocidade angular, pela inclinação de órbita relativa ao Equador, entre outras.

### 1.3.2 Sensores a Bordo de Satélites

Sensores a bordo de satélites permitem a realização de medidas consistentes multi-temporais, relativas a grandes áreas, durante períodos de tempo que chegam a décadas. Muitos sensores são transportados em satélites em órbitas próximas dos pólos e sincronizadas com o sol, pois cruzam sempre o Equador na mesma hora solar, a fim de atingirem uma cobertura global e uma geometria de iluminação consistente. Este é o caso dos satélites das séries Landsat e SPOT e do NOAA/AVHRR. O intervalo de repetição varia entre esses satélites dependendo de sua altitude e velocidade. Outros sensores são transportados em satélites em órbitas geoestacionárias a fim de proverem uma grande frequência de cobertura de uma mesma região. Este é o caso dos satélites meteorológicos GOES.

Esta seção apresenta inicialmente os principais satélites cujos dados são recebidos no Brasil e em seguida discute brevemente sistemas GPS (*Global Positioning System*).

#### Principais satélites no Brasil

Dentre os principais satélites cujos dados são recebidos no Brasil, destacam-se as séries Landsat e SPOT, o TIROS/NOAA, o Meteosat e o ERS. A China e o Brasil pretendem lançar em 1997 o satélite CBERS, com três sensores, similares ao MSS, ao TM e ao AVHRR, a serem discutidos nesta seção.

A geração de imagens a partir satélites varia de diversas formas. A tabela 1.2, adaptada de [Cam95], apresenta características gerais dos principais satélites (com seus sensores) disponíveis no Brasil.

O Landsat e o SPOT são na verdade duas séries de satélites comumente usadas na obtenção de imagens para gerência de recursos terrestres. O TIROS/NOAA é utilizado em gerência

Satélite	Instrumento	Resolução Espectral	Resolução Espacial	Resolução Temporal
LANDSAT	MSS	4	80 m	16 dias
	TM	7	30 m	16 dias
SPOT	XS	3	20 m	26 dias
	PAN	1	10 m	26 dias
TIROS/NOAA	AVHRR	5	1100 m	6 horas
METEOSAT	MSS	4	8000 m	30 minutos
ERS	SAR banda-C	1	25 m	25 dias

Tabela 1.2: Características de satélites cujos dados são recebidos no Brasil

de recursos terrestres e em estudos de vegetação e aplicações meteorológicas. O Meteosat possui aplicações na área de meteorologia. O ERS-1, o primeiro a possuir um radar, destaca-se na gerência de recursos terrestres. Trata-se, no entanto, de um satélite experimental, com cobertura limitada. No restante desta seção, os satélites Landsat e SPOT são discutidos em mais detalhes.

O programa Landsat, inicialmente um programa de pesquisa da NASA, é atualmente operado comercialmente pela firma EOSAT (“The Earth Observation Satellite Company”). Ele inclui cinco satélites diferentes, Landsat-1 a Landsat-5. O satélite Landsat-6 não obteve sucesso ao tentar atingir a sua órbita e o Landsat-7 está programado para 1998. A órbita dos satélites Landsat é próxima dos pólos e sincronizada com o sol. Os Landsat-4 e -5 foram postos em órbitas mais baixas (705km) que os anteriores (920km) a fim de permitir imagens de melhor resolução.

O sensor MSS (“Multi-Spectral Scanner”), presente em toda a série Landsat, possui quatro bandas, do verde ao infra-vermelho próximo. Os Landsat-1 a -3 eram dotados do sensor RBV (“Return Beam Vidicon camera”), semelhante a uma câmara de televisão, que operava três bandas diferentes, do visível ao infra-vermelho próximo. Os Landsat-4 e -5 foram equipados com o sensor TM (*Thematic Mapper*). O TM é um aperfeiçoamento do MSS, adicionando mais bandas e melhorando as resoluções espectral, radiométrica e espacial. Ele possui 7 bandas, do visível ao infra-vermelho termal. O ETM (“Enhanced Thematic Mapper”), que estaria presente no Landsat-6, adicionaria ao TM uma banda pancromática (preto e branco) com resolução de 15m. No caso dos satélites Landsat-4 e -5, a banda TM 6 (infra-vermelha termal) possui resolução de 120m.

O programa SPOT (“Système pour l’Observation de la Terre”) foi iniciado pelo governo francês, e atualmente seu desenvolvimento e operação são de responsabilidade do CNES (“Centre National d’Etudes Spatiales”), incluindo os satélites SPOT-1 a SPOT-3. Os satélites SPOT também possuem órbita sincronizada com o sol e próxima dos pólos, a 832km de altitude.

Os SPOT-1 a -3 são dotados do sensor HRV (“High Resolution Visible”), que registra três canais multiespectrais, do verde ao infra-vermelho próximo, ou um pancromático (preto e bran-

co), de alta resolução. Espera-se que a próxima geração de satélites SPOT possua resolução de 5m em imagens pancromáticas. Uma característica importante do SPOT é a sua capacidade de pivotear a partir de ordens recebidas da Terra, possuindo um espelho móvel, com visada lateral. Isto permite adquirir uma imagem de uma mesma área 7 ou mais vezes (dependendo da posição em termos de latitude) durante o seu ciclo de 26 dias. Devido a estes dispositivos óticos é possível também se obter pares estereoscópicos de imagens (imagens de uma mesma área adquiridas sob ângulos diferentes), o que permite se realizar ortorectificação e obter dados tridimensionais. No processo de ortorectificação, as imagens são completamente corrigidas, eliminando distorções provenientes do posicionamento do sensor e de características da superfície do terreno. Algumas rotinas de ortofotografia exigem valores pré-determinados de sobreposição de imagens.

Enquanto o Landsat TM oferece melhor resolução espectral e radiométrica, o que facilita a análise de determinados fenômenos, o SPOT oferece melhor resolução espacial, especialmente em imagens pancromáticas de 10m, e uma geometria mais estável devido ao sensor utilizado. É muito comum o uso conjugado das imagens do SPOT e do Landsat, usufruindo dos detalhes da banda pancromática do SPOT juntamente com a sensibilidade espectral do Landsat TM.

É importante observar que os satélites estão (ou estavam) operacionais em períodos de tempo distintos (os Landsat-1 a -3 foram descontinuados em 1983) e que nem todos os tipos de produção de imagens estiveram disponíveis durante todo o período operacional dos satélites.

## GPS

*GPS* (“*Global Positioning System*”) é um sistema baseado em satélites que provê a medição de latitude, longitude e altura em qualquer ponto da Terra. Os satélites enviam mensagens específicas que são interpretadas por um receptor GPS. A distância entre o satélite e o receptor GPS pode ser calculada a partir dos sinais enviados e da velocidade do satélite. Para maior precisão, a localização do ponto em questão é determinada utilizando-se pelo menos quatro distâncias e trigonometria. Se 21 satélites estiverem em operação<sup>1</sup>, é possível se medir coordenadas na Terra em qualquer hora do dia.

A tecnologia de GPS foi criada com fins militares. Nos EUA, havia um serviço mais preciso para uso militar, e outro, para uso geral, menos preciso, degradando os sinais enviados pelos satélites. Rapidamente, fornecedores de receptores para GPS transpuseram este problema ajustando o dispositivo a partir de medidas em pontos previamente conhecidos, criando o DGPS (“*Differential GPS*”).

Os receptores GPS variam em precisão e funcionalidade. Por exemplo, alguns incluem programas que fazem transformações entres sistemas de coordenadas ou possuem dados de saída compatíveis com sistemas de SIGs comuns no mercado. Outros dispositivos permitem a leitura com os receptores em movimentos e são muito úteis para realizar mapeamento de terrenos com veículos.

---

<sup>1</sup>Devido à constelação de satélites hoje existente, este requisito não representa um problema.



### 1.3.3 Câmaras Fotográficas Aero-Transportadas

Em um SIG, fotografias aéreas possuem duas funções principais: como componente gráfico, servindo como fundo sobre o qual outras informações são apresentadas; e como fonte de dados, atualizando ou criando novos arquivos em escala grande ou servindo para controle de qualidade de dados já armazenados.

*Fotogrametria* é a ciência de se obter medidas acuradas e confeccionar mapas a partir de fotografias [Wol83]. Um projeto de mapeamento de terrenos utilizando fotografias aéreas inclui o planejamento de rotas para a aquisição das fotografias, a escolha de pontos de controle para georeferenciamento, a digitalização e geração automática dos dados a partir da fotografia.

Fotografias aéreas são obtidas a partir de um avião voando em linhas paralelas, chamadas *linhas de voo*. O planejamento de voo usualmente garante que haja sobreposição de fotos para a cobertura da área estudada. Sobreposições ocorrem lateralmente (na ordem de 15 a 40%) e adiante/atrás (na ordem de 55 a 65%). Terrenos acidentados exigem maior sobreposição. Com a sobreposição adiante/atrás, uma vista estereoscópica é possível, criando para o usuário uma imagem tridimensional.

As fotografias podem ser gravadas em filme (positivos, negativos ou diapositivos), papel, vidro ou em forma digital. As fotografias podem ser preto e branco ou coloridas, em infra-vermelho ou em comprimento de onda visível, onde a escolha é função da sensibilidade espectral do fenômeno geográfico a ser analisado.

Imagens em infra-vermelho foram desenvolvidas originalmente para fins militares, a fim de se distinguir objetos camuflados na vegetação. Atualmente, são muito úteis em estudos envolvendo a identificação de certas vegetações e de vida selvagem. Imagens em preto e branco ou coloridas normais têm funções similares para mapeamento em geral, sendo as primeiras mais baratas, porém as últimas mais adequadas à visão humana para a diferenciação de objetos (o homem é capaz de distinguir em torno de 100 níveis de cinza e por volta de 10.000 cores).

A escala de uma fotografia e sua área de cobertura são funções da distância focal da câmara fotográfica e da altura do voo. Distâncias focais mais curtas e vôos mais altos dão origem a fotos com escalas menores.

*Ortofotografia* é o nome dado a uma fotografia aérea que sofreu ortorretificação. Ortofotos são bastante utilizadas em aplicações de mapeamento (vide, por exemplo, o trabalho descrito em [Fil93]).

# Capítulo 2

## Sistemas de Informação Geográfica

### 2.1 Introdução

*Sistemas de Informação Geográfica* – SIGs – são sistemas automatizados usados para armazenar, analisar e manipular dados geográficos, ou seja, dados que representam objetos e fenômenos em que a localização geográfica é uma característica inerente à informação e indispensável para analisá-la [Aro89, Bul94].

SIGs comportam diferentes tipos de dados e aplicações, em várias áreas do conhecimento. Exemplos são otimização de tráfego, controle cadastral, gerenciamento de serviços de utilidade pública, demografia, cartografia, administração de recursos naturais, monitoramento costeiro, controle de epidemias, planejamento urbano. A utilização de SIGs facilita a integração de dados coletados de fontes heterogêneas, de forma transparente ao usuário final. Os usuários não estão restritos a especialistas em um domínio específico – cientistas, gerentes, técnicos, funcionários de administração de diversos níveis e o público em geral vêm usando tais sistemas com frequência cada vez maior.

Este capítulo conceitua de forma geral o que são Sistemas de Informação Geográfica e traça um breve panorama das suas aplicações, apresentando exemplos genéricos em várias áreas. O texto enfatiza diferenças importantes entre aplicações de áreas diferentes, como a escala de trabalho adotada e as propriedades espacio-temporais consideradas.

Dentre as referências genéricas sobre SIGs merecem destaque especial os dois volumes de *Geographical Information Systems*, editados por D. J. Maguire, M. F. Goodchild e D. W. Rhind [MGR93a], e a série de relatórios técnicos do *National Center for Geographic Information Analysis* - NCGIA [NCG89]. Outros exemplos são as referências [ABC<sup>+</sup>91, Aro89, PM90]. No Brasil, devem ser lembrados os textos [AS93, Par94, TMC92]. As referências [GB90, NCG89, Thi93] contêm direções gerais para pesquisa em SIGs.

## 2.2 Conceitos Básicos

### 2.2.1 Caracterização e Componentes

Devido à sua ampla gama de aplicações, há diferentes formas de se caracterizar SIGs [MGR93a, Cow90, Aro89, SMSE87, Bur86]. Cada tipo de definição prioriza um aspecto distinto. O enfoque de *banco de dados* define SIG como um SGBD não convencional, geográfico, que garante o gerenciamento de dados geográficos. A abordagem *“toolbox”* considera SIG como sendo um conjunto de ferramentas e algoritmos para manipulação de dados geográficos, tal como a produção de mapas. O enfoque *orientado a processos* utiliza o fato de que SIGs são coleções de subsistemas integrados, onde dados espaciais passam por uma seqüência de processos de conversão – coleta, armazenamento e manipulação. Finalmente, diferentes definições priorizam a *aplicação*, caracterizando o sistema segundo o tipo de dado manuseado, ou a *utilização*, tais como sistemas espaciais para apoio à tomada de decisões ou sistemas para análise de dados geográficos.

As definições de SIGs refletem, cada uma à sua maneira, a multiplicidade de usos e visões possíveis desta tecnologia e apontam para uma perspectiva interdisciplinar de sua utilização. A partir destes conceitos, é possível indicar duas importantes características de SIGs. Primeiro, tais sistemas possibilitam a integração, numa única base de dados, de informações geográficas provenientes de fontes diversas tais como dados cartográficos, dados de censo e cadastro urbano e rural, imagens de satélite e modelos numéricos de terreno. Segundo, SIGs oferecem mecanismos para recuperar, manipular e visualizar estes dados, através de algoritmos de manipulação e análise.

Numa visão abrangente, pode-se considerar que um SIG tem os seguintes componentes: interface com usuário; entrada e integração de dados; funções de processamento; visualização e plotagem; e armazenamento e recuperação de dados. A figura 2.1 indica o relacionamento entre estes componentes. Cada sistema, em função de seus objetivos e necessidades, implementa estes componentes de forma distinta, mas todos estão usualmente presentes num SIG. Esta seção descreve brevemente tais componentes, deixando os detalhes para os outros capítulos deste texto.

Historicamente, o primeiro tipo de interface utilizado foi a linguagem de comandos que, mesmo podendo possuir um grande poder expressivo, torna-se complexa à medida em que o sistema cresce em funcionalidade, o que dificulta o seu uso. Para contornar este problema, muitos sistemas dispõem de facilidades para a criação de macro-comandos. A disponibilidade de ambientes computacionais interativos deu origem a interfaces baseadas em menus. Mais fáceis de operar, estas interfaces tendem a ter menor poder expressivo.

Existem quatro formas principais de entrada de dados em SIGs: a entrada de dados via caderneta de campo, a digitalização em mesa, a digitalização ótica e a leitura de dados na forma digital, incluindo a importação de dados em outros formatos. Historicamente, muitos levantamentos (topográficos, parâmetros de solos e outros) utilizam cadernetas de campo para armazenar os resultados, que naturalmente devem ser inseridos no sistema após um controle

de qualidade. Com o advento do GPS (“Global Positioning System”), discutido na seção 1.3.2, tornou-se possível realizar trabalhos de campo com alto grau de precisão e com registro digital direto. A digitalização de mesa ainda é o modo mais utilizado para a entrada de dados a partir de mapas existentes. É um processo usualmente custoso e demorado, envolvendo os passos de digitalização de linhas, ajuste de nós, geração de topologia e rotulação (identificação) de cada dado geográfico. A digitalização ótica é feita por dispositivos de varredura (*scanners*) e vem sendo cada vez mais utilizada. A tecnologia mais comum é baseada em câmaras CCD (*charge coupled devices*) sendo necessário o uso de dispositivos de boa resolução para se conseguir resultados de qualidade. Após a obtenção de dados em varredura são utilizados algoritmos de conversão para o formato vetorial, sendo que muitos destes requerem a intervenção humana parcial. Na importação de dados digitais, é muito importante aproveitar o investimento já feito, eventualmente por outras instituições, na coleta e armazenamento de dados geográficos. No Brasil, as principais fontes de dados são as bases do IBGE, do INPE e do Centro de Cartografia Automatizada do Exército.

As funções de processamento são naturalmente dependentes dos tipo de dados envolvidos. A análise geográfica engloba funções como superposição, ponderação, medidas (área, perímetro), mapas de distância, tabulação cruzada, dentre outras. O processamento digital de imagens envolve funções como retificação, contraste, filtragem, realce e classificação. Modelos numéricos de terreno permitem a geração de mapas de declividade e aspecto, cálculo de volumes, análise de perfis, além da própria geração do modelo a partir de pontos esparsos ou linhas, entre outras funções. Operações sobre redes incluem caminhos ótimos, caminhos críticos e ligação topológica. Já as consultas aos bancos de dados podem ser espaciais ou não.

Os ambientes de visualização de um sistema são consequência do paradigma adotado para a interface. Quanto à produção cartográfica, alguns sistemas dispõem de recursos altamente sofisticados de apresentação gráfica, englobando a definição de uma área de plotagem, colocação de legendas, textos explicativos e notas de crédito. Já o aparecimento de padrões “de facto”, como o PostScript e o HPGL, vem facilitar o desenvolvimento de funções de plotagem.

Os dados de um SIG são geralmente organizados sob a forma de um banco de dados geográficos. Tradicionalmente, os SIGs armazenavam os dados geográficos em arquivos internos. Este tipo de solução vem sendo substituída pelo uso cada vez maior de SGBD.

Estes componentes se relacionam de forma hierárquica. No nível mais próximo ao usuário, a interface homem-máquina define como o sistema é operado e controlado. No nível intermediário, um SIG deve ter mecanismos de processamento de dados espaciais (entrada, edição, análise, visualização e saída). No nível mais interno do sistema, um sistema de gerência de bancos de dados geográficos oferece armazenamento e recuperação dos dados espaciais e seus atributos.

De uma forma geral, as funções de processamento de um SIG operam sobre dados em uma área de trabalho definida pelo usuário. A ligação entre os dados geográficos e as funções de processamento do SIG é feita por mecanismos de seleção e consulta que definem restrições sobre o conjunto de dados, que podem ser espaciais ou não.

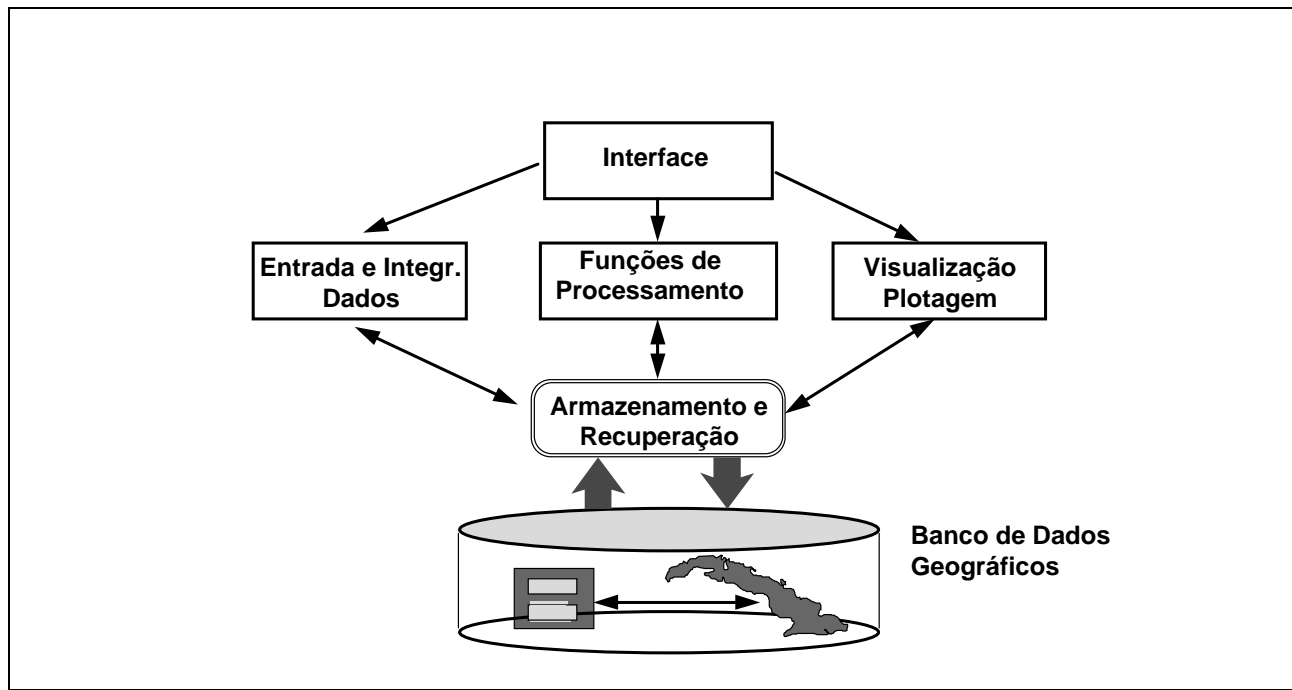


Figura 2.1: Arquitetura de Sistemas de Informação Geográfica

### 2.2.2 Processo de Implantação

Sob certo ponto de vista, o processo de implantação de um SIG divide-se em três grandes fases: modelagem do mundo real; criação do banco de dados geográfico; e operação.

A fase de modelagem do mundo real engloba a modelagem de processos e de dados e consiste em selecionar fenômenos e entidades de interesse, abstraindo-os e generalizando-os. Diferentes conjuntos de fenômenos podem ser escolhidos para descrever distintas visões do mundo, para uma mesma região, em um dado instante.

Um banco de dados geográfico é um repositório da informação coletada empiricamente sobre os fenômenos do mundo real [ea92, Ege95]. A criação de um banco de dados geográficos exige várias etapas: coleta dos dados relativos aos fenômenos de interesse identificados na modelagem; correção dos dados coletados (devido, por exemplo, a erros introduzidos pelos dispositivos de coleta); e georeferenciamento dos dados (associando a cada conjunto de dados informação sobre sua localização geográfica). Esta fase representa uma grande parcela do custo total do desenvolvimento de um SIG, que pode ser minimizado por uma modelagem adequada.

A fase de operação refere-se tanto ao uso em si do SIG, quanto ao desenvolvimento de aplicações específicas por parte dos usuários a partir dos dados armazenados, reconstruindo visões (particulares) da realidade.

Retornando à primeira fase, *modelagem de processos* refere-se a uma modelagem matemática que descreve operações envolvendo a representação e manipulação de dados, incluindo a simulação de fenômenos naturais [BX94b]. Este tipo de modelagem começa com a seleção dos

fenômenos e de um modelo (matemático) que permita descrevê-los e simulá-los, definindo os dados a coletar. Em seguida, dados e modelo matemático são calibrados e refinados, em um processo iterativo, até atingir um nível determinado de qualidade e adequação [BX94a]. Os procedimentos de simulação numérica variam com a natureza da aplicação e com a extensão e escala do fenômeno observado [GPS93].

Um modelo de dados fornece ferramentas formais para descrever a organização lógica de um banco de dados, bem como define as operações de manipulação de dados permitidas. *Modelagem de dados* refere-se ao processo de abstrair os fenômenos do mundo real para criar a organização lógica do banco de dados. No caso de aplicações geográficas, as técnicas tradicionais de modelagem devem ser estendidas para incluir questões específicas de dados geográficos.

Os primeiros trabalhos sobre modelos de dados geográficos se ocupavam principalmente com estruturas geométricas e espaciais. Os modelos propostos correspondiam a estruturas de dados sofisticadas (vide, por exemplo, a descrição de alguns destes “modelos” em [Aro89]). Esta concepção de modelo foi incorporada pela maioria dos sistemas comerciais atuais, onde o usuário realiza a “modelagem” dos dados definindo diretamente estruturas de baixo nível. Esta filosofia é refletida para os níveis mais altos de definição de dados, impedindo muitas vezes os usuários de especificar as entidades de interesse. Entretanto, esta não é uma abordagem apropriada para modelagem de dados geográficos pois os usuários raramente são especialistas em computação, mas sim nos diferentes domínios de aplicação. Além deste ponto básico, esta abordagem não leva a modelos úteis à fase inicial de coleta e pré-processamento dos dados, nem a modelos que facilitem a reutilização dos dados por outras aplicações. O capítulo 4 apresenta abordagens mais adequadas que minimizam estes problemas.

### 2.2.3 Breve Histórico

O problema de análise e manipulação de entidades que existem em um contexto espaço-temporal não é novo. A forma mais antiga – e ainda mais comum – de processar e apresentar este tipo de dado é através de mapas. Os elementos de um mapa são em geral armazenados de forma georeferenciada segundo um sistema de coordenadas (latitude, longitude e elevação em relação ao nível do mar). A criação de mapas exige levantamento de dados, medidas de seus valores e localização, padronização, armazenamento e finalmente apresentação. Diversos mapas em papel podem ser processados de forma a correlacionar seus dados através de sua sobreposição sob forma de folhas transparentes. Todo esse processo – produção em papel, armazenamento, sobreposição – é muito caro do ponto de vista de armazenamento e atualização manual.

Desta forma, as primeiras tentativas de automatizar o processamento de dados georeferenciados ocorreram, segundo [ABC<sup>+</sup>91], nos anos 50, na Grã Bretanha e nos USA, visando diminuir os custos de produção e atualização de mapas. No primeiro caso, foi desenvolvido um sistema de produção de mapas para pesquisa em botânica, onde os dados haviam sido previamente perfurados em cartões. No segundo caso, tratava-se de um sistema desenvolvido em Chicago para mostrar graficamente o volume de tráfego em algumas vias da cidade.

Os primeiros SIGs propriamente ditos datam dos anos 60. Seu desenvolvimento, no Cana-

dá, fez parte de um plano estratégico governamental de longo prazo para criar um inventário automatizado de recursos naturais. Os principais objetivos eram o planejamento de recursos naturais e o uso do solo.

Durante os anos 70, desenvolveram-se fundamentos matemáticos voltados para a cartografia. Surgiu, então, a topologia aplicada, permitindo análises espaciais entre elementos cartográficos.

Até então, apenas grandes organizações utilizavam SIGs em sistemas de grande porte. Segundo [Tom94], a maioria das aplicações estava voltada ao mapeamento digital, com funções analíticas. Nos anos 80, com a popularização e barateamento das estações de trabalho, computadores pessoais e bancos de dados, o uso de SIG foi difundido com a incorporação de muitas funções de análise espacial.

Se o progresso tecnológico, por um lado, facilitou o desenvolvimento de SIGs, também permitiu aperfeiçoar os mecanismos de aquisição de dados georeferenciados. Com isto, aumentou a complexidade de coleta, armazenamento, manipulação e visualização dos dados, em função do seu volume, variedade e heterogeneidade.

Atualmente, as aplicações de SIGs variam na extensão da área geográfica considerada (que pode abranger desde um quarteirão em uma cidade até o globo terrestre); equipamento utilizado (desde um computador pessoal até super-computadores); e abrangência (de interesse particular até patrocínio de agências governamentais abrangendo diferentes países). Recentemente, aplicações SIG começaram a incorporar novas tecnologias tais como sistemas especialistas [JEM93] e técnicas de orientação a objetos [Wor94a, AA93, GR93, MMS93, DRS93a, SA93, Her92, CSF92].

A referência [Cam95] considera a existência de três gerações de sistemas de informação geográfica, resumidos na figura 2.1.

A primeira geração, baseada em CAD cartográfico, caracteriza-se por sistemas herdeiros da tradição de Cartografia, com suporte de bancos de dados limitado e cujo paradigma típico de trabalho é o mapa (chamado de “cobertura” ou de “plano de informação”). Esta classe de sistemas é utilizada principalmente em projetos isolados, sem a preocupação de gerar arquivos digitais de dados. Esta geração também pode ser caracterizada como sistemas orientados a projeto (“project-oriented GIS”).

A segunda geração de SIGs, baseada em bancos de dados geográficos, chegou ao mercado no início da década de 90 e caracteriza-se por ser concebida para uso em ambientes cliente-servidor, acoplado a gerenciadores de bancos de dados relacionais e com pacotes adicionais para processamento de imagens. Desenvolvida em ambientes multiplataforma com interfaces em janelas, esta geração também pode ser vista como sistemas para suporte a instituições (“enterprise-oriented GIS”). O capítulo 11 discute o sistema SPRING, que pode ser enquadrado nesta geração.

Pode-se prever, para o final da década de 90, o aparecimento de uma terceira geração de SIGs, baseada em bibliotecas digitais geográficas ou centros de dados geográficos (ver capítulo 13), caracterizada pelo gerenciamento de grandes bases de dados geográficos, com acesso através de redes locais e remotas, públicas ou privadas. Para esta terceira geração, o crescimento dos bancos de dados geográficos e a necessidade de seu compartilhamento com outras instituições requer o recurso de tecnologias como bancos de dados distribuídos e federativos. Estes

sistemas deverão seguir os requisitos de interoperabilidade, de maneira a permitir o acesso de informações espaciais por SIGs distintos. A terceira geração de SIG pode ainda ser vista como o desenvolvimento de sistemas orientados à troca de informações entre uma instituição e os demais componentes da sociedade (“society-oriented GIS”).

	1ª geração (1980-1990)	2ª geração (1990-1997)	3ª geração (1997 - ?)
tecnologia	CAD, cartografia	BD, imagens	sist. distribuídos
uso principal	desenho de mapas	análise espacial	centro de dados
ambiente	projetos isolados	cliente-servidor	multi-servidores
sistemas	pacotes separados	sistema integrado	interoperabilidade

Tabela 2.1: Evolução da tecnologia de SIGs

## 2.3 Aplicações

### 2.3.1 Classificação das Aplicações

O domínio de aplicações em SIG está se ampliando cada vez mais, acompanhando a evolução dos dispositivos de coleta e as facilidades computacionais em geral.

Um fenômeno geográfico pode ser analisado de forma e precisão diferentes dependendo do objetivo da aplicação. Assim sendo, um mesmo conjunto de dados armazenados poderá ter tratamentos distintos. Por exemplo, dados pluviométricos coletados em um conjunto de estações pluviométricas ao longo do tempo podem ser interpolados para determinar a distribuição de chuvas em certas regiões durante um período, visando planejar atividades de plantio. Os mesmos dados podem ser combinados com informações sobre a bacia hidrográfica das regiões, declividade e permeabilidade do solo, para auxiliar estudos de erosão. Esta característica causa um impacto direto na coleta, modelagem e armazenamento dos dados georeferenciados.

Por outro lado, cada aplicação requer a manipulação de fenômenos geográficos distintos, associados a diferentes características e propriedades que variam no espaço e no tempo. Além disso, os usuários SIG têm também uma grande variedade de perfis, como cientistas especialistas em um determinado domínio do conhecimento (por exemplo, biólogos, geólogos, sociólogos), técnicos (engenheiros, arquitetos) ou especialistas em administração e planejamento urbano.

Em função desta amplitude de perfis de usuários, tipos de dados e necessidades das aplicações, SIG também precisam prover aos usuários e projetistas de aplicações um conjunto adequado de funções de análise e manipulação dos dados geográficos. A partir de tais necessidades, surgem diferentes especializações do termo SIG como LIS (“Land Information Systems”), NRIS (“Natural Resource Information System”), SIS (“Soil Information Systems”).

[MGR93b] classifica as aplicações em: *sócio-econômicas*, envolvendo o uso da terra, seres humanos e a infraestrutura existente; *ambientais*, enfocando o meio ambiente e o uso de recursos



naturais; e de *gerenciamento*, envolvendo a realização de estudos e projeções que determinam onde e como alocar recursos para remediar problemas ou garantir a preservação de determinadas características. O capítulo corrente está baseado nesta classificação.

Em princípio, a classe de aplicações determina a utilização de certas escalas, fontes de dados e, certamente, funções de análise. Aplicações sócio-econômicas são geralmente voltadas para escalas grandes (1:200 a 1:20.000), ocupando-se de problemas localizados. Aplicações ambientais, por sua vez, são em geral relacionadas a problemas em escalas menores (1:20.000 ou menores), com conseqüente perda de precisão de medida. Esta regra de escala, fonte e precisão dos dados nem sempre é aplicável: estudos ambientais podem se ocupar de regiões de pequena extensão, exigindo escalas maiores; estudos sócio-econômicos podem abranger migrações populacionais, em escala global.

Em alguns casos, torna-se difícil determinar a classe de uma aplicação: por exemplo, o estudo do impacto ambiental devido a correntes migratórias envolve tanto fatores sócio-econômicos quanto físicos.

### 2.3.2 Aplicações Sócio-Econômicas

Aplicações sócio-econômicas tanto podem ser realizadas com o objetivo de planejamento (análise preliminar) quanto de avaliação de mudanças em uma região em resposta a uma determinada política (análise posterior). Dentre as aplicações sócio-econômicas, segundo [Ram94], distinguem-se os grupos de origem: *uso da terra*, incluindo cadastros rurais, agroindústria e irrigação; *ocupação humana*, envolvendo cadastros urbanos e regionais, sistemas para serviços de utilidade pública; e *atividades econômicas*, agrupando marketing e indústrias.

Enquadram-se na classe de aplicações sócio-econômicas os sistemas de informação sobre uso da terra e os sistemas para mapeamento automático/gerência de facilidades. Exemplos típicos são o acompanhamento e inventário de cadastros imobiliários rurais ou urbanos; definição de uma política para uso de solo; aplicações envolvendo serviços de utilidade pública (redes de telefonia, eletricidade, esgotos, transportes); sistemas de auxílio à navegação; estudos de marketing; e alocação de recursos em geral para manutenção ou expansão da infraestrutura de uma região.

Os dados utilizados em aplicações sócio-econômicas são freqüentemente obtidos através de coletas censitárias, mapas urbanos digitalizados e fotografias aéreas. Aplicações que utilizam dados de censo, por exemplo, preocupam-se com distribuições estatísticas de populações em um certo espaço, associadas à infraestrutura existente. Neste caso, escalas menores são aceitáveis, dependendo da extensão da região estudada. Aplicações voltadas a planejamento de infraestrutura (por exemplo, rede de telefonia ou tráfego) exigem escalas de maior precisão: a colocação de um transformador em um certo ponto em uma rede elétrica deve ter precisão de metros; a definição de uma rota de ônibus urbano precisa levar em consideração a largura das ruas para prever pontos de engarrafamento e locais de manobra.

Segue uma discussão em mais detalhes alguns tipos de aplicações sócio-econômicas.

## Sistemas de Informações sobre Terra

Um sistema de informação sobre uso da terra [Dal93], ou LIS, abreviação de “Land Information System”, manipula basicamente limites de propriedades ou regiões, com mapas e descrições associados, contendo valor (de venda, aluguel, transferência, impostos, etc...), uso (rural ou urbano, vegetação, etc...), construções, infraestrutura (água, gás, eletricidade, etc...), população e outros. Os dados em um LIS devem ser os mais atualizados e completos possíveis, devido às consequências sociais e econômicas que um registro mal feito acarreta, como disputas de terras e cobrança incorreta de impostos. O custo de levantamento destes dados é bastante alto, chegando até a superar o valor da terra em casos extremos, e o processo é muitas vezes lento.

Entre os benefícios advindos do uso de um LIS, têm-se a redução dos casos de disputa de propriedades, a simplificação do processo de reforma agrária, a monitoração do mercado de terras e uma maior eficiência na cobrança de impostos.

Por exemplo, o estudo em [ZFS89] avalia um conjunto de mudanças em uma região rural nos EUA. Dados de questionários respondidos pelas comunidades da região foram inseridos em um SIG e analisados em conjunto com fatores como mudanças no uso de solo e na vegetação, obtidos a partir de fotos aéreas. A avaliação destes resultados permitiu a definição de novos critérios para expansão imobiliária na região.

## Sistemas para Serviços de Utilidade Pública

Existe um crescente uso de SIGs pelos serviços de utilidade pública no mundo, desde ferramentas mais simples para mapeamento automático e gerência de facilidades (“AM/FM - Automated Mapping and Facilities Management”) até sistemas sofisticados, envolvendo simulação.

A natureza dos serviços de utilidade pública varia enormemente de país para país, refletindo-se nos SIGs utilizados. De modo geral, a indústria provê serviços para consumidores comerciais e domésticos, fazendo uso de alguma forma de mapeamento de sua rede. SIGs permitiram que estes registros geográficos fossem relacionados a dados alfanuméricos. O tipo de planta, a complexidade da rede e a sua taxa de crescimento e atualização influenciam a funcionalidade do SIG. Muitas plantas não possuem uma localização precisa de objetos, o que pode acarretar complicações, caso seja necessário sobrepor mapas.

Mahoney [Mah93] subdivide as aplicações relacionadas a serviços de utilidade pública em *redes de fluxo*, incluindo gás e água, e *redes de cabos*, incluindo eletricidade, telecomunicações e televisão a cabo. Uma outra forma de classificar tais aplicações consiste em distinguir entre redes subterrâneas e redes de superfície.

Os objetos de uma rede de fluxo correspondem geralmente a dutos, possuindo atributos como diâmetro, material de construção e idade de um duto. É essencial que exista conectividade nas ligações da rede no banco de dados para a análise da rede de fluxo. As redes de cabos são usualmente inferiores em tamanho, quando comparadas com as redes de fluxo. Registros de eletricidade são em geral separados por voltagem de transmissão e distribuição. Em telecomunicações, a rede deve manter registro da localização dos dutos físicos em redes subterrâneas,

permitindo identificar até consumidores individuais. Em termos de planejamento, é necessário saber a disponibilidade de espaço nos dutos. Em televisão a cabo, os serviços são uma combinação de serviços para redes subterrâneas e redes de superfície.

O georeferenciamento dos registros de serviços de utilidade pública pode ser considerado de duas formas: através de um mapa básico topográfico, que funciona como um arcabouço espacial onde os dados são georeferenciados; ou a partir da localização dos dutos de fluxo ou cabos. O custo de aquisição de um mapa básico, a partir de um levantamento fotográfico aéreo, por exemplo, é geralmente bastante alto, chegando de 10 a 25% do custo total do projeto.

Usualmente, a altura é mantida nos próprios mapas, evitando a necessidade de se lidar com dados tridimensionais, como mapas de topografia, exceto quando operações de colocação de dutos são necessárias. A escala de trabalho varia de 1:200 a 1:10.000 nas áreas urbanas, admitindo-se escalas menores fora destas. As plantas geográfica e de registros podem ser armazenadas em um SIG em uma estrutura temporal, representando o presente, o passado e o planejado.

## Sistemas de Censo

Sistemas de censo ajudam a monitorar mudanças nas características da população e são importantes para planejamento tanto a nível global quanto local.

Existem vários métodos de se realizar censos. Censos exaustivos são caros e usualmente feitos em intervalos de dez anos, quando uma série de perguntas relativas à vida dos habitantes são postas (renda, acesso a telefone, eletrodomésticos, etc...). Naturalmente, análises estatísticas derivadas durante o intervalo entre censos podem se tornar irreais devido à desatualização dos dados. Tradicionalmente, o papel de SIGs é grande no estágio de pós-processamento das informações, onde dados são analisados e facilmente espacializados gerando mapas.

Alguns países, como os escandinavos, mantêm seus dados constantemente atualizados, agregando informação de indivíduos, inclusive espacial, via endereço postal, em áreas relevantes administrativamente. Para isso, fontes de informação (nascimentos, casamentos, mortes, mudanças de casa) são devidamente registradas pelas autoridades.

Uma terceira forma de se obter dados de censo explora a relação entre extensão de área ocupada e população, com a ajuda de cobertura por imagens de satélite e fotografias aéreas, detectando-se manchas urbanas, e utilizando SIGs. Esta técnica não fornece informações sócio-econômicas e é ineficiente para áreas rurais.

Merece destaque o sistema TIGER (“Topologically Integrated Geographic Encoding and Referencing”), do Census Bureau do EUA, que provê a mais detalhada rede de características e cobertura administrativa disponível no país [Rhi93].

## Outras Aplicações Sócio-Econômicas

Sistemas de navegação, marítima ou terrestre, também podem se beneficiar bastante de SIGs.

A área de navegação terrestre inclui sistemas experimentais de tráfego, reportando engarrafamentos, sistemas de monitoramento de frotas e sistemas de navegação para automóveis [Whi93]. No último caso, os SIGs podem agregar funcionalidade de sistemas hipermídia [Cas93], oferecendo facilidades como apresentação de mapas, direções verbais, caminhos, procura de destinos por endereço e indicação de infraestruturas existentes.

Na área de navegação marítima, o desenvolvimento de cartas náuticas digitais apresenta-se como um caminho natural para otimizar o uso dos recursos das embarcações [Oli95a]. De fato, o aumento do tráfego marítimo em determinadas regiões do planeta, aliado às condições atmosféricas e ao estado do mar, demanda cada vez mais o desenvolvimento de sistemas de auxílio à navegação que minimizem a ocorrência de acidentes que possam, por exemplo, resultar na destruição da natureza através do derramamento de imensas cargas de óleo ou elementos químicos. As cartas náuticas são fundamentais como instrumento de segurança da navegação, além de fonte de dados para pesquisa e análise dos ambientes por elas representados.

Outra aplicação sócio-econômica para SIGs são sistemas de informações de mercado. Análises de mercado necessitam muitas vezes de informações espacialmente diferenciadas sobre consumidores e competidores. Sistemas de informações de mercado beneficiam-se das funcionalidades de SIGs principalmente quando estes estão acoplados a ferramentas para análise financeira, sistemas de apoio à decisão e sistemas especialistas.

### 2.3.3 Aplicações Ambientais

Algumas aplicações ambientais, conforme ressaltado por [Fed93], precedem o uso de computadores. A modelagem espacial de fenômenos hidrológicos, por exemplo, começou há mais de cem anos e o uso de técnicas analíticas em problemas biológicos e ecológicos foi iniciado nos anos 20. SIGs vieram agilizar este processamento.

As aplicações ambientais variam enormemente de área. Baseado em [Ram94], distinguem-se dois grupos de origem: o *meio ambiente*, incluindo ecologia, clima, gerenciamento florestal e poluição; e o *uso dos recursos naturais*, envolvendo extrativismo vegetal, extrativismo mineral, energia, recursos hídricos e oceânica.

Enquadram-se em aplicações de meio ambiente modelagem climática e ambiental, previsão numérica do tempo, monitoração do desflorestamento e monitoração da emissão e ação de poluentes. Ilustrando tipos aplicações de uso de recursos naturais têm-se identificação e mapeamento mineral e petrolífero, planejamento e supervisão de redes hidroelétricas, gerenciamento costeiro e marítimo, e sistemas de informação de recursos hidrológicos.

As aplicações podem ter abrangência em uma escala global, continental, regional ou até mesmo local. Aplicações ambientais são em geral relacionadas a problemas em escalas menores que as sócio-econômicas (1:20.000 ou menores), com conseqüente perda de precisão de medida. Os dados são principalmente obtidos através de sensoriamento remoto, como fotos de satélite ou imagens de radar, complementados por amostras coletadas no campo. Assim, por exemplo, a distribuição da vegetação em uma região não se ocupa de plantas individuais, mas das espécies

dominantes.

Segue uma discussão em mais detalhes sobre alguns tipos de aplicações ambientais.

### Sistemas de Informação de Solos

SIGs auxiliam sistemas de informação de solos na catalogação de perfis de solo coletados, no mapeamento através de interpolação de dados gerando variações espaciais, e no uso dessas informações para análise e modelagem [Bur93]. O uso de sistemas de informação de solos não está limitado apenas a grandes catálogos nacionais, mas aplica-se também a estudos locais, tais como estudos voltados para a agricultura em fazendas.

A primeira fase do desenvolvimento de um sistema de informação de solos consiste da catalogação de perfis. Perfis de solos são difíceis de registrar pois amostras são coletadas nos diversos horizontes que ocorrem na natureza. Porém, estes horizontes não são nem identificados, nem classificados de forma clara. Assim, o número de horizontes pode variar enormemente, mesmo em uma única unidade de mapeamento. Sistemas voltados para catálogo de dados de solos estão entre os primeiros usos de SIGs, trazendo como benefício o desenvolvimento de padrões para catalogação.

A segunda fase do desenvolvimento de um sistema de informação de solos consiste na definição de métodos rigorosos para mapeamento. Embora o fenômeno da variação espacial contínua de propriedades de solos seja conhecido há bastante tempo, somente no final da década de 70 é que os cientistas de solos passaram a dedicar um esforço maior a técnicas de interpolação, incluindo métodos geoestatísticos. A distribuição espacial de fenômenos geológicos, pontuais em uma escala regional, tais como depósitos de minerais e epicentros sísmicos, pode parcialmente ser explicada e prevista a partir de mapas e imagens mostrando variáveis geológicas, geofísicas e geotécnicas.

A terceira fase de quantificação de informações de solos inclui, em diferentes direções, o uso da informação para análise e modelagem e o uso de sistemas especialistas.

### Estudo Ambiental de Mudanças Globais

Atualmente, um grande desafio mundial é o programa de *mudanças globais*. Ele parte do pressuposto de que mudanças climáticas, de fauna, flora e recursos naturais só podem ser explicados em uma escala global, de forma interdisciplinar, envolvendo biólogos, ecologistas, geólogos, geofísicos e outros cientistas. Neste contexto, SIGs desempenham um papel fundamental. Com estes novos estudos interdisciplinares, que consideram a Terra como um sistema único completo, há necessidade da revisão de várias funções de um SIG, desde o simples tratamento de projeções cartográficas e problemas de representação até questões conceituais em modelagem de sistemas.

Este esforço mundial motivou uma coleta de dados em todo o planeta, com a cooperação de várias nações. Bancos de dados de dados temáticos globais estão sendo criados e integrados. Os dados globais podem ser divididos em três categorias: *dados globais de referência*, de longa

duração, com escala temporal de décadas a séculos (por exemplo, parâmetros climáticos, de solos, de densidade populacional); *dados globais sinópticos para monitoração*, de média duração, com escala temporal de meses a anos, refletindo mudanças globais em uma escala espacial e temporal (por exemplo, cobertura de neve, ventos e precipitação); e *dados globais tipo séries temporais para monitoração*, consistindo de séries temporais, índices de medidas ou médias para células (por exemplo, índice de vegetação e temperatura oceânica).

Existem vários programas globais que merecem destaque [CHK93], três das quais são apresentadas a seguir.

A NASA possui uma iniciativa chamada “Missão do Planeta Terra”, que visa construir modelos do sistema terrestre de forma a permitir gerenciamento do clima global. É provavelmente o maior e mais completo dos programas globais. Os dados estão sendo coletados para diferentes regiões e em épocas distintas, sendo integrados de forma a permitir acesso em escala global por parte de cientistas, que construirão modelos matemáticos para avaliar a evolução dos diversos fenômenos de interesse.

O programa inclui a criação de uma infraestrutura de gerenciamento de dados georeferenciados, permitindo a usuários de diferentes perfis cooperarem em suas pesquisas relativas ao meio ambiente. Esta infraestrutura, chamada EOSDIS (“Earth Observation System Data and Information System”), será discutida em mais detalhe no capítulo 13.

O “Programa de Mudanças Globais e Climáticas” do NOAA tem o objetivo de prover previsões confiáveis sobre mudanças climáticas globais e suas implicações regionais em uma escala de tempo que varia de meses a séculos. Para isso, são coletadas imagens de satélites meteorológicos, dados de estações terrestres, entre outros.

O ICSU (“International Council of Scientific Unions”), integrado por várias nações, possui o “Programa Internacional de Geosfera-Biosfera”, que visa entender e descrever os processos físicos, interativos e biológicos que regulam o sistema terrestre, o ambiente único que provê a vida, as mudanças que estão ocorrendo neste sistema, e o modo como elas são influenciadas pelas ações humanas.

## Outras Aplicações Ambientais

Esta seção referencia diversas aplicações ambientais, ilustrando a variedade existente.

CORINE é um programa envolvendo 12 países da Comunidade Européia com o objetivo de desenvolver um banco de dados ambiental em escala continental. Sua criação foi uma reação aos problemas de chuvas ácidas, conservação da natureza e conflitos relativos ao uso da terra no Mediterrâneo. [Mou93] discute o programa desde a sua criação, com as fases de análise de requisitos e entrada de dados, até os passos futuros.

[WSS90] descreve vários projetos ambientais que utilizam SIGs nos EUA. Um exemplo é o estudo do impacto que mudanças de zoneamento trouxeram à qualidade da água em uma região, utilizando dados coletados durante 37 anos. Um segundo projeto é o uso de um SIG para determinar o traçado de estradas que atravessem parques nacionais, levando em consideração o

gradiente dos terrenos, o número e tipo dos cursos d'água a serem cortados e a visibilidade do terreno. Outros exemplos incluem a adequação de áreas para criação de espécies em extinção, a previsão de mudanças naturais e o gerenciamento e monitoração de florestas nativas.

O projeto apresentado em [Kli94] discute a avaliação de diversas regiões com relação à sua adequação para a criação de parques de vida selvagem na Nova Zelândia. As diferentes características de hidrografia, facilidades de transporte, infraestrutura e vegetação de cada região foram correlacionadas a fatores como quantidade de artefatos humanos, solidão, características naturais e distância de locais habitados. Com isso, determinou-se quais as áreas mais adequadas para estabelecer acampamentos, turismo ecológico e mesmo caçada. Certos fatores (por exemplo, distância) receberam peso distinto de acordo com o perfil das pessoas que desejariam utilizar tais facilidades.

Aplicações de modelagem hidrológica estão principalmente voltadas a estudos sobre a qualidade e a quantidade de água em uma determinada região. Os dados envolvem não apenas informações sobre a geometria e topologia de bacias hidrográficas, lagos e correntes subterrâneas, mas também fatores como composição química, caracterização das populações da região analisada e fontes poluentes. O estudo do fluxo de correntes exige informações sobre declive, solo, cobertura vegetal, dentre outros. Dentre as aplicações relacionadas destacam-se gerenciamento de bacias hidrográficas, modelagem de fontes subterrâneas e de erosão.

### 2.3.4 Aplicações de Gerenciamento

Administrações municipais, regionais e nacionais têm cada vez mais utilizado SIGs como uma ferramenta de auxílio à tomada de decisões, tanto para a definição de novas políticas de planejamento quanto para a avaliação de decisões tomadas. Como exemplo desta classe de aplicações há planejamento de tráfego urbano, planejamento e controle de obras públicas, planejamento da defesa civil. É crescente principalmente o uso de SIGs como apoio ao planejamento ambiental ou urbano. Em muitos casos, tal planejamento é auxiliado através do acoplamento de sistemas especialistas a SIGs.

[JEM93] enumera vários exemplos de protótipos que utilizam tal tipo de acoplamento, para estudos de impacto ambiental: o sistema EVA nos EUA, no caso específico de consequências de testes militares; o sistema italiano SILVIA e o sistema canadense SAGEE.

[Wor94b] discute as vantagens do uso de SIGs em diferentes níveis governamentais na Grã-Bretanha, mostrando a economia proporcionada. Esta economia é caracterizada como mensurável (no reuso de dados e na rapidez de geração de cenários para auxílio à tomada de decisões) ou intangível (benefícios à população, disponibilidade dos dados).

[MGR93b] contém exemplos do uso de SIG para planejamento urbano, onde tais sistemas são usados para, por exemplo, identificar regiões de expansão urbana, determinar localizações de prédios de serviços (corpo de bombeiros, delegacias de polícia) e definir locais de aterros sanitários.

O estudo de [LHM94] mostra o uso de SIG na Grã-Bretanha para auxílio à avaliação de

propriedades, visando análise de mercado e taxa  o governamental.

[BDN93] descreve a experi  ncia da implanta  o de SIG na prefeitura municipal de Santo Andr  , enquanto [SQ93] aborda algumas condicionantes do sucesso de SIGs e discute aspectos estrat  gicos da concep  o, projeto e implanta  o destes sistemas na gest  o municipal.

O gerenciamento de recursos agr  colas    outra   rea de destaque para SIGs. [AS93] apresenta in  meros exemplos de aplica  es na agricultura, tais como para an  lise de   reas de cultivo ou identifica  o de   pocas de estiagem. O uso de SIG na agricultura    tamb  m discutido por [Roc95].





# Capítulo 3

## Caracterização de Dados Geográficos

### 3.1 Introdução

O termo *dado espacial* denota qualquer tipo de dados que descreve fenômenos aos quais esteja associada alguma dimensão espacial. Por exemplo, as estruturas moleculares de um composto químico são dados espaciais. Os dados utilizados em SIGs pertencem a uma classe particular de dados espaciais: os *dados geo-referenciados* ou *dados geográficos*. O termo denota dados que descrevem fatos, objetos e fenômenos do globo terrestre associados à sua localização sobre a superfície terrestre, num certo instante ou período de tempo.

Este capítulo discute informalmente as principais categorias de dados utilizados em um SIG preparando o leitor para a abordagem mais formal do capítulo seguinte. Em particular, o capítulo introduz as noções fundamentais de campo e objeto geográfico [FG90, ea92, Cou92].

### 3.2 Características Gerais de Dados Geográficos

Dados geo-referenciados são comumente caracterizados a partir de três componentes fundamentais [Aro89, Dan90, Peu94]:

- características não-espaciais, descrevendo o fenômeno sendo estudado, tais como o nome e o tipo da variável;
- características espaciais, informando a localização espacial do fenômeno, ou seja, seu geo-referenciamento, associada a propriedades geométricas e topológicas;
- características temporais, identificando o tempo para o qual tais dados são considerados, isto é, quando foram coletados e sua validade.

A distribuição espacial dos dados geo-referenciados pode se restringir a uma, duas ou três dimensões. Exemplos são medidas pluviométricas em locais pré-determinados (dados pontuais);

especificação de uma rede viária (dados lineares); descrição de uma área de vegetação (dados em 2D); ou fenômenos atmosféricos (dados em 3D). Enquanto determinados fatores permanecem relativamente constantes ao longo do tempo (por exemplo, tipo de solo ou bacia hidrográfica), outros aspectos podem variar rapidamente, em função de mudanças políticas ou sócio-econômicas.

Outro aspecto muito importante reside no fato de que os fenômenos geo-referenciados não existem sozinhos no espaço, ou seja, tão importante quanto localizá-los é descobrir e representar os seus relacionamentos. Estes relacionamentos são inúmeros e, de fato, dependem da percepção do usuário. Tais relacionamentos são muitas vezes imprecisos (por exemplo, “perto”, “à direita de”) e dependentes do contexto.

De uma forma geral, consultas a dados em SIGs podem envolver tanto o estado de um fenômeno quanto a sua distribuição espacial e temporal. As consultas podem se limitar a um fenômeno específico ou a relacionamentos espaço-temporais entre fenômenos geográficos distintos. As consultas típicas de aplicações SIG podem ser caracterizadas como compostas ao longo de três eixos [Peu94]: *onde*, *o que* e *quando*. *Onde* se refere a características espaciais, enquanto *o que* se refere às características não espaciais. Cada consulta fixa ao menos um dos eixos e faz variar os dados ao longo dos outros dois:

- *quando + onde*  $\rightarrow$  *o que*: descreve o conjunto de fenômenos geográficos (*o que*) presentes em uma localização ou em um conjunto de localizações (*onde*), dada uma referência temporal (*quando*). Por exemplo, “Quais os tipos de uso de solo encontrados na Bacia do Rio Piracicaba no período 1980-1995?”
- *quando + o que*  $\rightarrow$  *onde*: descreve uma localização ou seu conjunto (*onde*) ocupada por um ou vários fenômenos geográficos (*o que*) em um dado conjunto de intervalos de tempo (*quando*). Por exemplo, “Quais as áreas no Estado de São Paulo ocupadas por plantações de cana no período 1950-1980?”
- *o que + onde*  $\rightarrow$  *quando*: descreve o conjunto de períodos (*quando*) em que um determinado conjunto de fenômenos geográficos (*o que*) ocupou um conjunto de localizações. Por exemplo, “Qual o período em que a região onde hoje se encontra a UNICAMP foi ocupada por uma plantação de café?”

Na maior parte dos casos, a dimensão temporal é fixa, ou seja, os usuários determinam o conjunto de dados para trabalho em um determinado instante e raramente executam operações que envolvem variações temporais. Alguns exemplos dos processos de análise espacial típicos de um SIG estão apresentados na tabela 3.1, adaptada de [MGR93a].

Análise	Pergunta Geral	Exemplo
Condição	“O que está...?”	“Qual a população desta cidade?”
Localização	“Onde está...?”	“Quais as áreas com declividade acima de 20?”
Tendência	“O que mudou...?”	“Esta terra era produtiva há 5 anos?”
Roteamento	“Por onde ir...?”	“Qual o melhor caminho para o metrô?”
Padrões	“Qual o padrão...?”	“Qual a distribuição da dengue em Fortaleza?”
Modelos	“O que sucede se...?”	“Qual o impacto no clima se desmatarmos a Amazônia?”

Tabela 3.1: Exemplos de análise espacial

### 3.3 Campos e Objetos Geográficos

#### 3.3.1 Modelos de Campos e de Objetos

No contexto de aplicações de SIG, o mundo real é freqüentemente modelado segundo duas visões complementares: o modelo de *campos* e o modelo de *objetos* [FG90, ea92, Cou92]. O modelo de campos (*field model*) enxerga o mundo como uma superfície contínua, sobre a qual os fenômenos geográficos a serem observados variam segundo diferentes distribuições. Um campo é formalizado como uma função matemática cujo domínio é uma (abstração da) região geográfica e cujo contradomínio é o conjunto de valores que o campo pode tomar. Caso se deseje incluir a variação do campo ao longo do tempo, basta considerar que o domínio da função é um conjunto de pares  $(p, t)$  onde  $p$  representa um ponto da região geográfica e  $t$  um instante de tempo. Por exemplo, um campo definindo a cobertura vegetal de uma região será modelado como uma função cujo domínio é uma abstração da região e cujo contradomínio é o conjunto de tipos de cobertura vegetal; a cada ponto da região, a função associa o tipo (ou tipos) de vegetação nele predominante. Esta visão enfatiza a descrição da variação do fenômeno geográfico sem se preocupar com a identificação de entidades independentes.

O modelo de objetos (*object model*) representa o mundo como uma superfície ocupada por objetos identificáveis, com geometria e características próprias. Estes objetos não são necessariamente associados a qualquer fenômeno geográfico específico e podem inclusive ocupar a mesma localização geográfica. Artefatos humanos (redes viárias, edificações) são tipicamente modelados como objetos.

A questão da identidade é fundamental para distinguir os campos dos objetos: por exemplo, existem milhares de áreas no Brasil classificadas como “vegetação arbustiva” (valor de campo), mas apenas uma “UNICAMP” (objeto identificável). Outra distinção entre campos e objetos é

o princípio de *restrição de preenchimento do plano* – (*planar enforcement*), que postula que uma superfície descrita por um campo está particionada em regiões disjuntas, onde a cada região está associado apenas um valor do fenômeno geográfico representado. Este valor pode ser simples ou composto, dependendo da escala, dispositivo de coleta e fenômeno geográfico considerado. A restrição de preenchimento planar não precisa ser garantida no caso do modelo de objetos, pois estes podem ser tanto disjuntos quanto compartilhar o mesmo espaço. Por exemplo, sejam os casos de mapas de vegetação e planta de um loteamento. Num mapa de vegetação, cada ponto da região geográfica está associado a um tipo específico de vegetação (por exemplo, “arbustiva”); na planta, objetos distintos podem conviver em uma mesma localização (por exemplo, um lote e uma edificação).

### 3.3.2 Representações para Campos e Objetos

A modelagem de dados geográficos difere da tradicional não apenas devido às características espaciais, mas também por envolver a questão da representação, que varia conforme a perspectiva do usuário ou aplicação, ou segundo fatores meramente técnicos. Isto frequentemente significa que diferentes representações coexistem para um mesmo campo ou objeto [Chr93]. Por exemplo, o objeto UNICAMP pode ser representado de forma diferente por topógrafos, especialistas em educação, em saneamento, ou ecologistas. Estas alternativas de representação coexistirão mesmo quando os fatores técnicos forem fixados e o fenômeno considerado não mudar.

Os fatores técnicos que levam a múltiplas representações incluem, por exemplo, resolução, escala e projeção cartográfica. Em particular, o termo *resolução* se refere ao nível de abstração com que se deseja considerar um campo ou objeto. Assim, a UNICAMP pode ser considerado como uma região atômica (alto nível de abstração) ou como composto por um conjunto de entidades – os prédios (nível de abstração mais detalhado). Em alguns casos, a variação de resolução, como citado em [Rig95], pode ser implementada por mudança de escala: a região da UNICAMP é considerada atômica em uma escala pequena e composta por prédios em uma escala maior. No entanto, nem sempre uma mudança de escala é suficiente para estabelecer correspondência entre duas representações de um mesmo fenômeno em resoluções diferentes.

Do ponto de vista de implementação de um SIG, o gerenciamento de representações múltiplas deve ser considerado em vários níveis: interface (apresentação), modelagem e estruturas de dados. A cada um desses níveis correspondem diferentes problemas: redundância de dados, consistência e multiplicidade de comportamentos de um mesmo campo ou objeto, em função de suas representações. Além destes pontos, deve-se levar em conta que uma representação pode ser materializada, quando armazenada explicitamente no banco de dados, ou calculada, quando há uma função registrada no SIG capaz de computá-la a partir de outra representação.

A dicotomia de modelagem de campos ou objetos se reflete, no nível de representação, no chamado debate *raster* versus *vector* [Cou92]. Campos são frequentemente representados no formato de *tesselação* ou *matricial*, ou seja, em uma matriz cujos elementos são unidades poligonais regulares do espaço (*células*). Os relacionamentos topológicos no espaço são implicitamente determinados a partir da vizinhança das células e as coordenadas geográficas são obtidas

indiretamente a partir da posição da célula na matriz. A cada célula corresponde um valor do campo representado (ou seja, não pode haver dois valores distintos para uma mesma célula). O valor associado a cada célula é considerado como válido para toda a região por ela representada, o que pode causar erros consideráveis dependendo da escala de obtenção dos dados coletados. O termo *raster* designa células retangulares, mas na maioria das vezes é usado como termo genérico para a representação matricial. Células retangulares são também chamadas de *pixels*. Outros formatos para células são triângulos e hexágonos [KHFM93]. Operações típicas neste tipo de representação são a sobreposição de matrizes, combinando os valores das células através de funções matemáticas, ou a abstração de conjuntos de várias células adjacentes em uma única célula, cujo valor é calculado a partir dos valores das células selecionadas (generalização).

Já um objeto geográfico é tipicamente representados no formato *vetorial*, ou seja, sua geometria é descrita utilizando *pontos*, *linhas* e *polígonos*. Naturalmente, deve haver uma transformação bem definida entre o sistema de coordenadas utilizado na descrição geométrica e o sistema de coordenadas geográficas adotado. Linhas são formadas por seqüências de pontos; e polígonos (abertos ou fechados) por seqüências de linhas. Operações comuns neste tipo de representação são operações topológicas (por exemplo, adjacência, inclusão) e métricas (distância, área). Representações vetoriais podem ter diferentes níveis de sofisticação. Representações *não estruturadas* se restringem a armazenar listas de coordenadas sem qualquer preocupação com otimização do processamento (por exemplo, o modo *spaghetti*). Representações *topológicas* permitem armazenar, associada à localização, informação sobre relacionamentos de contigüidade e vizinhança dos elementos armazenados [SMSE87].

## 3.4 Principais Classes de Dados Geográficos

Esta seção discute brevemente algumas das principais classes para tratamento de dados geográficos em Geoprocessamento, tais como mapas temáticos, mapas cadastrais, redes, modelos numéricos de terreno e imagens.

### 3.4.1 Mapas Temáticos e Mapas Cadastrais

Mapas temáticos são mapas que mostram uma região geográfica particionada em polígonos, segundo os valores relativos a um *tema* (por exemplo, uso do solo, aptidão agrícola). Os valores dos dados são em geral inseridos no sistema por digitalização ou, de forma mais automatizada, a partir de classificação de imagens.

A maior parte dos sistemas armazena dados para estes tipos de mapas usando uma representação topológica, onde as arestas determinam os limites entre regiões. A topologia construída é do tipo aresta-vértice-face: arestas se conectam entre si através de vértices (pontos inicial e final) e arestas que circundam uma área definem uma face (polígono ou região).

Um mapa temático também pode ser armazenado no formato matricial (*raster*). Neste caso, a área correspondente ao mapa é dividida em células de tamanho fixo. Cada célula terá um

valor correspondente ao tema mais freqüente naquela localização espacial.

Em mapas temáticos, os polígonos apresentados são resultado de funções de análise e classificação de dados e não correspondem a elementos identificáveis do mundo real. Mapas cadastrais, ao contrário, apresentam objetos identificáveis (por exemplo, lotes de terreno). Por exemplo, os lotes de uma cidade são elementos do espaço geográfico que possuem atributos (por exemplo, proprietário, valor venal, IPTU devido). Os dados são em geral armazenados usando uma representação topológica.

### 3.4.2 Redes

Em Geoprocessamento, o conceito de rede denota as informações associadas a serviços de utilidade pública, como água, luz e telefone; redes relativas a bacias hidrográficas; e rodovias. As informações espaciais são usualmente armazenadas em forma de grafo que armazena informações sobre recursos que fluem entre localizações geográficas distintas. Os grafos usam topologia arco-nó, onde os arcos têm um sentido de fluxo e os nós podem ser fontes ou sorvedouros.

Como observa Goodchild [Goo92], uma rede é um sistema de endereçamento 1-D embutido no espaço 2-D. Para citar um exemplo, tome-se uma rede elétrica, que tem, entre outros, os componentes: postes, transformadores, sub-estações, linhas de transmissão e chaves. As linhas de transmissão serão representadas topologicamente como os arcos de um grafo orientado, estando as demais informações concentradas em seus nós.

Operações típicas sobre rede são cálculo de caminho ótimo e crítico. Vários algoritmos de cálculo de propriedades da rede podem ser resolvidos apenas considerando a topologia da rede e de seus atributos.

A área de redes é um grande motivador para inovações em SIG, merecendo destaque a integração de dados, a segmentação dinâmica, linguagem de visualização e a capacidade de adaptação.

Outro aspecto necessário para aplicações de redes é a capacidade de definir diferentes cortes lógicos de uma rede sem necessidade de replicar a estrutura topológica da rede. Por exemplo, ao se asfaltar parte de uma estrada de terra, será preciso atualizar esta informação, sem ter de redigitalizar todas as coordenadas de localização da estrada. Esta capacidade, usualmente denotada por segmentação dinâmica, permite separar os diferentes níveis de informação relativos a uma mesma rede.

O pacote mínimo disponível nos sistemas comerciais consiste tipicamente de cálculo de caminho ótimo e crítico. Este pacote básico é insuficiente para a realização da maioria das aplicações, pois cada usuário tem necessidades completamente distintas. No caso de um sistema telefônico, uma questão pode ser: “quais são todos os telefones servidos por uma dada caixa terminal?”. Já para uma rede de água, pode-se perguntar: “Se injetarmos uma dada percentagem de cloro na caixa d’água de um bairro, qual a concentração final nas casas?”

### 3.4.3 Modelos Numéricos de Terreno

O termo modelo numérico de terreno (ou MNT) é utilizado para denotar a representação uma grandeza que varia continuamente no espaço. Comumente associados à altimetria, também podem ser utilizados para modelar unidades geológicas, como teor de minerais ou propriedades do solo ou subsolo (como aeromagnetismo).

Entre os usos de modelos numéricos de terreno, pode-se citar [Bur86]:

- Armazenamento de dados de altimetria para gerar mapas topográficos;
- Análises de corte-aterro para projeto de estradas e barragens;
- Cômputo de mapas de declividade e exposição para apoio a análises de geomorfologia e erodibilidade;
- Análise de variáveis geofísicas e geoquímicas;
- Apresentação tridimensional (em combinação com outras variáveis).

Uma das formas de obter dados relativos a uma grandeza que varia continuamente no espaço é através de seleção de pontos nos quais é feita a amostragem da grandeza (por exemplo, a composição do solo de uma região é determinada a partir de amostras retiradas de locais predefinidos). Outra forma de aquisição de dados é através da digitalização de isolinhas a partir de mapas existentes ou valores computados a partir de imagens digitais, como é o caso de altimetria. *Isolinhas* são linhas definidas como uma série de coordenadas  $x$  e  $y$  e rotuladas por uma grandeza  $z$ .

Existem dois grupos básicos de representação para MNTs: grades regulares e grades triangulares. Uma *grade regular* é uma matriz de elementos com espaçamento fixo, onde a cada elemento é associado o valor estimado da grandeza na posição geográfica coberta pelo elemento. As grades regulares são obtidas por interpolação das amostras ou, alternativamente, geradas por reprodutores com saída digital. Uma *grade triangular* é uma estrutura topológica vetorial do tipo arco-nó, onde os nós são conectados formando triângulos que não se sobrepõem e cobrem totalmente a área de interesse. Uma grade triangular é formada por conexão entre amostras, com distribuição espacial possivelmente irregular, utilizando algum método de triangulação. O método de triangulação mais comum é o de Delaunay [Tsa93, FG94] (sujeita a restrições).

Os procedimentos de interpolação espacial variam em função das suposições, de suas limitações, do formato dos dados de entrada e do uso a que se destinam, usualmente supondo uma variação gradual e contínua das grandezas no espaço. Algumas rotinas mais sofisticadas aceitam ser informadas sobre rupturas na distribuição espacial das grandezas (por exemplo, um abismo no terreno no caso de altimetria).

Para a geração de grades regulares, os procedimentos de interpolação variam de acordo com a grandeza medida. No caso de altimetria, é comum o uso de funções de ponderação por inverso do quadrado da distância, onde o valor de um elemento da matriz é calculado como



a média entre os valores conhecidos de entrada, tal que os elementos mais próximos recebem maiores pesos. Já para variáveis geofísicas, procedimentos de variáveis geofísicas, são utilizados procedimentos de filtragem bidimensional ou de geoestatística (como a krigeagem).

As grades triangulares são normalmente melhores para representar a variação do terreno, pois capturam a complexidade do relevo sem a necessidade de grande quantidade de dados redundantes, pois regiões mais acidentadas podem ser representadas por uma maior quantidade de nós. As grades regulares têm grande redundância em terrenos uniformes e dificuldade de adaptação a relevos de natureza distinta no mesmo mapa, por causa da grade de amostragem fixa.

Os modelos numéricos de terreno também podem ser convertidos para mapas temáticos e para imagens. Em ambos os casos, a grandeza numérica é quantizada, seja para um número pequeno de valores (caso de mapas temáticos), seja para a variação associada a imagens (valores discretos).

### 3.4.4 Imagens

Tradicionalmente, muitos SIGs utilizam mapas como a forma básica para a construção dos bancos de dados geográficos e tratam imagens obtidas por sensoriamento remoto apenas como uma forma de captura indireta de informação espacial a ser incorporada a tais mapas. Porém, com o desenvolvimento das áreas de Sensoriamento Remoto e Processamento Digital de Imagens, aliadas a técnicas para armazenamento, recuperação e apresentação de imagens, estas passaram a ser utilizadas em conjunto, ou até mesmo em substituição, a mapas.

Uma imagem de um objeto real é, em princípio, contínua tanto na variação espacial como nos níveis de cinza. Para que uma representação digital dessa imagem possa ser criada, é necessário discretizá-la tanto no espaço – amostragem – quanto na amplitude – quantização.

Uma *imagem digital* consiste em uma matriz de números digitais chamados de *pixels* (uma abreviação de “picture element”). Cada pixel corresponde a um retângulo na superfície da imagem original, não-digital.

Em uma *imagem espectral*, os pixels contêm valores relacionados à reflectância ou à emitância do solo e sua cobertura para a área retratada pela imagem. De um modo geral, denomina-se *imagem multiespectral* a coleção de imagens de uma mesma cena, num mesmo instante, obtida por vários sensores com respostas espectrais diferentes.

As medidas registradas pelos sensores variam de fenômeno para fenômeno. Por exemplo, a banda TM 4 do Landsat (ver seção 1.3), no infra-vermelho próximo é especialmente boa para detectar diferenças na vegetação e no solo, enquanto a banda TM 1, no azul, tem boa penetração em corpos aquosos. Já fotografias aéreas em infra-vermelho são apropriadas para se distinguir objetos de vegetação. A resolução de uma imagem espectral, dada pela área coberta por pixels, é função do sensor, de características da cena e do pré-processamento dos dados.

Uma imagem de  $k$  bandas tem  $k$  níveis de cinza associados a cada pixel, um para cada banda espectral. Uma *assinatura espectral* representa um vetor de dimensão igual ao número

de bandas de uma imagem, cujas coordenadas são medidas de radiância.

Uma imagem colorida é, na realidade, um conjunto de três imagens obtidas por três sensores com curvas de resposta espectral diferentes (por exemplo, verde, azul e vermelho). A sensação de cor que se experimenta é devido à existência, na retina, destes três tipos de sensores.

O processamento digital de imagens compreende um conjunto de operações que permitem manipulações numéricas sobre imagens digitais. A aplicação de técnicas de análise de imagens pode permitir a identificação de qualquer fenómeno, a partir dos dados fornecidos por uma variedade de sensores remotos atuando sobre determinada área.

Na análise de imagens, a entrada do processamento é uma imagem e a saída é uma descrição não pictórica da imagem, num processo chamado de *redução de dados*. Na manipulação numérica em geral, tanto a entrada quanto a saída são imagens. Existem duas grandes classes de operações: a *transformação radiométrica*, onde os valores dos pixels são alterados sem modificação da geometria; e a *transformação geométrica*, onde a geometria da imagem é alterada.

Pode-se ter dois tipos de transformação radiométrica: as *pontuais*, onde a transformação do valor de um pixel só depende do valor do pixel na imagem original; e as de *vizinhança ou locais*, onde o valor resultante do pixel depende também dos valores dos pixels vizinhos.

Por fim, a manipulação de imagens pode também dar origem a *imagens temáticas*, onde os valores dos pontos não correspondem a medidas físicas, mas sim a códigos representando temas (como “floresta”, “água” e outros).



## PARTE II



# Capítulo 4

## Modelagem de Dados Geográficos

### 4.1 Introdução

Um modelo de dados deve fornecer ferramentas para descrever a organização lógica de bancos de dados, bem como definir as operações de manipulação de dados permitidas. Durante o desenvolvimento de uma aplicação específica, o processo de modelagem, quando bem conduzido, produz uma visão abstrata da realidade. Isto facilita o acesso e a reutilização dos dados, bem como a expansão do banco de dados e a sua integração com outros criados para aplicações distintas, mas que descrevem a mesma realidade. Todas estas observações aplicam-se ao caso de SIG, exceto que devem ser estendidas para incluir questões específicas a Geoprocessamento.

Este capítulo endereça o problema de modelagem em SIG, apresentando um modelo de dados geográficos específico, que será utilizado em outros capítulos deste texto. O modelo de dados apresentado está organizado em níveis de abstração, distinguindo claramente entre abstrações conceituais de um fenômeno do mundo real e suas representações. O modelo baseia-se nos trabalhos dos autores [CFS<sup>+</sup>94, MJ93, MCS<sup>+</sup>94] e adota o enfoque multinível de [GV93]. O capítulo termina com uma discussão sobre algumas propostas de padronização para dados geográficos [Sur94, MW92, Car94].

Propostas mais antigas de modelos de dados geográficos baseiam-se no modelo relacional de dados, porém estudos mais recentes recomendam o uso de modelos orientados a objetos. Além de flexíveis, tais modelos facilitam a especificação incremental de aplicações, característica importante em SIG.

Atualmente, o modelo de dados orientado a objetos mais completo utilizado comercialmente é o proposto pelo padrão SAIF, descrito no final deste capítulo. As referências [Chr90, Cou92, FR93, Goo91, KT92, RM92] contêm discussões básicas sobre modelagem de dados geográficos enquanto que [MMS93, WHM90, SV92, SA93, DRS93b, AA93, PMP93, Wor94a, Ege95] incluem considerações envolvendo orientação a objetos. As referências [Lan93a, MJ93, NTE92, Peu93, FCF92, Bot95] combinam aspectos temporais com modelagem de dados geográficos, a referência [HS92a] aborda aspectos de hipermídia, e a referência [ZM92] endereça aspectos de inteligência artificial.

## 4.2 Um Modelo de Dados Geográficos

Esta seção descreve os aspectos estruturais de um modelo de dados geográfico que apresenta uma abordagem unificada das visões de campos e objetos e permite a existência de múltiplas representações para um mesmo fenômeno geográfico. O modelo separa a especificação em diferentes níveis de abstração, liberando assim o usuário da necessidade de se envolver com detalhes de implementação física. Este modelo de dados é utilizado no sistema SPRING 2.0, descrito no capítulo 11.

Para facilitar o entendimento do modelo, descrevemos seus detalhes utilizando terminologia de bancos de dados orientados a objetos. Note-se, no entanto, que a modelagem orientada a objetos não obriga o armazenamento em um SGBD orientado a objetos, mas simplesmente visa dar ao usuário maior flexibilidade na modelagem incremental da realidade.

Neste texto, utilizamos o modelo orientado a objetos baseado em classes de [Bee89]. Um objeto é uma instância de uma classe e é caracterizado por seu *estado* (conjunto de valores de atributos) e *comportamento* (conjunto de operações ou métodos que podem ser aplicados ao objeto). Um objeto  $o$  pode ser composto por outros objetos  $o_1, \dots, o_n$ , caso em que  $o$  é chamado *complexo* e  $o_1, \dots, o_n$  são os *componentes* de  $o$ . Este processo de composição é realizado através da aplicação de *construtores* – por exemplo, construtores de conjunto – que permitem a especificação progressiva de objetos cada vez mais complexos a partir de componentes previamente definidos. Objetos não complexos são denominados *simples*. As classes são estruturadas em hierarquias de herança; os ancestrais de uma classe  $C$  na hierarquia são as *superclasses* de  $C$  e seus descendentes as *subclasses* de  $C$ . Os descendentes de uma classe *herdam* sua estrutura e comportamento.

### 4.2.1 Níveis de Especificação

O modelo de dados adota uma estratégia de especificação que identifica quatro níveis de abstração:

**nível do mundo real:** contém os elementos da realidade geográfica a serem modelados como, por exemplo, rios, temperatura, redes telefônicas.

**nível conceitual:** comporta o ferramental para modelar formalmente campos e objetos geográficos em um nível alto de abstração. Este nível determina as classes orientadas a objetos básicas que deverão ser criadas no banco de dados. Neste nível devem ser também definidas as operações e a linguagem de manipulação de dados disponíveis para o usuário.

**nível de representação:** associa as classes de campos e objetos geográficos identificadas no nível conceitual a classes de representações, que podem variar conforme a escala, a projeção cartográfica escolhida, a época de aquisição do dado, ou mesmo conforme a visão do usuário ou aplicação, como já discutido no capítulo 3.

**nível de implementação (físico ou interno):** define padrões, formas de armazenamento e estruturas de dados para implementar as diferentes representações. As decisões de implementação abordadas admitem um número muito grande de variações, em função das aplicações às quais o sistema é voltado, a disponibilidade de algoritmos e o desempenho do hardware. Este assunto será discutido em detalhe no capítulo 9.

Assim, um campo ou objeto geográfico é especificado formalmente pelo usuário no nível conceitual e associado a diferentes representações, cada uma das quais por sua vez armazenada em uma variedade de estruturas físicas. Com este modelo, as dicotomias tradicionais de modelagem – campos versus objeto – e de representação – matricial versus vetorial – podem ser devidamente equacionadas.

Este enfoque multi-nível de modelagem de dados geográficos indica claramente que a interface do usuário em um SIG deve refletir o nível conceitual, mais próximo de sua realidade, ocultando tanto quanto possível os detalhes de representação e implementação. Isso faz com que o usuário trabalhe com conceitos abstratos do mundo real, sem obrigá-lo a entender os problemas de implementação. Ao mesmo tempo, permite especificar operações de diversos tipos para um mesmo objeto conceitual, que podem variar de acordo com o tipo de representação adotada. Por outro lado, não impede que um usuário muito especializado crie novas operações sobre representações, específicas para suas aplicações.

Estes níveis podem ser comparados aos encontrados na arquitetura ANSI-SPARC de sistemas de bancos de dados [ANS75]. Os níveis conceitual e de implementação são os mesmos daquela arquitetura, mas o nível de representação não tem correspondente na área de bancos de dados, pois o problema de representação múltipla raramente ocorre em aplicações convencionais. Esta modelagem multinível exige, no entanto, decisões de mapeamento a cada nível. A passagem do mundo real para o mundo conceitual depende do domínio de aplicação e a passagem do nível conceitual para o nível de representação, de decisões técnicas mais detalhadas.

Por exemplo, imagens de satélite e grandezas geofísicas são normalmente definidas como campos, enquanto mapas cadastrais ou urbanos correspondem a objetos geográficos [Cam95]. Levantamentos temáticos, por outro lado, podem se prestar a duas interpretações, conforme seu uso: quando se tratar de trabalhos de inventário, são modelados como campos (ou, mais especificamente como campos temáticos). No caso de estudos detalhados em médias e grandes escalas, onde cada região é caracterizada por qualificadores específicos (por exemplo, em estudos de fatores sanitários), é conveniente que estes levantamentos sejam associados a objetos geográficos [Cam95].

As duas seções seguintes deste capítulo discutem os níveis conceitual e de representação do modelo, deixando o tratamento das operações e da linguagem de manipulação de dados para os capítulos 5 e 6, respectivamente.



### 4.2.2 Nível Conceitual

O nível conceitual do nosso modelo permite modelar o mundo real como uma coleção de classes orientadas a objetos, classificadas em classes *convencionais* e classes *georeferenciadas* ou *geo-classes*. As instâncias das geo-classes modelam campos e objetos geográficos, enquanto as instâncias das classes convencionais correspondem ao conceito padrão de classes em SGBD orientados a objetos. Os objetos das classes convencionais são denominados *objetos não espaciais*.

Uma mesma entidade do mundo real pode ser modelada em uma geo-classe ou não, dependendo da situação. Por exemplo, um transformador em uma aplicação de rede elétrica, quando não estiver instalado e, portanto, for apenas um item de estoque, pode ser modelado como uma instância de uma classe convencional, com atributos que descrevem suas características tais como peso ou potência. O mesmo transformador, quando instalado em um local da rede, deverá ser considerado então como uma instância de uma geo-classe. As conseqüências desta dupla possibilidade de modelagem serão discutidas na seção 4.2.4. Como a modelagem de dados tradicionais é exaustivamente abordada na literatura de bancos de dados, de agora em diante nos ocuparemos apenas da modelagem de geo-classes.

As instâncias de geo-classes estão associadas a regiões da superfície terrestre, chamadas *geo-regiões*. A geometria de uma geo-região é descrita em um sistema de coordenadas apropriado, que deve ser explícita ou implicitamente fornecido. Diferentes geo-regiões poderão estar descritas em sistemas de coordenadas distintos. No restante do texto, suporemos sempre um sistema de coordenadas planas, induzido por uma transformação cartográfica projetiva e uma dada escala, e consideraremos pontos e linhas poligonais (polilinhas) como geo-regiões degeneradas.

Existem duas hierarquias básicas de geo-classes, cujas raízes são as classes GEOCAMPO e GEOOBJETO. As instâncias de GEOCAMPO, chamadas de *geo-campos*, descrevem campos geográficos e possuem os seguintes atributos espaciais:

- LOCALIZAÇÃO: descreve uma geo-região  $R$ , chamada de *localização* da instância;
- CONTRADOMÍNIO: descreve um conjunto de valores  $V$ , chamado de *contradomínio* da instância.
- MAPEAMENTO: descreve uma função  $f: R \mapsto V$ , chamado de *mapeamento* da instância, modelando um campo geográfico sobre  $R$  que toma valores em  $V$ .

Estes atributos são obrigatórios e formam a *componente espacial* do geo-campo. O projetista da aplicação poderá definir sub-classes da classe GEOCAMPO com novos atributos como, por exemplo, a data de coleta ou dispositivo utilizado. Os atributos convencionais de um geo-campo formam a sua *componente convencional*.

O modelo oferece três subclasses de GEOCAMPO:

- TEMÁTICO: uma instância desta classe é chamada de um *geo-campo temático* e caracteriza-se pelo contradomínio  $V$  ser um conjunto finito arbitrário, cujos elementos definem os

*temas* do geo-campo. Por exemplo, um geo-campo de pedologia de uma geo-região é caracterizado pelo conjunto de temas  $V = \{\text{latosolo roxo, litosolo, cambisolo ...}\}$ , conforme os tipos de solo encontrados.

- NUMÉRICO (ou MNT): uma instância desta classe é chamada de *modelo numérico de terreno* e possui como contradomínio um subconjunto dos reais. Um MNT pode ser utilizado para modelar qualquer grandeza que varie continuamente no espaço, como uma variável geoquímica ou o campo magnético.
- DADOSSENSORREMOTO: esta classe é uma especialização de NUMÉRICO. Uma instância desta classe é chamada de *dados de sensor remoto* e possui como contradomínio um subconjunto dos naturais. A instância recebe este nome pois corresponde em geral a dados obtidos através da discretização da resposta recebida por um sensor (passivo ou ativo) de uma área da superfície terrestre, numa faixa do espectro eletromagnético.

As instâncias de GEOOBJETO, chamadas de *geo-objetos*, descrevem objetos geográficos e possuem o seguinte atributo espacial:

- LOCALIZAÇÃO: descreve uma geo-região, chamada novamente de *localização* da instância.

A localização de um geo-objeto pode ser explicitamente armazenada ou pode ser computada e forma a *componente espacial* do geo-objeto. Os atributos convencionais de um geo-objeto formam a sua *componente convencional*. Novamente, o projetista da aplicação poderá definir subclasses de GEOOBJETO com novos atributos.

Um geo-objeto pode ser elementar, composto ou fraco. Um *geo-objeto elementar* é um geo-objeto que não possui outros geo-objetos como componentes e que sempre tem a sua localização explicitamente armazenada. Um *geo-objeto composto* é um geo-objeto que contém outros geo-objetos como componentes. Um geo-objeto composto pode possuir uma localização explicitamente armazenada, ou tê-la calculada a partir das localizações dos geo-objetos componentes. Um *geo-objeto fraco* é um geo-objeto que existe somente enquanto fizer parte de um geo-objeto composto.

Por exemplo, podemos modelar estradas como geo-objetos e uma malha viária como um geo-objeto composto cujos geo-objetos componentes são as estradas individuais. Se a malha viária contém algumas estradas vicinais que não são suficientemente importantes para serem tratadas como geo-objetos independentes, estas podem ser modeladas como geo-objetos fracos em relação à malha. Alternativamente, se não for necessário especificar cada estrada componente, a malha inteira pode ser modelada como um geo-objeto elementar, cuja localização é uma geo-região complexa.

Note que não é feita nenhuma restrição quanto à consistência entre a localização de um geo-objeto composto e a de seus componentes. Consideramos que esta consistência não deva fazer parte do modelo, cabendo ao usuário assegurá-la.

A noção de geo-objeto composto pode abranger diversos níveis de encapsulamento, dependendo dos componentes e da localização considerados. O geo-objeto UNICAMP, por exemplo,

corresponde à universidade, seus geo-objetos (prédios) e objetos não espaciais (alunos). A universidade está situada no Estado de São Paulo, que pode por sua vez ser também modelado como um geo-objeto composto (com centenas de milhares de geo-objetos componentes e associações entre os mesmos).

### 4.2.3 Nível de Representação

Uma *representação* especifica a componente espacial de um geo-objeto ou de um geo-campo. Assim, uma representação de um geo-campo especifica a localização, o contradomínio e o mapeamento do geo-campo. Já uma representação de um geo-objeto especifica apenas a sua localização, quando explicitamente definida. Note que, em ambos os casos, a especificação da localização deverá indicar detalhes como, por exemplo, a transformação cartográfica projetiva e a escala utilizadas. A discussão nesta seção aborda informalmente estas questões, evitando uma descrição longa dos detalhes.

Existem duas hierarquias básicas de classes de representações, cujas raízes são as classes `REPGEOCAMPO` e `REPGEOBJETO`. As instâncias destas classes são chamadas de *representações de geocampos* e *de geoobjetos*, respectivamente, e naturalmente descrevem representações de geo-campos e geo-objetos. No resto desta seção discutiremos algumas classes destas hierarquias, remetendo o leitor a [Cam95] para uma descrição completa.

Para a classe `REPGEOBJETO`, o modelo oferece subclasses cujas instâncias possuem atributos cujos valores são elementos geométricos simples, como pontos, linhas e regiões com e sem buracos (ver definição na seção 5.2.2), ou elementos complexos construídos a partir destes. Discutiremos em mais detalhe as classes `REPELEMENTAR` e `REPCOMPLEXA`.

Uma instância de `REPELEMENTAR` define um elemento geométrico simples – ponto, linha e região com e sem buracos – em um sistema de coordenadas bem definido. Esta instância representará a localização de apenas um geo-objeto. No nível de implementação, esta instância poderá estar armazenada junto com o objeto ou não, conforme discutido em detalhe no capítulo 9.

Uma instância de `REPCOMPLEXA` define um conjunto  $C$  de elementos geométricos simples ou complexos, em um sistema de coordenadas bem definido, abstraindo em certo sentido a noção tradicional de mapa cartográfico. O modelo permite associar elementos ou mesmo subconjuntos de  $C$  a geo-objetos distintos e interpretá-los como representações das localizações destes geo-objetos. Em outras palavras, uma instância de `REPCOMPLEXA` poderá representar a localização de vários geo-objetos e, no nível de implementação, tipicamente será armazenada separadamente dos geo-objetos aos quais está associada.

Por exemplo, podemos interpretar uma instância  $c$  de `REPCOMPLEXA`, definindo um conjunto  $L$  de linhas, como determinando a localização das estradas de uma região. Sejam  $E_1, \dots, E_n$  geo-objetos correspondendo a estradas específicas. Então, o modelo permite associar linhas  $l_1, \dots, l_n$  em  $L$  a  $E_1, \dots, E_n$  e interpretá-las como representando as suas localizações.

Passando agora para representações de geo-campos, as instâncias de `REPGEOCAMPO` pos-

suem os seguintes atributos, muito semelhantes aos dos geo-campos:

- LOCALIZAÇÃO: descreve uma geo-região  $R$ , chamada de *localização* da instância;
- DOMÍNIO: descreve um conjunto de objetos  $D$ ;
- CONTRADOMÍNIO: descreve um conjunto de valores  $V$  (que supomos incluir o valor indefinido  $\lambda$ );
- MAPEAMENTO: descreve uma função  $g : D \mapsto V$ ;

O modelo oferece as seguintes subclasses principais de REPGEOCAMPO, que variam conforme a definição de  $D$ :

- REPTESSELAÇÃO:  $D$  especifica uma tesselação de  $R$ , ou seja, uma subdivisão de  $R$  em células regulares que não se superpõem e que cobrem completamente  $R$ .
- REPGRADE2D:  $D$  especifica uma grade regular de pontos em  $R$ .
- GRADETRIANGULAR:  $D$  especifica uma grade triangular de pontos cobrindo  $R$ .
- SUBDIVISÃOPLANAR:  $D$  especifica uma partição de  $R$  em regiões simples (definidas vetorialmente, no sentido da seção 5.2.2) que não se superpõem e que cobrem completamente  $R$ .
- REPI SOLINHAS:  $D$  especifica um conjunto de *isolinhas*, ou seja, um conjunto de linhas fechadas que não se cruzam nem se tocam, contidas em  $R$ .
- REPPONTOS:  $D$  especifica um conjunto de pontos contidos em  $R$ .

Uma instância da classe REPTESSELAÇÃO é chamada de uma *representação por tesselação*, ou simplesmente uma *tesselação*. As instâncias das outras classes são denominadas de forma semelhante.

As instâncias destas classes contêm atributos que permitem construir uma aproximação para o mapeamento do geo-campo representado, gerada através de uma função de interpolação bem definida. Em outras palavras, cada uma destas classes é por sua vez apenas a raiz de uma hierarquia de classes, cada uma das quais associada a uma especificação precisa da função de interpolação utilizada e, no caso das duas primeiras classes, da tesselação ou formato da grade adotados.

À guisa de ilustração, considere a existência da classe REPI MAGEM, especialização de REPTESSELAÇÃO. Uma instância desta classe caracteriza-se por especificar uma geo-região retangular  $R$  e uma tesselação que divide  $R$  em retângulos regulares, dispostos em  $m$  linhas e  $n$  colunas. Dados  $i \in [1, m]$  e  $j \in [1, n]$ , existe então um e apenas um retângulo, denotado por  $r_{i,j}$ . Neste caso, a função  $g$  mapeia cada retângulo  $r_{i,j}$  em um ponto de  $V$ . Assim, podemos adotar uma função de interpolação simples que, dada a função  $g$ , induz o mapeamento parcial

$f : R \mapsto V$  tal que, para todo ponto  $p \in R$ ,  $f(p) = g(r_{i,j})$  sse  $g(r_{i,j}) \neq \lambda$  e  $p$  está no retângulo  $r_{i,j}$ ; e  $f(p)$  é indefinida sse  $g(r_{i,j}) = \lambda$  e  $p$  está no retângulo  $r_{i,j}$ . Em outras palavras, a função  $g$  mais a informação sobre como  $R$  divide-se em retângulos regulares é suficiente para especificar a representação do mapeamento associado ao geo-campo.

Por fim, o modelo de dados introduz uma classe, REPRESENTA, para capturar a associação entre geo-campos ou geo-objetos e suas representações. Cada instância desta classe deve ser entendida como um objeto complexo associando um geo-campo ou geo-objeto  $o$  e uma representação  $r$ , permitindo navegar de  $o$  para  $r$  e vice-versa, fazendo assim o papel de um *hyperlink*, em analogia a sistemas hipermídia [HS92a].

Note que este esquema é flexível o suficiente para permitir que:

- a mesma representação induza as componentes espaciais de mais de um geo-campo ou geo-objeto;
- a componente espacial de um geo-campo ou geo-objeto seja induzida por várias representações;
- apenas uma parte da representação induza a componente espacial do geo-campo ou geo-objeto, situação novamente indicada por novos atributos.

Retornando ao exemplo das estradas, seja  $c$  a instância de REPCOMPLEXA, definindo um conjunto  $L$  de linhas, e interpretada como determinando a localização das estradas de uma região. Considere que as estradas são geo-objetos separados. Haverá então uma série de instâncias de REPRESENTA associando tais estradas individuais a  $c$  e indicando quais linhas de  $L$  representam que estradas. Estas linhas específicas de  $L$  fazem um papel semelhante a *âncoras* em sistemas hipermídia, quando a instância de REPRESENTA é entendida como um *hyperlink*.

Por fim, observamos que a associação entre um geo-objeto composto e uma representação pode ser tanto direta, através de uma instância de REPRESENTA, quanto indireta através de instâncias que associem componentes do geo-objeto à representação. Por exemplo, suponha novamente uma malha viária modelada como um geo-objeto composto  $M$ , cujos geo-objetos componentes  $E_1, \dots, E_n$  são as estradas individuais. Seja  $c$  a instância de REPCOMPLEXA anterior. Haverá então uma associação indireta entre  $M$  e  $c$  através das instâncias de REPRESENTA ligando estradas em  $M$  e  $c$ .

#### 4.2.4 Extensões ao Modelo

O modelo proposto deve ser considerado como uma base sobre a qual diferentes extensões podem ser consideradas. Algumas destas extensões são comentadas a seguir e correspondem a problemas em aberto.

Fenômenos georeferenciados são, por natureza, variáveis no tempo. O modelo pode ser, assim, estendido para levar em consideração a dimensão temporal. Nessa extensão, tanto

geo-campos quanto geo-objetos passam a ser definidos em função do instante de coleta dos dados (correspondente à noção de *tempo válido* em bancos de dados temporais). Esta extensão implica, por sua vez, na solução de vários problemas que ainda estão em aberto e que fazem parte de pesquisa de ponta tanto em SIG quanto em bancos de dados, desde o nível de modelagem conceitual até o nível de implementação. Aos interessados sugere-se a leitura, por exemplo, de [Sno92, SS94, Lan93b].

O modelo parte do pressuposto que um fenômeno pode ser modelado usando geo-classes e classes convencionais. Se a evolução temporal for considerada, pode ocorrer que determinada instância de uma classe convencional passe a fazer parte de uma geo-classe e vice-versa. Frequentemente, um mesmo fenômeno pode passar de georeferenciado a convencional várias vezes durante sua vida, dependendo do instante em que for observado (vide, por exemplo, o caso do transformador da seção 4.2.2). Do ponto de vista de bancos de dados, a questão de *migração* de objetos é um problema em aberto.

Uma alternativa para solucionar este problema seria eliminar a distinção entre classes convencionais e geo-classes e supor a existência de uma única hierarquia, GEOCLASSE, em que as componentes espaciais possam ter valores nulos quando não forem georeferenciadas. Esta alternativa acarreta, porém, problemas para a definição das funções de mapeamento de geo-campos, cujo domínio (geo-região) pode ser inexistente. Além disto, apesar de mais simples do ponto de vista de implementação, tal alternativa complica a definição do modelo e, por conseguinte, traz maiores problemas para o usuário.

A discussão do nosso modelo é restrita ao espaço bidimensional. A extensão para 3D difere principalmente do ponto de vista de representação. De fato, no nível conceitual, a caracterização de geo-objetos ou geo-campos não é afetada, embora não se possa mais considerar a definição adotada de geo-região. Por outro lado, as representações disponíveis em 3D oferecem maior leque de variedade e, principalmente, maior número de opções de operações a serem aplicadas. A maioria das aplicações SIG está restrita ao espaço 2D, devido à complexidade das funções de análise no espaço 3D, aos problemas de visualização e, principalmente, à questão da modelagem pelo usuário. Esta última apresenta problemas tanto do ponto de vista de modelagem de processos geográficos quanto de dados georeferenciados, inclusive de cunho cognitivo.

## 4.3 Padrões para SIG

Em geral, padrões facilitam o compartilhamento, a integração e a transferência de dados. Padrões para SIG usualmente incluem padrões para linguagens de especificação, transferência de dados, geocodificação e documentação de metadados e formatos. Além desses, estão surgindo também padrões para qualidade de dados e bibliotecas de objetos espaciais.

Esta seção apresenta um panorama da situação de padrões SIG em nível mundial e descreve brevemente os principais padrões SIG.

### 4.3.1 Visão Geral

Padrões para SIG vêm sendo definidos em diversos níveis: internacional, nacional, federal, industrial ou qualquer outra forma de especificação aberta ao público. Alguns são legalmente estabelecidos enquanto outros são padrões “de facto”, ou seja, adotados por consenso por uma parte significativa da comunidade.

A nível internacional, a ISO (“International Organization for Standardization”) criou em 1994 o Comitê Técnico ISO 211 voltado para informações geográficas e geomática. Atualmente, está desenvolvendo o padrão ISO SQL3, com extensões multimídia e espaciais, chamadas de SQL/MM SPATIAL [ANS94]. Já o ISO 8211 (BS 6690) especifica uma descrição de arquivo para troca de informações, formada por um dicionário seguido dos dados propriamente ditos.

Nos EUA, o FGDC (“Federal Geographic Data Committee”), ligado ao NIST (“National Institute for Standards and Technology”), promove o estabelecimento de padrões SIG federais. O FGDC lidera os esforços do programa NSDI (“National Spatial Data Infrastructure”), cujos objetivos principais são desenvolver capacidade para [Tos94]: (1) achar conjuntos de dados espaciais e identificar usos apropriados destes conjuntos de dados; (2) transferir ou mover dados sem perda de conteúdo ou estrutura; e (3) coletar dados para suportar múltiplos propósitos. O objetivo (1) foi contemplado com o desenvolvimento de padrões para metadados espaciais. O objetivo (2) está contemplado através da adoção do SDTS (“Spatial Data Transfer Standard”). O SDTS [MW92] foi adotado como FIPS (“Federal Information Processing Standard”) 173 em 1992. Ainda no âmbito federal, existem outros padrões populares nos EUA, vários dos quais de origem militar.

No nível de indústria e setores privados norte-americanos, o ANSI (“American National Standards Institute”) criou o comitê técnico X3L1 em 1993 para o desenvolvimento de um padrão SIG (voluntário). Entre os membros do X3L1 estão o consórcio Open GIS, formado por vendedores, academia, governo e organizações privadas da comunidade SIG e de informática, que está promovendo o desenvolvimento do padrão OGIS (“Open Geodata Interoperability Specification”).

No Canadá, o SAIF (“Spatial Archive and Interchange Format”) foi aprovado como um padrão federal e estadual em 1993, através de um processo do Comitê em Geomática do CGSB (“Canadian General Standards Boards”), tendo sido desenvolvido pela British Columbia Environment, Lands, and Parks. O SAIF tem influenciado o modelo de dados adotado pelo OGIS e a parte espacial do SQL/MM.

O DIGIWG (Digital Geographic Information Work Group), composto por representantes da Organização do Tratado do Atlântico Norte (OTAN), desenvolveu o DIGEST, uma família de padrões para troca de dados espaciais, originalmente voltada para aplicações militares, suportando dados vetoriais, incluindo topologia, e dados matriciais.

O Comitê Técnico 287 do CEN (Comitê Europeu de Padronização), que agrega 16 países da Comunidade Européia, tem a missão de definir padrões para informações geográficas. O CERCO (Comitê Europeu de Representantes de Cartografia Oficial), uma organização de 20 países, tem o objetivo de desenvolvimento de um formato de transferência europeu. As propostas do

CERCO deverão ser consideradas pelo CEN.

[SMJ94] divide em diversos níveis as especificações para trocas de dados, segundo a figura 4.1

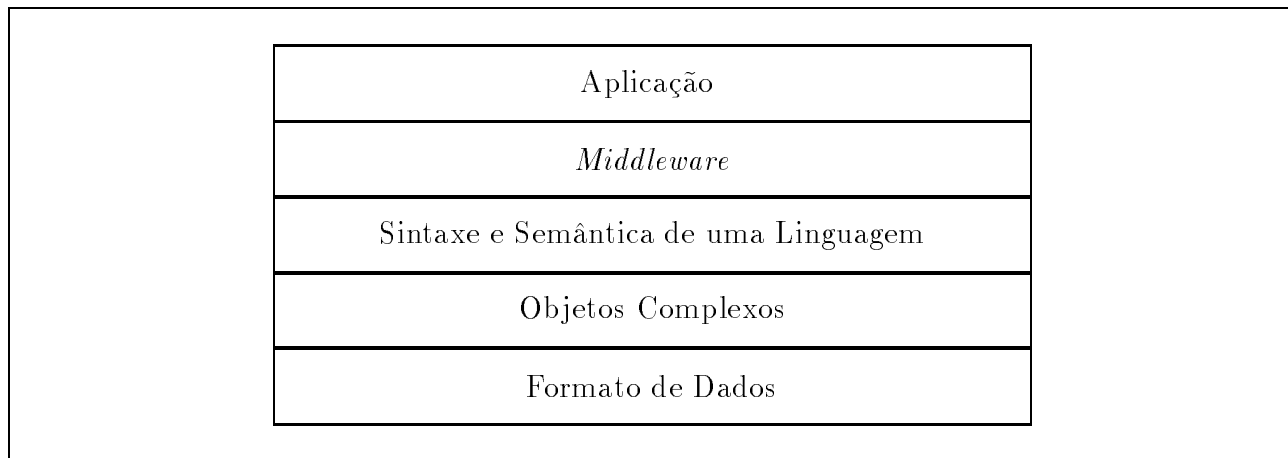


Figura 4.1: Níveis de especificação para troca de dados

O *middleware*, entre a aplicação e os dados, deve conter ferramentas necessárias para se construir, por exemplo, sistemas abertos distribuídos embutindo SGBDs e SIGs. É onde o OGIS está situado.

Os quatro níveis inferiores são dedicados à especificação de troca de dados. O primeiro nível é ocupado pela linguagem de especificação, com sintaxe e semântica bem definidas, onde se situam o ISO SQL3 e o SAIF. Este nível provê a portabilidade do modelo de uma aplicação. Os três níveis seguintes são para transferência de dados, onde o SDTS se situa.

### 4.3.2 O Padrão SAIF

O padrão SAIF [Sur94] propõe uma linguagem para especificação e troca de dados, baseada no modelo de orientação a objetos. Ele é bastante independente do esquema (binário) de codificação dos dados.

O SAIF permite a definição de “profiles” contendo regras e restrições sobre a linguagem. Seu uso facilita a tradução de modelos de SIGs e outros padrões para o SAIF e o compartilhamento de dados através da adoção de um “profile” comum.

O modelo de orientação a objetos do SAIF incorpora conceitos de identidade, generalização, agregação, herança, associações simples entre objetos, entre outros. Não contempla porém a especificação de funções (ou métodos) que, por sua vez, estão sendo definidas no padrão ISO SQL3, baseado no SAIF.

Conceitualmente, o SAIF distingue entre representações de fenômenos do mundo real e representações do espaço e tempo em que estes existem. Representações de fenômenos do mundo real são chamadas de objetos geográficos e podem ter diversos relacionamentos entre si. Estradas,



florestas, medições de temperatura, derramamento de óleo são exemplos de objetos geográficos. Seus relacionamentos podem indicar que estradas se cruzam, derramamentos de óleo ocorrem durante uma tempestade e assim por diante. A posição de cada fenômeno no espaço é representada por um objeto espacial. Se espaço e tempo forem considerados, a posição é representada por um objeto espaço-temporal.

O modelo do SAIF oferece algumas classes construtoras geográficas básicas, a partir das quais outras podem ser criadas. Exemplos são classes espaciais, temporais, geométricas, para relacionamentos explícitos, para metadados (incluindo qualidade dos dados e referenciamento espacial e temporal), para a definição de estruturas de tipos, para anotação, dentre outras. Descrevemos brevemente algumas classes construtoras de mais alto nível.

- **GeographicObjects.** Objetos desta classe representam algum fenômeno do mundo real. Sua localização no espaço e no tempo é dada através de relacionamentos com objetos das classes **SpatialObject** ou **SpatiotemporalObject**. Usuários podem criar novas especializações da classe **GeographicObjects**, que já possui as seguintes subclasses padrão:

- **SpatialDataSet.** Objetos desta classe contêm uma série de instâncias das classes **GeographicObjects** e **Annotation** (veja a seguir). Normalmente possuem alguma semelhança quanto a metadados ou a geo-referenciamento.
- **Coverage.** Objetos desta classe descrevem algum fenômeno distribuído sobre uma região arbitrária do espaço, podendo modelar fenômenos contínuos, como altitude, ou discretos, como uso de solo. Têm a função de mapear uma posição no espaço em um valor, através de uma função espacial. Uma função espacial tem como domínio sempre um conjunto de objetos da classe **SpatialObject** porém seu contra-domínio pode ser qualquer conjunto, seja numérico ou de objetos. Como o SAIF não oferece suporte para a definição de métodos ou funções, apenas os aspectos estruturais dos dados podem ser modelados.

A classe **Coverage** possui três subclasses padrão, que podem ser especializadas pelo usuário:

- \* **Raster.** Subclasses desta classe são usadas para descrever dados *raster* típicos, onde valores são associados a posições em uma grade – por exemplo, imagens de satélite. A grade pode ser definida em termos de uma estrutura regular, multi-dimensional, retangular, hexagonal ou triangular.
- \* **Field.** Objetos desta classe descrevem a distribuição de um fenômeno simples sobre uma dada porção do espaço, por exemplo, um modelo de elevação digital. A geometria de uma instância da classe **SpatialObject** representa uma região contínua no espaço, quando associada a uma instância da classe **Field**.
- \* **PartitionedCoverage.** Objetos desta classe são usados para modelar uma série de fenômenos geográficos discretos, tais como tipos de floresta. Sua principal característica é a não superposição dos fenômenos, sejam esses bi- ou tridimensionais.

- **Graph**. Um objeto desta classe consiste em um conjunto de conexões entre objetos da classe **GeographicObjects**, definidas como relacionamentos. Exemplo destes podem ser rotas de aviões e rede de rodovias.
  - **GeographicComposite**. Um objeto desta classe pode conter outros objetos da classe **GeographicObjects**. A recursão é permitida.
- 
- **SpatialReferencing**. Os objetos guardam informações sobre sistema de coordenadas referentes a uma projeção horizontal (por exemplo, UTM e Lambert) e a definição de um valor vertical (por exemplo distância em metros a partir de um *datum* ou do centro da Terra), e sobre um sistema de referências geodésicas, que define *data* horizontais e verticais.
  - **SpatialObject**. Cada objeto desta classe relaciona-se com um objeto da classe **GeographicObjects** e com um objeto da classe **SpatialReferencing**. Logo, ele contém informação puramente geométrica e também a interpretação desta no contexto do mundo real. Construtores espaciais métricos (por exemplo, distância e vizinhança) e não-métricos (por exemplo, topologia) podem ser definidos sobre estes objetos. Não é permitida a especialização da classe por usuários.
  - **GeometricObject**. Objetos desta classe representam abstrações matemáticas, como pontos, linhas, polígonos e tetraedros. Construtores espaciais topológicos (por exemplo, adjacência e inclusão) podem ser diretamente aplicadas aos objetos. O SAIF provê uma vasta gama de subclasses para **GeometricObject**, de zero a quatro dimensões, mas não é permitida a especialização da classe por usuários.
  - **TimeObject**. Os objetos representam um dado tempo, uma data, um intervalo, uma duração ou uma coleção de tais valores. O tempo pode ser especificado como parte de uma coordenada e, assim, ser incluído diretamente em uma geometria. Esta abordagem só é prática quando um objeto é pontual. Não é permitida a especialização da classe por usuários.
  - **TemporalObject**, **TemporalReference** e **SpatialTemporalObject**. Referências temporais devem ser fornecidas quando existem objetos da classe **TimeObject**, ou o tempo for incorporado a coordenadas. Os objetos das classes **TemporalObject** e **TemporalReferencing** são definidos analogamente a objetos das classes **SpatialObject** e **SpatialReferencing**. Já a classe **SpatialTemporalObject** é uma especialização das classes **SpatialObject** e **TemporalObject**, herdando portanto as suas propriedades.
  - **Annotation**. Textos e símbolos devem ser espacialmente referenciados, tendo os objetos da classe **Annotation** uma estrutura semelhante aos da classe **SpatialObject**.
  - **SpatialRelationship** e **TemporalRelationship**. Objetos destas classes permitem a definição explícita de relacionamentos espaciais e temporais entre objetos, oferecendo uma variedade de associações métricas, topológicas e temporais.

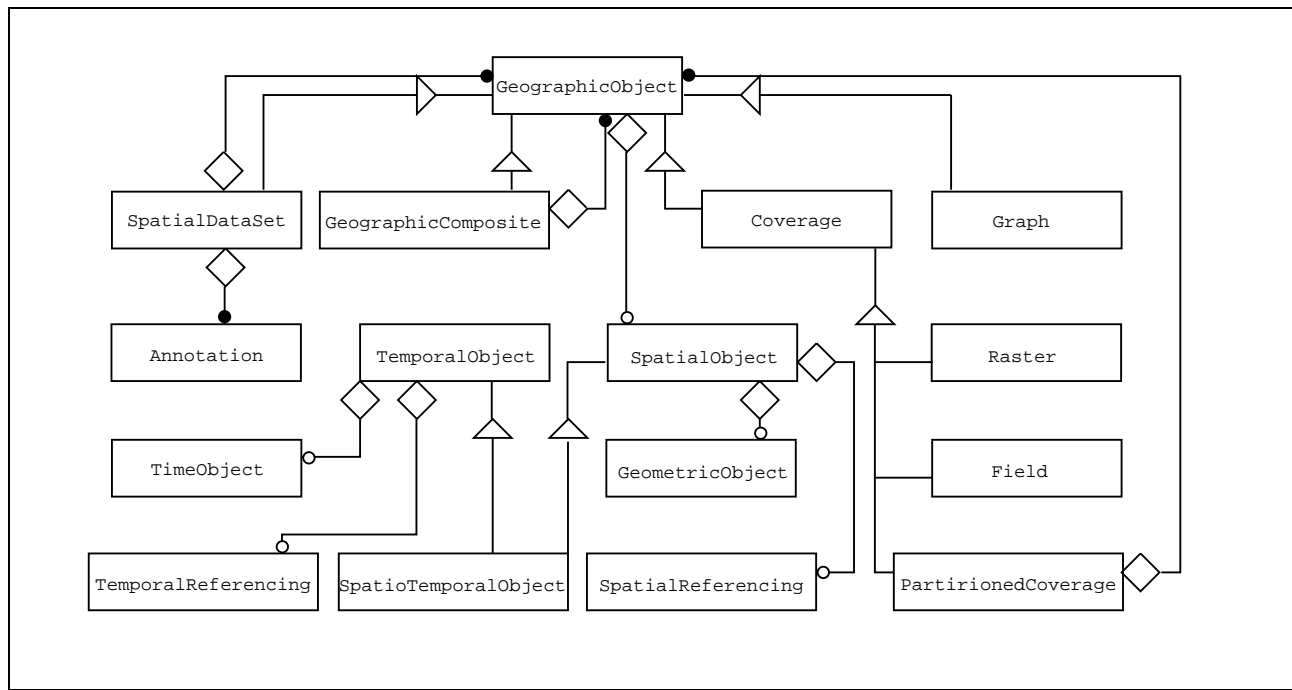


Figura 4.2: O modelo de dados (parcial) do SAIF

A figura seguinte representa graficamente na notação OMT [Rum91] parte das classes e associações apresentadas.

Maiores informações sobre o padrão podem ser encontradas em [SAI].

### 4.3.3 O Padrão SDTS

O SDTS [MW92] é um padrão para transferência de dados espaciais entre sistemas de computadores possivelmente diferentes. Ele especifica construtores de troca, formatos de endereçamento, estrutura e conteúdo para dados geo-referenciados vetoriais e matriciais.

O padrão SDTS está descrito em três partes. A Parte 1 apresenta especificações lógicas requeridas para a transferência de dados espaciais e possui três componentes principais: um modelo conceitual de dados espaciais, a descrição de componentes para qualidade de dados e a descrição de construtores lógicos para formatos de transferência. A Parte 2 contém uma espécie de glossário de termos, com a definição de entidades, atributos, sinônimos e outros. A Parte 3 especifica como implementar a Parte 1 usando o padrão ISOANSI 8211 de troca de dados.

A tradução entre um sistema de dados espaciais específico e o SDTS correspondente é feita em etapas. Inicialmente uma visão da informação espacial é traduzida nos conceitos e termos da Parte 2 e então representada como objetos espaciais, segundo definido na Parte 1. Os objetos espaciais, juntamente com atributos e metadados, são transferidos usando estruturas lógicas definidas também na Parte 1. Estas estruturas são fisicamente formatadas em arquivos, segundo o especificado na Parte 3.

O modelo conceitual SDTS possui três partes: um modelo de entidades espaciais, descrevendo entidades do mundo real, tais como cidades, fazendas, rodovias e seus atributos; um modelo de objetos espaciais, descrevendo um conjunto de objetos espaciais tais como pontos, linhas, polígonos; e um modelo de fenômenos espaciais, descrevendo o relacionamento entre os dois primeiros. É interessante observar que o SDTS reservou o termo “entidade” para elementos do mundo real, o termo “objeto” para o mundo digital e o termo “feição” (“feature”) tanto para a entidade do mundo real como para sua representação através de um objeto.

O SDTS define cerca de 13 objetos espaciais simples, com 0, 1 e 2 dimensões, orientados para representações em superfícies. Eles são divididos em dois grupos. O primeiro contém apenas objetos com geometria, sem topologia, sendo formado por pontos, segmentos de linhas dados por dois pontos, seqüência de segmentos de linhas (“strings”), arcos definidos por uma expressão matemática, anéis geométricos dados por arcos e “strings” fechados, área interior, polígonos geométricos com buracos, pixels e células de grade. O segundo grupo contém objetos com geometria e topologia, tais como nós, ligações entre nós (“links”), cadeias definidas por segmentos de linha ou arcos limitados por nós, anéis geométrico-topológicos dados por uma seqüência de cadeias, e polígonos geométrico-topológicos. O modelo admite agregações. Assim, um objeto tridimensional pode ser construído como um objeto composto a partir de dois objetos bidimensionais, segundo regras definidas.

Uma transferência de dados espaciais significa uma troca de informações sobre uma feição, ou seja, entidades do mundo real e suas representações espaciais. Para usar o SDTS, um usuário deve descrever sua visão própria da realidade cartográfica e geográfica em termos de entidades e atributos e então relacionar as primitivas geométricas e outras representações digitais do seu sistema com os objetos espaciais SDTS.

A especificação lógica para transferência de dados espaciais é feita através de módulos, classificados em: globais, provendo metadados, incluindo parâmetros necessários para a interpretação da transferência (por exemplo, título, data, escala e projeção); qualidade de dados (por exemplo, precisão de localização e consistência); atributos, em função do tipo do dado; objetos espaciais, seguindo o modelo conceitual; e representações gráficas, com características de apresentação (por exemplo, fonte e cor).

Maiores informações sobre o padrão podem ser encontradas em [SDT].

#### 4.3.4 O Padrão OGIS

O objetivo do OGIS [Car94] é permitir que o usuário tenha acesso a dados geográficos local ou remotamente, sem estar limitado por modelos de dados proprietários, formatos de arquivos ou SIGs que gerenciam os dados, em uma dada localidade.

O projeto do OGIS utiliza três conceitos básicos. Primeiro, todos os dados, processos e servidores são objetos, baseados em uma tecnologia de objetos distribuídos (tal como CORBA ou OLE [Obj94, Bar92]) que tratam da interoperabilidade, comunicação e gerência de objetos e serviços distribuídos. Segundo, um modelo de objetos geográficos para informações espaciais

é proposto. Este modelo é baseado no SAIF e é compatível com o SDTS e o DIGEST. Terceiro, o OGIS permitirá serviços a serem implementados fornecendo ligações entre o modelo de objetos e formatos de dados privados, definindo uma interface de procedimentos padrão para cada elemento do modelo e provendo base ampla e consistente de funções de Geoprocessamento.

O arcabouço do OGIS é uma biblioteca de classes que podem instanciar objetos para modelar tanto dados como funções da aplicação. Estes modelos são o VGM (“Virtual Geodata Model”) e APM (“Applications Programming Model”).

O VGM segue o padrão ISO SQL3, que por sua vez é baseado no SAIF, descrito na seção 4.3.2, estendendo-o com dados de catálogo semelhantes aos metadados definidos pelo FGCD e funções de Geoprocessamento, não necessariamente encapsuladas por classes. Aplicações e serviços usam o VGM como um modelo de dados conceitual para acessar os dados geográficos. Por exemplo, um gerente de acesso pode transformar um dado em formato arco-nó privado em um objeto VGM, instanciando classes para geometrias, relacionamentos espaciais, conteúdos temáticos e metadados.

O APM facilita a criação e implementação de aplicações e serviços. Aplicações são decorrentes de requisitos de um usuário, definidas por um engenheiro de aplicação. Serviços são aplicações já pré-definidas, que capacitam um servidor a interagir com outras aplicações clientes ou com outros serviços. Exemplos de aplicações e serviços são os gerentes de acesso, serviços de transformação, serviços de Geoprocessamento, aplicações OGIS, aplicações já existentes e aplicações não geográficas.

Aplicações em conformidade com OGIS poderão usar dados acessíveis em diversos sistemas de armazenamento com diferentes formatos, através de serviços de gerentes de acesso.

# Capítulo 5

## Operações sobre Dados Geográficos

### 5.1 Introdução

A funcionalidade esperada de um SIG quanto à recuperação e à manipulação de dados varia enormemente em função da área de aplicação e do tipo de usuário [MGR93a]. Por exemplo, na visão de aplicações cartográficas, o SIG deve prover serviços de processamento e apresentação de mapas; na visão de aplicações cadastrais, o SIG deve prover serviços próximos dos tradicionais de banco de dados; na área de administração de recursos naturais, o SIG deve prover serviços sofisticados de análise espacial.

A despeito destas disparidades, há várias alternativas para categorizar as operações de um SIG de forma a facilitar a análise das suas características. Quanto ao resultado, há *operações de construção*, que criam novos objetos, *operações de atualização*, que modificam valores de atributos dos objetos aos quais são aplicadas (sem criar novos objetos), *operações escalares*, que recebem como entrada um ou mais objetos e retornam um valor escalar e *operações booleanas*, que recebem como entrada um ou mais objetos e retornam um valor booleano. Ortogonalmente, há operações *convencionais* e *espaciais*, que dependem apenas da componente convencional ou também da componente espacial dos objetos, respectivamente. As operações booleanas espaciais também são chamadas de *relacionamentos espaciais*. Finalmente, as operações podem ser agrupadas segundo as classes dos objetos envolvidos, sejam eles geo-campos geo-objetos ou ambos.

Em lugar de apresentar uma lista exaustiva de operações espaciais, este capítulo discute como definir certas categorias de operações, enfatizando os aspectos próprios de Geoprocessamento (em oposição a Computação Gráfica) ou que são peculiares ao modelo de dados apresentado no capítulo 4. Em particular, a seção 5.2.2 aborda em mais detalhe o problema de definir relacionamentos topológicos entre representações vetoriais simples. Estas coleções de operações são a base para a definição de linguagens de consulta espacial de alto nível, que permitem a manipulação de conjuntos de objetos a partir daquelas armazenadas no banco de dados geográfico.

Há vários trabalhos sistematizando operações em um SIG [Goo87, MD91, Bur92], ou mesmo

enumerando as propriedades desejáveis para uma álgebra de objetos geográficos [Gut94]. Porém, em geral, estes trabalhos abordam o problema de duas perspectivas distintas: operações de consulta sobre geo-objetos [Ege94], e operações de manipulação sobre geo-campos [Tom90], sem unificar as duas visões. Formalizações dos relacionamentos espaciais entre representações vetoriais podem ser encontradas ainda em [CSE94, MEf94], referências que tomaremos como base para a seção 5.2.

## 5.2 Análise das Operações sobre Geo-Objetos

Esta seção analisa inicialmente o efeito que a introdução de um nível de representação separado provoca na definição de operações sobre geo-objetos. Em seguida, discute em detalhe o problema de especificar relacionamentos topológicos, uma questão que recebeu ampla atenção na literatura de Geoprocessamento.

### 5.2.1 Papel das Representações

No que tange a geo-objetos, a definição de operações espaciais nada mais faz do que traduzir, a partir do nível conceitual, operações sobre representações de localizações (a parte espacial de geo-objetos). As operações sobre representações variam enormemente de complexidade, já que a localização de um geo-objeto pode ser determinada por um simples ponto, em um extremo, ou como componente de uma representação complexa, no outro extremo.

Esta tradução é direta no caso da definição de operações espaciais escalares e de relacionamentos espaciais para geo-objetos, sendo necessário apenas definir uma forma de escolher entre representação alternativas, realizar transformações de escala e projeção cartográfica sobre as representações, se necessário, e aplicar a operação subjacente apropriada. Por exemplo, determinar se um geo-objeto  $g_1$  está contido em outro  $g_2$  significa determinar se a representação escolhida de  $g_1$  está contida na representação escolhida de  $g_2$ . A seção 5.2.2 analisa em detalhe a definição deste tipo de relacionamentos para representações espaciais simples.

Já a definição de operações de atualização ou construção sobre geo-objetos suscita alguns problemas mais complexos, ilustrados pela análise do seguinte exemplo. Considere uma classe  $Re$ , especialização de REPELEMENTAR, e suponha que exista uma operação  $\sigma : Re \mapsto Re$ . Seja  $G$  uma classe de geo-objetos e assuma que as localizações dos geo-objetos em  $G$  tenham uma única representação em  $Re$ . As instancias da classe REPRESENTA induzem uma função  $repr : G \mapsto Re$  tal que  $repr(g) = r$  sse  $(r, g)$  é um par em REPRESENTA. Podemos imediatamente introduzir, para  $G$ , uma operação espacial de atualização  $\Sigma$  tal que, aplicada a  $g \in G$ , substitui a representação da localização de  $g$  por  $\sigma(repr(g))$ . Ou seja,  $\Sigma$  altera o conjunto de instancias associado naquele momento a REPRESENTA, substituindo o par  $(repr(g), g)$  por  $(\sigma(repr(g)), g)$ . Uma operação de construção para  $G$  pode ser definida de forma inteiramente semelhante.

É instrutivo analisar a definição de  $\Sigma$  em detalhe. Em primeiro lugar, o usuário deve estar consciente de que a definição de  $\Sigma$  implicitamente escolhe  $Re$  como a fonte de representações

para  $G$ , embora nada impeça que as instâncias de  $G$  possuam representações em outras classes.

Em segundo lugar, ao aplicar  $\Sigma$  a  $g$ , a suposição de que as localizações dos geo-objetos em  $G$  possuem uma única representação em  $Re$  desobriga o usuário de escolher representações específicas de  $g$ . Se esta suposição fosse eliminada, o usuário seria obrigado a informar sobre quais representações de  $g$  a operação  $\Sigma$  deveria ser aplicada, ou seja, a operação passaria a ter dois novos parâmetros. Note que esta observação de fato se aplica a quaisquer operações sobre geo-objetos.

Em terceiro lugar, como  $Re$  contém representações simples e é fechada em  $\sigma$ , é possível definir a atualização de REPRESENTA de forma imediata. No caso mais geral, a definição de uma operação de construção ou de atualização para geo-objetos poderá exigir a manipulação explícita de representações e de instâncias do relacionamento REPRESENTA. Por exemplo, suponha uma representação  $r$  definida como um conjunto de pontos, linhas e regiões, e considere um geo-objeto  $g$  cuja localização é representada por um subconjunto  $s$  destes elementos, situação formalizada por uma instância  $i$  de REPRESENTA relacionando  $r$  e  $g$  e tendo  $s$  como valor de um atributo específico  $A$ . Relembrando,  $i$  age como um *hyperlink* entre  $r$  e  $g$ , onde o subconjunto  $s$  de  $r$  faz o papel da âncora em  $r$ . Neste caso, uma operação de atualização da localização de  $g$  poderá escolher um outro subconjunto de  $r$  para representar a localização de  $g$ , sem alterar  $r$  propriamente dito, mas alterando o valor do atributo  $A$  de  $i$ , ou modificar algum elemento em  $s$ , alterando portanto a própria representação  $r$ .

Em resumo, a definição de operações de atualização ou construção sobre geo-objetos envolve não apenas operações de atualização ou construção sobre representações, mas também operações sobre o relacionamento REPRESENTA. Por outro lado, a definição de operações espaciais escalares e de relacionamentos espaciais para geo-objetos depende apenas de operações subjacentes sobre representações e de uma forma de escolher entre representações alternativas.

### 5.2.2 Relacionamentos Topológicos

Esta seção faz uma breve análise da questão de relacionamentos espaciais em geral, envolvendo representações de geo-objetos, analisando a seguir, em detalhe, como definir relacionamentos topológicos entre pontos, linhas e regiões simples, assunto fundamental para a definição de linguagens de consulta espacial. Ela segue as referências [EF91, CdFvO93, Cam95] e sua leitura é opcional.

Existem vários tipos de relacionamentos espaciais definidos sobre geo-objetos, dentre os quais encontramos os relacionamentos direcionais e os topológicos. Os relacionamentos *direcionais*, como “acima de” e “perto de”, retornam um valor booleano e descrevem como os geo-objetos estão posicionados uns em relação aos outros. A definição de um relacionamento direcional em geral envolve um marco de referência, um objeto de referência e o objeto em questão. O marco de referência determina a direção na qual o objeto em questão está localizado em relação ao objeto de referência. Estudos sobre sentenças espaciais em linguagem natural revelam que os relacionamentos direcionais dependem de aspectos cognitivos, que variam culturalmente [FM91].



Os relacionamentos *topológicos*, como “dentro de” e “adjacente a”, retornam um valor booleano e são invariantes face a transformações biunívocas e bicontínuas, como transformações de escala, translação e rotação. A definição de um conjunto mínimo de relacionamentos topológicos é objeto de muito debate na literatura. Freeman [Fre75] define um conjunto de treze relacionamentos: “à esquerda de”, “à direita de”, “acima” (mais alto que, sobre), “abaixo” (sob), “atrás”, “próximo a”, “longe de”, “ao lado de” (adjacente a), “tocando”, “dentro de”, “fora de”, “entre”. Feutchwangler [Feu93] lista seis: “adjacência”, “proximidade”, “subdivisão”, “sobreposição”, “vizinho mais próximo” e “sub-região”. Egenhofer e Herring [EH90] propõem: “disjunto”, “encontram”, “igual”, “dentro de”, “contém”, “coberto”, “coberto por” e “sobreposição”. Finalmente, Clementini et al. [CdFvO93] indicam cinco relacionamentos: “dentro de”, “superposto a”, “tocando”, “cruzando” e “disjunto”. A questão da definição de relacionamentos topológicos é discutida em detalhe no resto desta seção.

No que se segue, o espaço topológico considerado será o  $\mathbb{R}^2$  e as definições serão todas breves e informais. Os *elementos topológicos simples* são de três tipos: um *ponto*; uma *linha simples*, que não se intercepta a si mesma e que é ou *circular* ou possui apenas dois *pontos terminais*; uma *região simples*, que é conectada, ou seja, que não é a união de conjuntos disjuntos de pontos, e que não contém buracos.

A *dimensão* de um conjunto de elementos topológicos simples  $\Omega$  é dada por:

$$\begin{aligned} \dim(\Omega) = \lambda & \leftrightarrow \Omega = \emptyset \\ \dim(\Omega) = 0 & \leftrightarrow \Omega \text{ contém pelo menos um ponto} \\ & \text{e nenhuma linha ou região simples} \\ \dim(\Omega) = 1 & \leftrightarrow \Omega \text{ contém pelo menos uma linha} \\ & \text{e nenhuma região simples} \\ \dim(\Omega) = 2 & \leftrightarrow \Omega \text{ contém pelo menos uma região simples} \end{aligned}$$

A *fronteira* de um elemento topológico simples  $\omega$ , denotada por  $\delta\omega$ , é definida da seguinte forma:

$$\begin{aligned} \delta\omega = \emptyset & \leftrightarrow \omega \text{ é um ponto ou } \omega \text{ é uma linha circular} \\ \delta\omega = \{P, Q\} & \leftrightarrow \omega \text{ é uma linha não circular} \\ & \text{e } P \text{ e } Q \text{ são seus pontos terminais} \\ \delta\omega = L & \leftrightarrow \omega \text{ é uma região simples e } L \text{ é a linha circular} \\ & \text{formada por todos os pontos de acumulação de } \omega. \end{aligned}$$

O *interior* de um elemento topológico  $\omega$ , denotado por  $\omega^0$ , é definido como

$$\omega^0 = \omega - \delta\omega.$$

Para caracterizar os relacionamentos entre duas regiões simples,  $A$  e  $B$ , Egenhofer e Herring [EH90] introduziram o *método da matriz de 4 interseções*, que analisa se o conjunto definido em cada entrada da matriz é vazio ou não (uma entrada não vazia é indicada por  $\neg\emptyset$ ):

Para o exemplo da figura 5.1, em que duas regiões,  $A$  e  $B$ , se tocam apenas nas fronteiras  $\delta A$  e  $\delta B$ , a matriz será:

$$\begin{pmatrix} \delta A \cap \delta B & \delta A \cap B^0 \\ A^0 \cap \delta B & A^0 \cap B^0 \end{pmatrix} = \begin{pmatrix} \neg\emptyset & \emptyset \\ \emptyset & \emptyset \end{pmatrix}$$

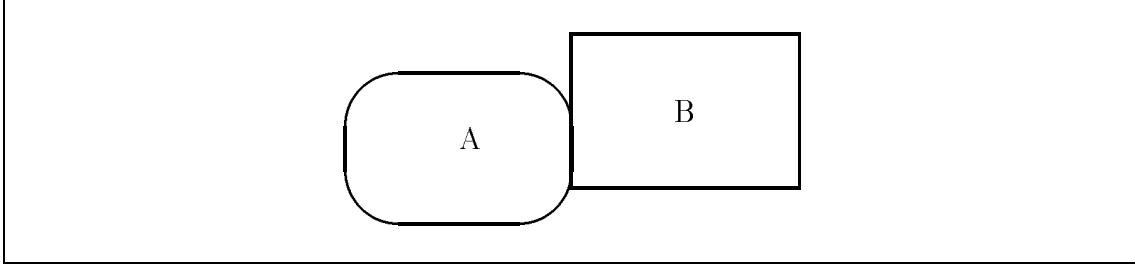


Figura 5.1: Exemplo da matriz de 4 interseções.

Como o método só considera se uma entrada é vazia ou não e como há quatro entradas na matriz, existem dezesseis ( $2^4$ ) possíveis combinações, apresentadas na tabela 5.1, das quais oito são válidas e duas são simétricas, resultando em seis relacionamentos válidos entre regiões: *disjoint*, *in*, *touch*, *equal*, *cover* e *overlap*. Note que as combinações excluídas violam propriedades das regiões simples. Por exemplo, a combinação na segunda linha da tabela é excluída pois, se os interiores de  $A$  e  $B$  se interceptam, a fronteira de  $A$  deveria interceptar o interior de  $B$ , ou vice-versa.

O método da matriz de 4 interseções não diferencia entre algumas situações não equivalentes, como as da figura 5.2, que possuem a mesma matriz apesar de representar relacionamentos distintos (“inclusão” e “cruzamento”):

$$\begin{pmatrix} \delta L \cap \delta A & \delta L \cap A^0 \\ L^0 \cap \delta A & L^0 \cap A^0 \end{pmatrix} = \begin{pmatrix} \emptyset & \neg\emptyset \\ \neg\emptyset & \neg\emptyset \end{pmatrix}$$

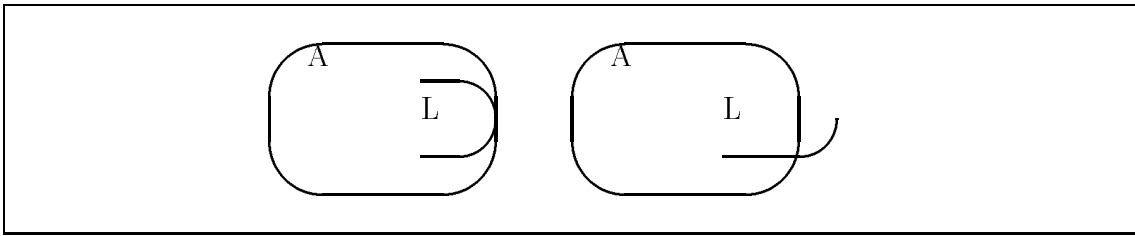


Figura 5.2: Topologias linha-região simples com a mesma matriz de 4 interseções.

Para equacionar problemas como este há várias abordagens na literatura, das quais discutiremos duas, o método estendido da matriz de 4 interseções e o cálculo de relacionamentos.

O *método estendido da matriz de 4 interseções*, discutido em [CSE94, Cam95], considera a dimensão de cada entrada da matriz de 4 interseções, em lugar de apenas avaliar se ela é vazia

$\delta A \cap \delta B$	$\delta A \cap B^0$	$A^0 \cap \delta B$	$A^0 \cap B^0$	nome do relacionamento
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	<i>A disjoint B</i>
$\emptyset$	$\emptyset$	$\emptyset$	$\neg\emptyset$	
$\emptyset$	$\emptyset$	$\neg\emptyset$	$\emptyset$	
$\emptyset$	$\emptyset$	$\neg\emptyset$	$\neg\emptyset$	<i>A in B</i>
$\emptyset$	$\neg\emptyset$	$\emptyset$	$\emptyset$	
$\emptyset$	$\neg\emptyset$	$\emptyset$	$\neg\emptyset$	<i>A in B</i>
$\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	
$\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	
$\neg\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	<i>A touch B</i>
$\neg\emptyset$	$\emptyset$	$\emptyset$	$\neg\emptyset$	<i>A equal B</i>
$\neg\emptyset$	$\emptyset$	$\neg\emptyset$	$\emptyset$	
$\neg\emptyset$	$\emptyset$	$\neg\emptyset$	$\neg\emptyset$	<i>A cover B</i>
$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	$\emptyset$	
$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	$\neg\emptyset$	<i>A cover B</i>
$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	
$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	<i>A overlap B</i>

Tabela 5.1: Relacionamentos topológicos possíveis entre regiões simples

ou não. Por exemplo, nas situações ilustradas na figura 5.2, temos  $\dim(\delta L \cap \delta A) = 1$ , no caso de “inclusão”, e  $\dim(\delta L \cap \delta A) = 0$ , no caso de “cruzamento”, embora em ambos os casos  $\delta L \cap \delta A \neq \emptyset$ . Como a dimensão de cada entrada admite agora quatro possíveis valores (correspondentes aos quatro possíveis valores da dimensão) e como há quatro entradas na matriz, temos um total de 256 ( $4^4$ ) combinações, das quais somente 52 são aplicáveis [CdFvO93]. Novamente, as combinações não aplicáveis são aquelas que apresentam inconsistências, de forma semelhante àquela da segunda linha da tabela 5.1. No entanto, mesmo com esta redução drástica, o número de combinações aplicáveis ainda é elevado demais e, portanto, não é prático criar um nome para o relacionamento definido por cada uma delas e torná-lo disponível aos usuários de um SIG através de uma linguagem de consulta espacial de alto nível. Por outro lado, este método tem o mérito de ser exaustivo e, portanto, de servir de balizador para outros métodos de análise.

O *cálculo de relacionamentos* [CdFvO93] define cinco relacionamentos entre elementos topológicos simples - *disjoint*, *in*, *touch*, *cover* e *overlap* - e três funções - *boundary*, *from* e *to* - e permite a composição de expressões booleanas complexas a partir deles. Por exemplo, se  $L$  e  $A$  são uma linha e uma região simples, respectivamente, a seguinte expressão é válida:

$$(L \text{ disjoint boundary}(A)) \vee (L \text{ touch boundary}(A))$$

A linguagem de consulta espacial conterá então estes relacionamentos e funções e deverá permitir a composição de expressões booleanas. [CdFvO93] prova que o cálculo de relacionamentos é suficiente para exprimir todas as combinações admitidas como aplicáveis pelo método

estendido da matriz de 4 interseções. Os relacionamentos e funções admitidos no cálculo de relacionamentos são definidos no resto desta seção e ilustrados na figura 5.3. No que se segue,  $\omega_1$  e  $\omega_2$  denotam dois elementos topológicos simples dos tipos indicados em cada caso.

O relacionamento *disjoint*, aplicável a dois elementos de quaisquer tipos, é tal que:

$$\omega_1 \text{ disjoint } \omega_2 \leftrightarrow (\omega_1 \cap \omega_2 = \emptyset)$$

O relacionamento *touch*, aplicável a dois elementos dos tipos região e região, linha e região, linha e linha, ponto e região, ponto e linha, é tal que

$$\omega_1 \text{ touch } \omega_2 \leftrightarrow (\omega_1 \cap \omega_2 \neq \emptyset) \wedge (\omega_1^0 \cap \omega_2^0 = \emptyset)$$

O relacionamento *overlap*, aplicável a dois elementos do mesmo tipo, é tal que

$$\begin{aligned} \omega_1 \text{ overlap } \omega_2 \leftrightarrow & (\omega_1 \cap \omega_2 \neq \omega_1) \wedge (\omega_1 \cap \omega_2 \neq \omega_2) \wedge \\ & \dim(\omega_1^0 \cap \omega_2^0) = \dim(\omega_1^0) = \dim(\omega_2^0) \end{aligned}$$

O relacionamento *in*, aplicável a dois elementos dos tipos região e região, linha e região, ponto e região, ponto e linha, é tal que

$$\omega_1 \text{ in } \omega_2 \leftrightarrow (\omega_1^0 \cap \omega_2^0 \neq \emptyset) \wedge (\omega_1 \cap \omega_2 = \omega_1)$$

O relacionamento *cross*, aplicável a dois elementos dos tipos linha e região, linha e linha, é tal que

$$\begin{aligned} \omega_1 \text{ cross } \omega_2 \leftrightarrow & (\omega_1 \cap \omega_2 \neq \omega_1) \wedge (\omega_1 \cap \omega_2 \neq \omega_2) \wedge \\ & \dim(\omega_1^0 \cap \omega_2^0) = (\max(\dim(\omega_1^0), \dim(\omega_2^0)) - 1) \end{aligned}$$

A função *boundary*, aplicável a um elemento topológico simples  $A$  do tipo área, retorna a linha circular que compõe a fronteira de  $A$ .

As funções *from* e *to* são aplicáveis a um elemento topológico simples  $L$  do tipo linha não circular e retornam os seus pontos terminais. Por definição, *from*( $L$ ) retorna o ponto terminal de  $L$  com a menor coordenada em  $X$  ou, caso os dois pontos terminais tenham a mesma coordenadas em  $X$ , o ponto terminal de  $L$  com a menor coordenada em  $Y$ , e *to*( $L$ ) retorna o outro ponto (como  $L$  é não circular, estas funções estão bem definidas). Note que é necessário definir artificialmente duas funções para tratar os pontos terminais separadamente, em lugar

de estender a função *boundary* para linhas, pois os relacionamentos topológicos admitem como argumento um ponto, mas não um conjunto com dois pontos.

Por exemplo, as situações da figura 5.2 são descritas pelas expressões

$$\begin{aligned} & (L \text{ in } A) \wedge (L \text{ overlap boundary}(A)) \\ & (L \text{ cross } A) \wedge (\text{from}(L) \text{ in } A) \end{aligned}$$

Em geral, o cálculo de relacionamentos permite definir com precisão relacionamentos topológicos complexos, sem introduzir um elenco muito extenso de conceitos, e portanto oferece um bom ponto de partida para a definição de linguagens de consulta espacial de alto nível. No entanto, ele precisa ser estendido para cobrir outros elementos topológicos mais complexos, como regiões com buracos, conforme apontado em [CdFvO93].

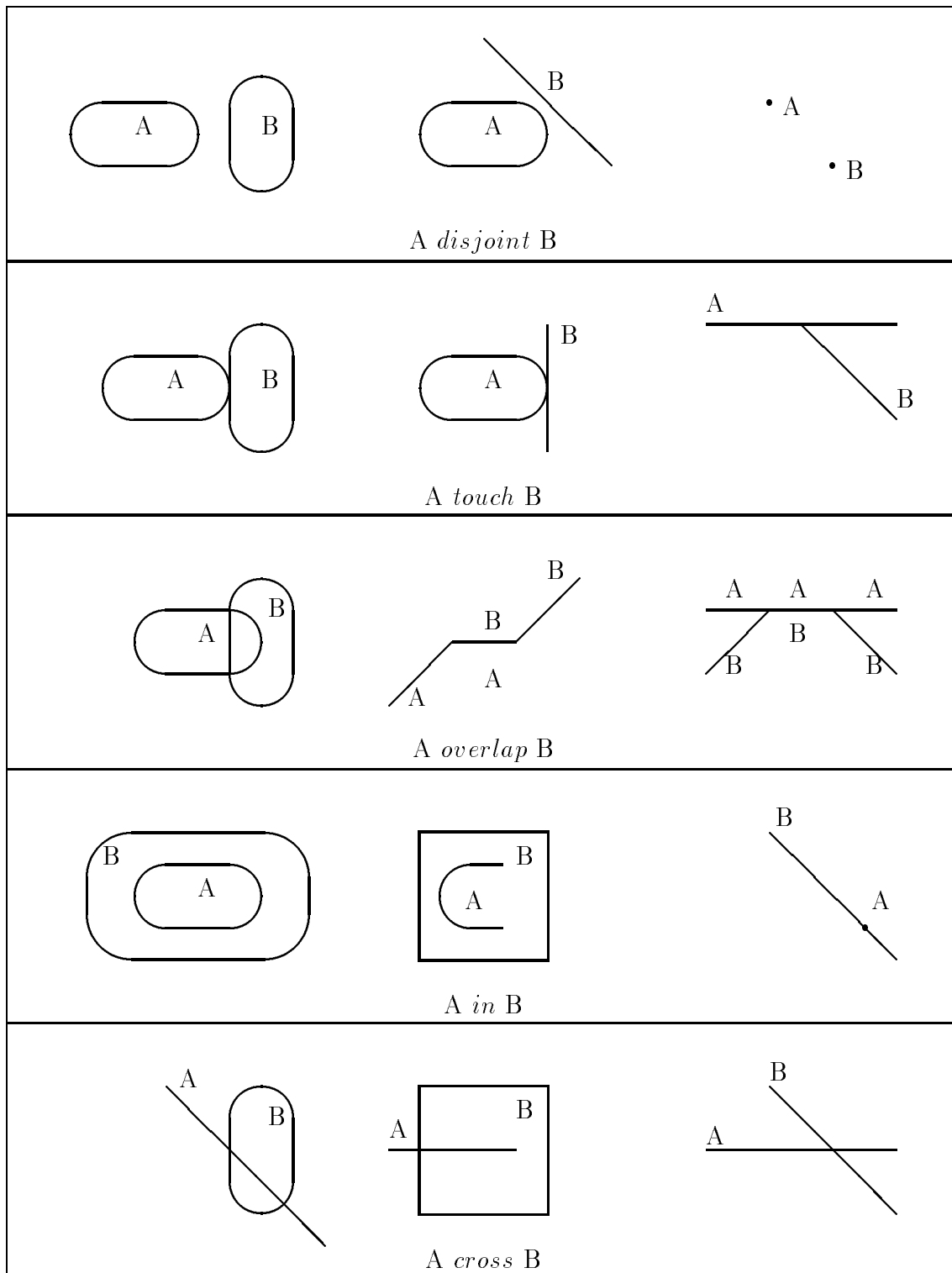


Figura 5.3: Exemplos de relacionamentos topológicos

### 5.3 Análise das Operações sobre Geo-Campos

Esta seção discute brevemente como definir operações sobre geo-campos, explorando o fato de que geo-campos são funções, uma característica enfatizada na descrição do modelo de dados. Esta abordagem mais abstrata apresenta vantagens quando comparada, por exemplo, com o enfoque das linguagens MAP (*Map Analysis Package*), proposta por Dana Tomlin [Tom90], e GRID, disponível no sistema ARC/INFO.

Resumidamente, MAP apresenta um conjunto de operações sobre representações matriciais de geo-campos temáticos e numéricos. GRID é uma implementação de MAP que, embora muito flexível, tem o sério inconveniente de ser fracamente tipada. Em GRID, um geo-campo temático com representação matricial é chamado de “grade de inteiros”, e um geo-campo numérico com representação matricial, de “grade de pontos flutuantes”, o que estabelece uma confusão entre o geo-campo e sua representação.

A discussão concentra-se em apenas duas categorias de operações, chamadas de pontuais e de vizinhança.

#### 5.3.1 Operações Pontuais

Uma operação é *pontual* quando o resultado é um geo-campo cujo valor em cada ponto  $p$  depende apenas dos valores em  $p$  dos geo-campos de entrada. Uma operação pontual é especificada por um mapeamento entre os contradomínios dos geo-campos de entrada e o contradomínio do geo-campo de saída.

Mais precisamente, seja  $R$  uma região e  $V_1, \dots, V_{n+1}$  conjuntos quaisquer. Seja  $\pi : V_1 \times \dots \times V_n \mapsto V_{n+1}$ . Para  $i = 1, \dots, n + 1$ , seja  $C_i$  a classe de todos os geo-campos que possuem  $R$  como localização e  $V_i$  como contradomínio. Denote o mapeamento de  $c_i \in C_i$  por  $f_{c_i}$ . A função  $\pi$  induz então uma *operação pontual*  $\Pi : C_1 \times \dots \times C_n \mapsto C_{n+1}$  tal que, para todo  $c_i \in C_i, i \in [1, n]$ ,  $\Pi(c_1, \dots, c_n) = c_{n+1}$  sse  $c_{n+1}$  é o geo-campo em  $C_{n+1}$  com mapeamento  $f_{c_{n+1}} : R \mapsto V_{n+1}$  tal que, para todo ponto  $p \in R$ ,  $f_{c_{n+1}}(p) = \pi(f_{c_1}(p), \dots, f_{c_n}(p))$ .

Guiando-se pelas classes de geo-campos introduzidas pelo nosso modelo, exemplos de operações pontuais unárias são: as operações de *ponderação*, mapeando um geo-campo temático em um numérico; as de *reclassificação*, mapeando um geo-campo temático em um temático; e as de *fatiamento* e *fatiamento em classes*, mapeando um geo-campo numérico em um temático. A figura 5.4 ilustra a conversão de um geo-campo temático de solos, com contradomínio  $V_1 = \{Le, Li, Ls, Aq\}$ , em um geo-campo temático ponderado de solos, com contradomínio  $V_2 = [0.0, 1.0]$ , através de uma operação de ponderação definida pelo mapeamento  $\{(Le, 0.60), (Li, 0.20), (Ls, 0.35), (Aq, 0.10)\}$ .

Como um exemplo mais complexo, tome-se a operação especificada pelo conjunto hipotético de regras expresso na tabela 5.2 que mapeia um geo-campo temático de solo, com contradomínio fixo  $V_1 = \{Le, Li, Ls, Aq\}$ , um geo-campo numérico de precipitação, com contradomínio  $V_2 = [0, 10000]$ , e um geo-campo numérico de declividade, com contradomínio  $V_3 = [0, 100]$ , em um

geo-campo temático de aptidão agrícola, com contradomínio fixo  $V_4 = \{Le, Li, Ls, Aq\}$ .

<i>Aptidão</i>	<i>Tipo de Solo</i>	<i>Precipitação Média</i>	<i>Declividade</i>
boa	latossolo roxo ou litossolo	>100mm	0-5%
média	cambissolo	100-50mm	5-10%
inadequada	aquoso	<50mm	>10%

Tabela 5.2: Regras para determinação de aptidão agrícola.

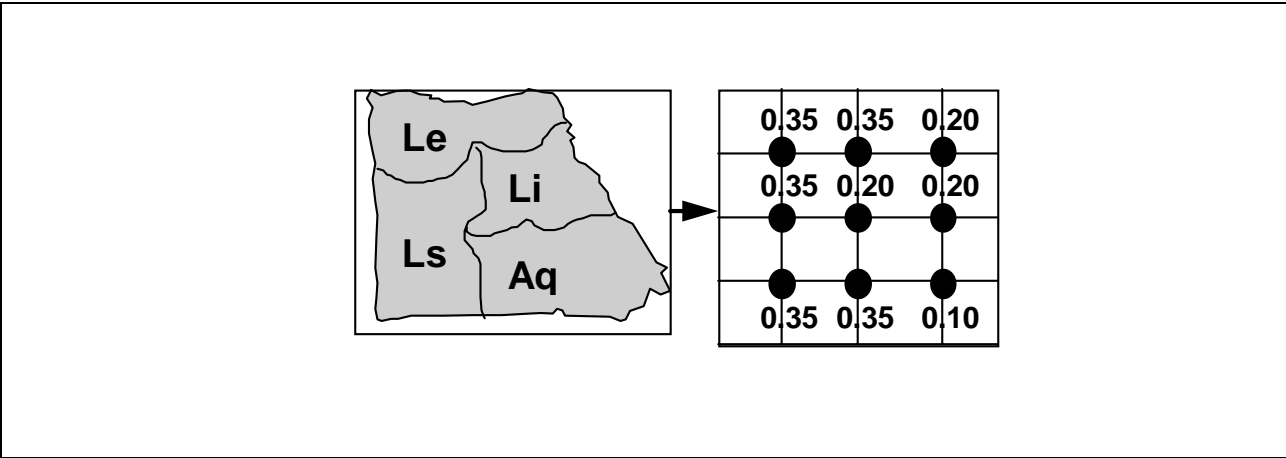


Figura 5.4: Exemplo de operação de ponderação.



### 5.3.2 Operações de Vizinhança

Uma operação é de *vizinhança* quando o resultado é um geo-campo cujo valor em cada ponto  $p$  depende dos valores dos geo-campos de entrada em uma vizinhança de  $p$ .

Exemplos incluem: cálculo do valor *mínimo*, *máximo*, *médio* e *modal* na vizinhança em torno de cada ponto; *filtros* para processamento de dados de sensor remoto; *interpolação espacial* para geo-campos numéricos (como médias por vizinho mais próximo); *cálculo de declividade e exposição* para geo-campos numéricos; *índices de diversidade* para geo-campos temáticos, onde o valor de saída está associado ao número de vizinhos, pertencentes a classes distintas, de um ponto de entrada. As operações de *classificação*, que mapeiam um dado de sensor remoto em um geo-campo temático, também podem ser consideradas operações de vizinhança, pois o problema neste caso é justamente determinar regiões homogêneas em torno dos pontos.

Como ilustração de uma operação de vizinhança, tomemos o caso de um índice de biodiversidade, estimado de tal forma que seja maior em áreas de contato ecológico entre regiões homogêneas. A figura 5.5 apresenta um geo-campo temático de vegetação e mostra o índice de diversidade computado para uma parte do geo-campo.

Um caso especial de operações de vizinhança são as operações *zonais* em que as vizinhanças são induzidas por um geo-campo temático. Seja  $\tau$  um geo-campo temático sobre uma região  $R$ . Intuitivamente, a função de vizinhança  $L_\tau$  é tal que, para todo  $p \in R$ ,  $L_\tau(p) = V$  sse  $V$  é o maior subconjunto de  $R$  tal que o valor do geo-campo em todos os pontos de  $V$  é o mesmo que em  $p$  e  $V$  é um subconjunto conectado, ou seja,  $V$  não é a união de subconjuntos disjuntos de  $R$ . A figura 5.6 exemplifica uma operação de máximo zonal definida sobre um geo-campo numérico, onde um geo-campo temático é utilizado para induzir as vizinhanças, e retornando o maior valor do geo-campo numérico para cada vizinhança.

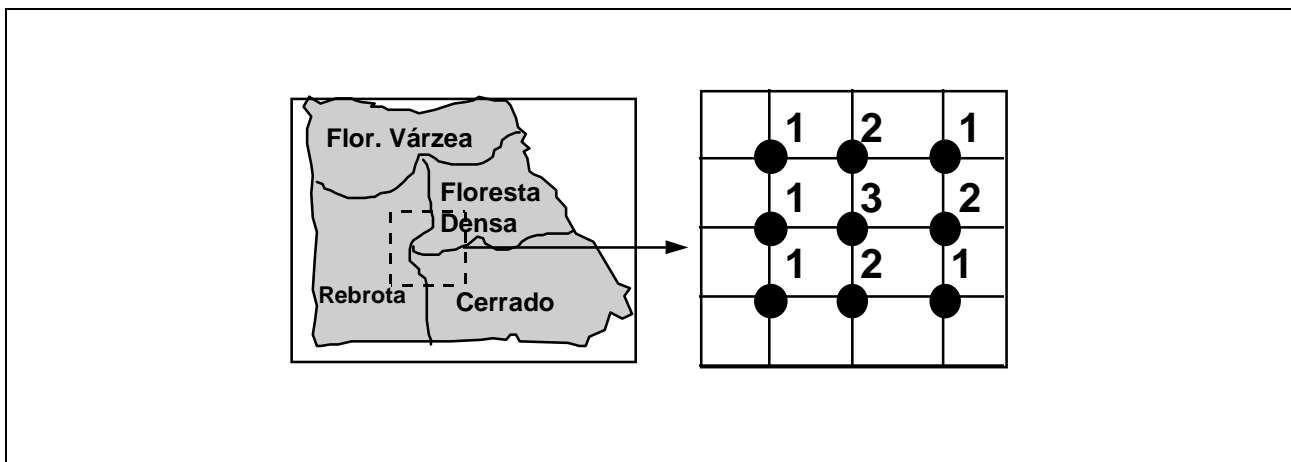


Figura 5.5: Exemplo de cálculo de índice de diversidade.

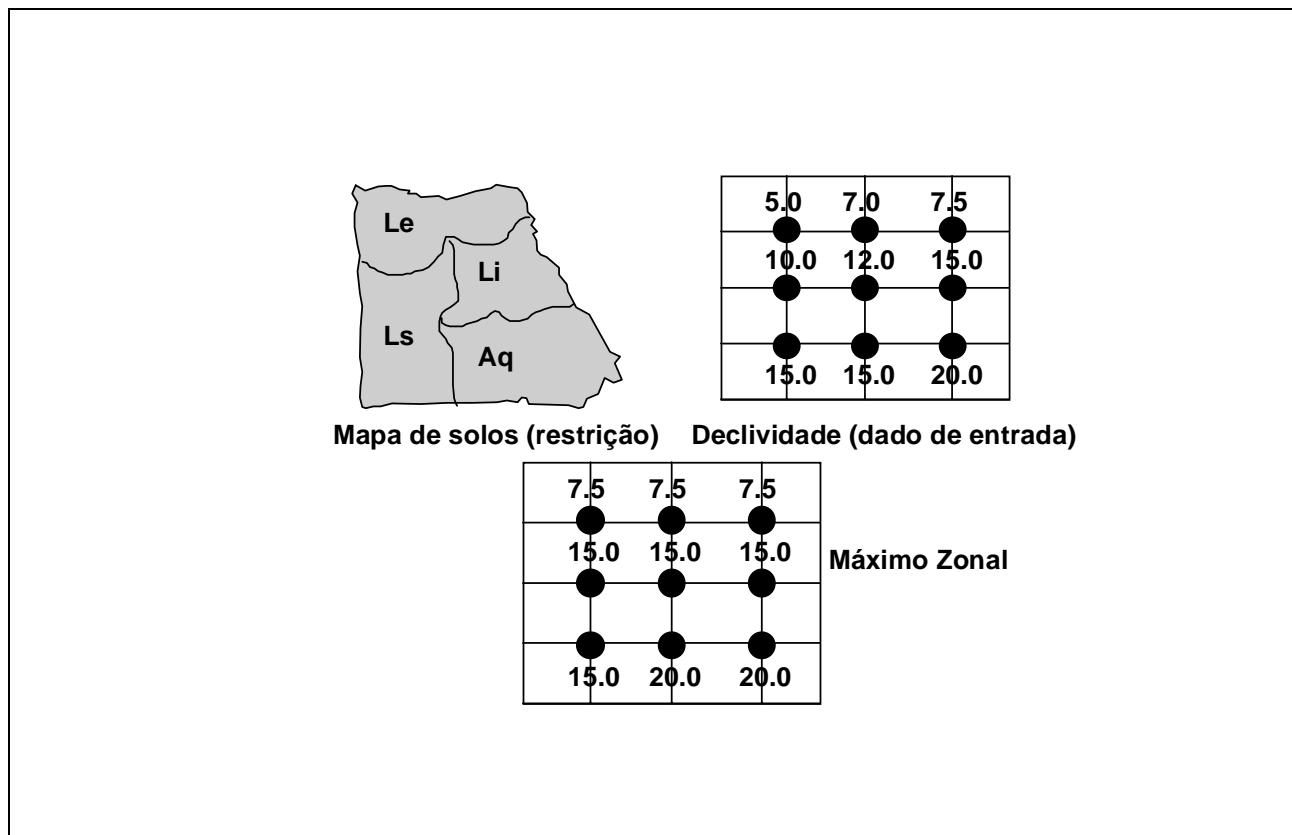


Figura 5.6: Exemplo de cálculo de máximo zonal.

## 5.4 Operações Mistas entre Geo-Campos e Geo-Objetos

Esta última seção introduz exemplos de operações mistas, envolvendo ao mesmo tempo geo-campos e geo-objetos.

Uma operação de *intersecção espacial* (*overlay*) recebe como entrada geo-campos e produz como saída uma coleção de geo-objetos, tendo tipicamente como atributos convencionais os valores dos geo-campos de entrada. Esta situação é comum em aplicações de zoneamento, onde se aplica a intersecção entre geo-campos temáticos para se obter regiões homogêneas, que são então identificadas como geo-objetos. A figura 5.7 ilustra a operação “Determine as regiões homogêneas da Austrália, como intersecção espacial dos geo-campos de vegetação, geomorfologia e solos.”

Na literatura, este tipo de intersecção espacial é muitas vezes classificado erroneamente como um tipo particular de junção espacial (cf. [Gut94]). Como veremos na seção 10.3.1, a operação de junção espacial parte de dois conjuntos de geo-objetos e produz, como resultado, pares de geo-objetos já existentes que satisfazem à restrição desejada. Já o tipo de intersecção espacial aqui discutido cria novos geo-objetos a partir de geo-campos. Deste modo, embora haja semelhanças entre os algoritmos utilizados para implementá-las, a operação de intersecção espacial é conceitualmente diferente das operações booleanas entre geo-campos e operações de junção espacial entre conjuntos de geo-objetos.

Nas definições que se seguem, adotamos as seguintes convenções:  $R$  denota uma geo-região;  $O$  denota um geo-objeto cuja localização é um ponto, linha ou região contido em  $R$ ;  $\Omega$  denota um conjunto de geo-objetos de uma mesma classe  $C$  tal que a união das localizações dos objetos em  $\Omega$  cobre completamente  $R$  e as localizações são mutuamente disjuntas;  $A$  denota um atributo da classe  $C$ ;  $\rho$  denota uma função que mapeia os valores do atributo  $A$  em um outro conjunto de valores.

A operação de *reclassificação* por  $\rho$ , quando aplicada a  $\Omega$ , gera um geo-campo  $\tau$  tal que, para cada ponto  $p \in R$ ,  $\tau(p) = \rho(v)$  sse  $v$  é o valor do atributo  $A$  do (único) geo-objeto em  $\Omega$  cuja localização espacial cobre  $p$ . O exemplo da figura 5.8 ilustra a criação de um geo-campo temático com o crescimento demográfico da América do Sul, gerado a partir dos valores do atributo DEMOGRAFIA dos geo-objetos que representam os países da América do Sul, reclassificando os valores em três faixas: de 0 a 2% a.a., de 2 a 3% a.a. e mais de 3% a.a.. A reclassificação por atributos também é chamada de *fusão* [Gut94].

A operação de *geração de mapas de distâncias*, quando aplicada a  $O$  e  $R$ , produz um geo-campo numérico indicando as distâncias de cada ponto em  $R$  a  $O$ .

Por fim, uma operação *zonal induzida* por  $\Omega$  é definida exatamente como na seção 5.3.2, exceto que utiliza a função de vizinhança  $L_\Omega$  tal que, para todo  $p \in R$ ,  $L_\Omega(p)$  é o conjunto de pontos que pertencem à localização do (único) geo-objeto em  $\Omega$  cuja localização espacial cobre  $p$ . Um exemplo informal de uma operação zonal deste tipo seria: “Dado um geo-campo numérico com a altimetria e um mapa de geo-objetos com os municípios do Vale do Paraíba, crie um novo geo-campo numérico onde cada município será representado por sua altitude média”.

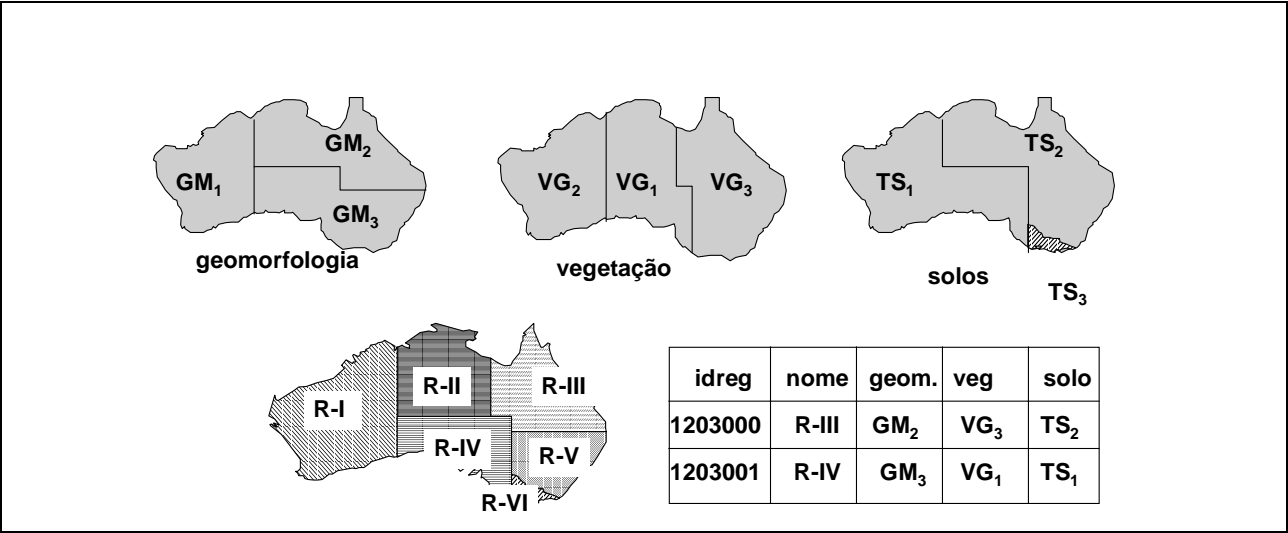


Figura 5.7: Exemplo de cálculo de interseção espacial.

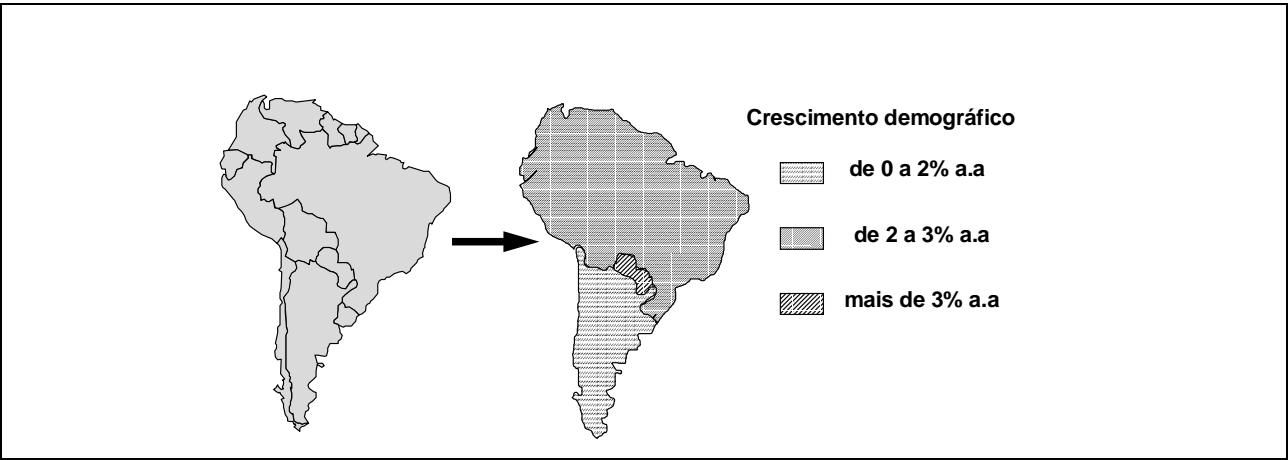


Figura 5.8: Exemplo de reclassificação por atributo.



# Capítulo 6

## Recuperação de Dados Geográficos

### 6.1 Introdução

Este capítulo aborda o problema de definir linguagens de consulta para bancos de dados geográficos. Inicialmente, a seção 6.2 analisa diversas alternativas e resume algumas referências importantes sobre o assunto. Baseando-se no modelo de dados e nas operações definidas nos capítulos 4 e 5, a seção 6.3 descreve uma linguagem de consulta e manipulação espacial, denominada LEGAL - Linguagem Espacial para Geoprocessamento Algébrico [Cam95].

A discussão sobre recuperação e manipulação de objetos geográficos, especialmente linguagens de consulta, abrange inúmeras referências, dentre as quais ressaltamos [Ooi90, RFS88, VvO92]. Este capítulo baseia-se principalmente nos trabalhos dos autores [CHF93a, HFC93, CFC94, Cam95] e de Egenhofer e Frank [EF88, Ege89, Ege92, Ege94].

### 6.2 Visão Geral de Linguagens de Consulta

Esta seção apresenta uma visão geral de linguagens de consultas para bancos de dados geográficos, classificando-as em: *textuais*, usualmente extensões de SQL que acomodam relacionamentos espaciais; *visuais*, privilegiando a manipulação direta de elementos visualizados; e *multimodais*, permitindo uma combinação de linguagem textual e interação via mouse com a tela. Os modos de interação podem ser *interativo/navegacional*, quando o usuário explora as possibilidades existentes, navegando de uma consulta a outra, ou baseado em *consulta/resultado*, quando o usuário solicita uma resposta a uma consulta específica.

#### 6.2.1 Linguagens Textuais

A maior parte das linguagens de consulta textuais para bancos de dados geográficos baseia-se em extensões da linguagem SQL. Paradoxalmente, a principal vantagem e fraqueza deste enfoque origina-se no fato de SQL ser baseada no modelo relacional. Se, por um lado, este fato

facilita a otimização de consultas, por outro lado, a abordagem relacional não é completamente apropriada para a modelagem de dados em aplicações de Geoprocessamento. As principais conclusões do esforço em torno de SQL são [FM91]:

- a inclusão de operadores espaciais em SQL não apresenta maiores problemas, desde que a semântica destes operadores seja definida apropriadamente;
- a inclusão do controle de apresentação e apontamento em SQL é possível, mas aconselha-se projetar uma linguagem separada para lidar com estas questões, em lugar de acomodá-las em cláusulas adicionais na estrutura “select-from-where” típica de SQL.

Exemplos desse enfoque são [Gut88, Ege89, Goh89, Ooi90, Wan94]. Em mais detalhe, a linguagem GQL de [Goh89] combina facilidades de SQL com parametros gráficos de GKS, sendo limitada à manipulação de dados em formato vetorial. A linguagem GEOQL de [Ooi90] estende o SQL com nove operadores espaciais (*intersects*, *adjacent*, *joins*, *ends-at*, *contains*, *situated-at*, *within*, *closest* e *furthest*), o que restringe o tipo de consulta que pode ser realizada. [Ege94] propõe uma extensão a SQL que, além da parte espacial, provê sintaxe para definição da apresentação do resultado da consulta. [Wan94], por sua vez, estende SQL com operadores espaciais que embutem noções nebulosas (perto, longe, quente, frio), previamente compiladas pelo usuário em predicados numéricos.

As propostas de [Gut88, SH91] estão dentre as poucas que não usam extensão de SQL, sendo ambas de difícil manipulação por um usuário final. A primeira prevê, além dos operadores e predicados relacionais, predicados geométricos, transformadores geométricos e operadores que retornam objetos geométricos ou escalares (para operações métricas). Os tipos geométricos manipulados correspondem a dados em formato vetorial. A linguagem de [SH91] está centrada em combinar operações de análise estatística (reclassificação, vizinhança e outras) com operações espaciais (relacionamentos geométricos e topológicos).

## 6.2.2 Linguagens Visuais

Linguagens visuais permitem a construção de uma consulta através da combinação de símbolos, ícones e texto visando facilitar o trabalho do usuário. Esta abordagem, porém, dificulta a formulação de uma semântica precisa para as consultas [BM92]. Exemplos desse tipo de enfoque são [DC91, VMS+93].

A referência [DC91] apresenta duas linguagens gráficas: uma para manipulação de aplicações de redes, denominada GROG, e outra para aplicações temáticas, denominada CIGALES, visando permitir a combinação de aspectos de campos e objetos. GROG permite a especificação de consultas que manipulam arestas e nós em grafos. Neste caso, a consulta é formulada por um grafo orientado, onde os nós correspondem a entidades e as arestas a operadores e relacionamentos. Já CIGALES manipula linhas e regiões. As consultas são também grafos, com outra semântica. O principal problema resultante da combinação dos dois enfoques reside exatamente no fato de que os grafos não são comparáveis e no fato de que sua combinação exige intervenção do usuário para disambiguar consultas.

A linguagem introduzida por [VMS<sup>+</sup>93] permite ao usuário combinar predicados textuais, criados a partir de tabelas de forma similar a QBE, a predicados visuais, combinando símbolos e ícones disponíveis em um menu do sistema. O maior problema é, de novo, determinar a semântica da consulta.

### 6.2.3 Ambientes Multimodais

Ambientes multimodais combinam manipulação direta a consultas visuais e textuais, de forma a tirar proveito dos aspectos positivos de cada uma. O principal obstáculo está em conseguir coordenar conceitos distintos de interação, além da semântica pouco clara da linguagem visual. Exemplos deste tipo de ambiente são descritos em [EF88, AKK94].

O trabalho de [EF88] é baseado em uma interface interativa, oferecendo um *layout* de tela com várias áreas, cada uma das quais dedicada a um tipo de função (consultas textuais, painel de operações, representação gráfica e outras). Consultas são expressas combinando palavras-chave, expressões em um dialeto de SQL estendido e operações definidas no painel através de ícones. Já a linguagem de [AKK94] é baseada no conceito de *mapas dinâmicos*, que combinam a noção de visões em bancos de dados com técnicas de visualização. Mapas dinâmicos são resultados de consultas textuais que podem posteriormente ser manipulados diretamente pelo usuário. A interação, neste caso, alterna consultas textuais (que retornam mapas) e manipulação direta desses mapas, em geral usando sobreposição. Cada mapa dinâmico é definido por um par (visão, visualização) onde a visão é o resultado da consulta e a visualização associa uma apresentação ao resultado da consulta.

### 6.2.4 Linguagens de Consulta sobre Imagens

Idealmente, deveria ser possível realizar consultas sobre imagens, aproveitando as técnicas que vêm sendo desenvolvidas nos últimos anos na área de bancos de dados de imagens, que visam permitir relacionar imagens a características de conteúdo, formas e textura. Neste contexto, os sistemas tipicamente pré-processam cada imagem para extrair tais características, que são então armazenadas junto com a própria imagem, a fim de acelerar o processamento das consultas.

Os enfoques de recuperação de imagens baseada em conteúdo têm se concentrado em duas direções [GR95]. Na primeira, algumas características principais da imagem são extraídas manualmente e armazenadas textualmente em um banco de dados (por exemplo, dados sobre uma região e fenômenos retratados na imagem), sendo a recuperação processada a partir de consultas sobre tais dados textuais. O segundo enfoque é baseado em técnicas de processamento de imagem, que realizam extração de características e determinação de objetos de interesse. O primeiro enfoque é mais rápido, pois pode aproveitar técnicas de otimização de consultas em bancos de dados tradicionais, mas limita o tipo de consulta a ser feito. O segundo é mais flexível, mas computacionalmente caro. Um sistema de consultas a imagens deveria, idealmente, combinar os dois tipos de tratamento.



## 6.3 A Linguagem LEGAL

Esta seção descreve uma linguagem de consulta e manipulação espacial, denominada LEGAL - Linguagem Espacial para Geoprocessamento Algébrico [Cam95].

LEGAL provê um ambiente geral para análise geográfica, estendendo SQL com operadores espaciais e facilidades para apresentação de geo-campos e geo-objetos. A linguagem é fortemente tipada, ou seja, cada objeto pertence a uma das classes introduzidas pelo modelo ou pelo esquema conceitual do banco de dados geográfico em questão. LEGAL oferece ainda construções específicas para a manipulação dos resultados de uma consulta dentro de um programa, baseadas no modelo definido pelo ODMG (“Object Data Management Group”) [Cat94, Cam95]. A linguagem permite ainda a aplicação individualizada das operações sobre geo-campos e geo-objetos definidas no capítulo 5.

### 6.3.1 Definição de Esquemas Conceituais

Em LEGAL o projetista usa o comando **CREATE** para especificar o esquema conceitual de um banco de dados geográfico definindo especializações das classes básicas do modelo descrito no capítulo 4. Para cada nova classe, o projetista poderá criar novos atributos, redefinir os domínios dos atributos introduzidos pelo modelo e definir os valores dos atributos de classe.

Por exemplo, o seguinte comando define a estrutura da classe **Uso-do-Solo** como uma especialização da classe **TEMATICO**:

```
CREATE Uso-do-Solo
    (nome CHAR(20),
     ano  INTEGER)
IS-A Tematico
    (temas={"Floresta Primaria",
           "Cerrado",
           "Floresta Secundaria"})
```

Intuitivamente, cada instância da classe **Uso-do-Solo** será um geo-campo temático com dois atributos convencionais próprios: **nome**, uma cadeia de 20 caracteres e **ano**, um inteiro. Já **temas** é um atributo de classe para **Tematico** que, neste caso, terá como valor o conjunto “Floresta Primaria”, “Cerrado”, “Floresta Secundaria”, que corresponde ao contradomínio do mapeamento de cada geo-campo em **Uso-do-Solo**.

De forma semelhante, o projetista pode definir uma classe de geo-objetos chamada **Hospital**, com três atributos próprios, através do comando:

```
CREATE Hospital
    (nome          CHAR(30),
     num-leitos    INTEGER,
```

```

        receita-SUS FLOAT)
IS-A GeoObjeto

```

Note que o tipo do atributo de localização de `Hospital` e `Uso-do-Solo` é herdado das superclasses correspondentes.

### 6.3.2 Criação de Geo-Objetos e Geo-Campos

Através do comando `INSERT`, o usuário cria novos objetos no banco de dados, sejam eles geo-campos, geo-objetos ou representações destes objetos. Por exemplo, uma instância de `Hospital` pode ser criada através do comando `INSERT`:

```

INSERT (nome = "Hospital Municipal SJC",
        num-leitos = 300,
        receita-SUS = 50.000,00)
IN Hospital

```

e uma instância de `Uso-do-Solo` através de:

```

INSERT (nome = "acao antropica 1990",
        ano = 1990)
IN Uso-do-Solo

```

O comando `INSERT` oferece ainda a possibilidade de criar ao mesmo tempo um geo-campo e a sua representação. Por exemplo, o seguinte comando cria uma instância de `Uso-do-Solo` representada por uma matriz:

```

INSERT (nome = "acao antropica 1990",
        ano = "1990")
IN Uso-do-Solo
REPRESENTED BY
(nome = "Acao Antropica 90 - Rio-Novo",
        resolucao-X = 100,
        resolucao-Y = 100,
        escala = 1:250.000)
IN Matricial

```

O novo geo-campo será uma instância da classe `Uso-do-Solo`, com nome “acao antropica 1990” e ano igual a “1990”. Este geo-campo terá uma representação matricial, onde os valores dos atributos pré-definidos são um nome “Acao Antropica 90 - Rio-Novo”, uma resolução em X e Y de 100m, uma escala de 1:250.000 e uma região (a definição de regiões é feita através de um comando especial, não discutido neste texto – ver [Cam95]).

### 6.3.3 Consultas

Uma consulta em LEGAL possui dois componentes: uma expressão de busca expressa em SQL estendido e uma resposta à consulta, que pode ser objeto de manipulação posterior.

O usuário pode formular consultas envolvendo classes tanto de geo-objetos quanto de geo-campos. No caso de geo-campos, dada a sua natureza, os operadores disponíveis em LEGAL permitem apenas recuperação baseada em atributos convencionais. Para geo-objetos, LEGAL oferece funções e relacionamentos espaciais, computadas sobre suas localizações.

Por exemplo, a consulta  $Q_1$  – “*Selecione os nomes de todos os municípios mineiros a menos de 50 km da cidade de Belo Horizonte*” – é formulada em LEGAL através do comando:

```
SELECT m.nome
      FROM m Municipio, c Cidade
      WHERE c.nome = "Belo Horizonte"
             and distance(m,c) < 50
```

As classes `Municipio` e `Cidade` são especializações de `GeoObjeto` e contêm como instâncias geo-objetos representando os municípios e cidades mineiros. Supõe-se que os atributos `nome` e `localizacao` tenham sido definidos para ambas as classes. A qualificação da consulta implicitamente indica que a função espacial `distance` deve operar sobre tais localizações, já que esta função possui como argumentos apenas as variáveis `m` e `c` varrendo respectivamente as instâncias de `Municipio` e `Cidade`.

Para geo-objetos, LEGAL oferece ainda a possibilidade das funções e relacionamentos espaciais serem computadas sobre localizações dadas por uma representação particular destes geo-objetos (ver seção 4.2.3), que passam a ter precedência sobre localizações definidas nos próprios geo-objetos. Esta opção é sinalizada pela cláusula `REPRESENTED BY`. Por exemplo, a consulta  $Q_1$  pode ainda ser formulada em LEGAL como:

```
SELECT m.nome
      FROM m Municipio, c Cidade
      WHERE c.nome = "Belo Horizonte"
             and distance(m,c) < 50
      REPRESENTED BY
      (SELECT p
        FROM p Mapa
        WHERE p.nome = "Minas Gerais")
```

As instâncias de `Mapa` são mapas dos estados brasileiros, com seus municípios e cidades, e também possuem o atributo `nome`.

A construção `REPRESENTED BY` indica que as localizações dos geo-objetos em `Municipio` e `Cidade` deverão ser obtidas da representação de nome `Minas Gerais`, recuperada pela consulta aninhada sobre `Mapa`. Esta construção afeta portanto a semântica da função espacial `distance`.

A cláusula `REPRESENTED BY` pode ainda ser associada a cada classe de objetos individualmente, como no exemplo abaixo:

```
SELECT m.nome
FROM   m Municipio REPRESENTED BY
      (SELECT p FROM   p Mapa-Municipio
      WHERE p.nome = "Minas Gerais")
      c Cidade   REPRESENTED BY
      (SELECT n FROM   n Mapa-Cidade
      WHERE n.nome = "Minas Gerais")
WHERE  c.nome = "Belo Horizonte"
      and distance(p,c) < 50
```

As classes `Mapa-Municipio` e `Mapa-Cidade` são duas especializações de `RepComplexa`. Suas instâncias são mapas dos estados brasileiros com seus municípios e cidades, respectivamente.

Por fim, observamos que nada impede que consultas em `LEGAL` envolvam também geo-campos e suas representações. Outros exemplos podem ser encontrados em [Cam95].

### 6.3.4 Manipulação de Objetos e Coleções

Conforme mencionado na introdução desta seção, `LEGAL` baseia-se no modelo definido pelo ODMG. Este modelo permite a definição de *coleções*, que são conjuntos ordenados de objetos de um mesmo tipo. Permite ainda manipular os objetos de uma coleção através da definição de um *iterador*, ao qual podem ser aplicados os operadores `next`, `first`, `last`, `more?`, `reset` e `delete`. O operador `next` posiciona o iterador sobre o próximo elemento da coleção. Após `next` ter sido aplicado ao último elemento da coleção, o valor do iterador será `nil`. A chamada inicial a `next` retorna o primeiro elemento da coleção, se o parâmetro `iteration-order` do iterador for `forward` (tomado como default), ou o último elemento, se `iteration-order` for `backward`. Os outros operadores são definidos de forma semelhante.

Para ilustrar estes construtores, considere novamente a consulta  $Q_1$ , formulada agora como um programa que cria a coleção dos pares  $(m, d)$  tais que  $m$  é um geo-objeto em `Municipio` a menos de 50 km da cidade da cidade de Belo Horizonte e  $d$  é a distância do geo-objeto a Belo Horizonte:

```
//
// definicao da colecao e do iterador
//
COLLECTION md (Municipio, FLOAT);
ITERATOR   i   OVER md;
//
// construcao da colecao
```

```
//
md := (SELECT (m, distance(m, c))
        FROM  m Municipio, c Cidade
        WHERE c.nome = "Belo Horizonte"
              and distance(m,c) < 50);

//
// interacao com o usuario
//
i := next(md);
WHILE i <> nil DO
BEGIN
    IF "usuario nao aceitou o objeto i"
        THEN delete(i);
    i := next(md);
END
```

O comando `COLLECTION` define uma variável, `md`, que recebe como valores uma coleção de pares ordenados, onde o primeiro elemento é um objeto definido da mesma forma que as instâncias de `Municipio` e o segundo, um real representado em ponto flutuante. O comando `ITERATOR` define um iterador `i` sobre `md` cujo parâmetro `iteration-order` é, por default, `forward`. O comando `SELECT` cria `md` como uma coleção de pares, onde o primeiro componente origina-se de `Municipio` e o segundo é um ponto flutuante (resultado da função `distancia`). Por fim, o comando `WHILE` varre o conjunto, elemento a elemento.

`LEGAL` permite também definir variáveis de programa para conter a resposta de consultas que retornam apenas um objeto. Por exemplo, o seguinte programa coloca na variável `t1` o geo-objeto correspondendo à cidade de Belo Horizonte:

```
VARIABLE t1 Cidade;
t1 := (SELECT c
        FROM  Cidade c
        WHERE c.nome = "Belo Horizonte");
```

O comando `VARIABLE` indica que a variável de programa `t1` terá como valor um objeto definido da mesma forma que as instâncias de `Cidade`. A definição de variáveis de programa torna mais confortável a aplicação de operadores a geo-campos e a geo-objetos individuais.

Concluimos esta seção exemplificando a utilização em `LEGAL` dos operadores do capítulo 5. Considere o banco de dados geográfico `AGRICULTURA` contendo as seguintes classes de geo-campos definidos sobre a mesma região:

```
CREATE Chuva      (nome CHAR(20)) IS-A Numerico;
CREATE Topografia (nome CHAR(20)) IS-A Numerico;
CREATE Solo       (nome CHAR(20))
```

```

IS-A Tematico (temas={"Le","Aq",...});
CREATE Aptidao (nome CHAR(20))
IS-A Tematico (temas={"Boa","Media","Baixa"});

```

O programa abaixo ilustra simplificadaamente o uso do operador **BOOLEAN** para determinar a aptidão agrícola da região abrangida pelo banco AGRICULTURA:

```

//
VARIABLE campo-chuva   Chuva;
VARIABLE campo-topo    Topografia;
VARIABLE campo-solos   Solo;
VARIABLE campo-aptidao Aptidao;
//
campo-chuva  := (SELECT c FROM c Chuva
                  WHERE c.nome = "Chuva92");
campo-topo   := (SELECT t FROM t Topografia
                  WHERE t.nome = "Topo92");
campo-solos  := (SELECT s FROM s Solo
                  WHERE s.nome = "Solos92");
//
campo-aptidao:= BOOLEAN
                ( "Boa"      : campo-solo.classe = "Le"  E
                  campo-chuva >= 1000      E
                  campo-topo <= 1500;
                  "Media"    : campo-solo.classe = "Aq"  E
                  campo-chuva >= 600       E
                  campo-topo <= 1000;
                  "Baixa"    : OTHERWISE )
//
nome.campo-aptidao := "Aptidao 92";
//
INSERT campo-aptidao IN Aptidao

```

O operador **BOOLEAN**, apesar do nome, cria um novo geo-campo a partir de outros. Ele exige que seja especificado um conjunto de condições a serem satisfeitas para cada tema do geo-campo de saída, conforme o exemplo acima, onde um geo-campo de aptidão de solos é calculado, a partir dos valores de chuvas, topografia e tipo de solo.



# Capítulo 7

## Apresentação de Dados Geográficos

### 7.1 Introdução

A apresentação de dados geográficos é um aspecto fundamental da interface de um SIG. O capítulo 6 discutiu a interação de um usuário com um SIG do ponto de vista de facilidades de consulta. Este capítulo complementa tal discussão, tratando dos aspectos de *apresentação* do resultado de uma consulta. A seção 7.2 aborda a questão da visualização de dados geográficos, destacando a apresentação de mapas, e exemplifica como tais problemas são tratados na linguagem LEGAL, proposta no capítulo 6; a seção 7.3 dá uma visão geral do projeto de interfaces para SIG.

Estudos de apresentação de dados geográficos se concentram em aspectos 2D, com poucas referências a dados em 3D [RK91]. Muitos dos trabalhos estão relacionados a linguagens [Ege89, Ege94], enquanto que apenas recentemente surgiu o interesse em arquiteturas de interface [Voi91, Voi94, OM95] e em aspectos cognitivos da apresentação [Mon91, Kuh91, LS92, Kuh93, Gou93, Woo93]. Atualmente, aspectos relacionados à apresentação vêm também sendo tratados no contexto de visualização científica [Jai93].

### 7.2 Visualização de Dados Geográficos

#### 7.2.1 Considerações Gerais

A interface de um SIG deve oferecer pelo menos as seguintes funções [Ege89]. Primeiro, deve oferecer facilidades para visualização de campos e objetos geográficos nos espaços 2D e 3D, incluindo comandos para controlar padrões de preenchimento. A interface deve ajudar o usuário a formular consultas utilizando a informação gráfica presente na tela. Deve, ainda, prover formas de combinar o resultados de várias consultas e de construir legendas que ajudem o usuário a examinar o conteúdo apresentado.

Interfaces devem também permitir especificar a construção de uma apresentação. Em SIGs,



isto significa considerar não apenas o que foi especificado na consulta, mas também informação adicional que inclua geo-campos e geo-objetos de interesse na região recuperada pela consulta, a partir do contexto da aplicação e do perfil do usuário. O sistema de interface deve, assim, dispor de uma grande gama de facilidades para definir apresentações e, inclusive, auxiliar o projetista a definir regras que permitam especificar a apresentação adequada.

O volume dos dados a apresentar e a extensão geográfica por eles coberta muitas vezes impedem a apresentação direta do resultado de uma consulta por inteiro. Duas técnicas podem ser utilizadas para ampliar o espaço de apresentação [Koi95]: ampliação física ou redução da informação. No primeiro caso, a tela do terminal é aumentada usando dispositivos como telas virtuais ou projetores de vídeo. No segundo caso, o mais encontrado em SIG, são oferecidas funções de interface gráfica (*scrolling*, *panning*), ou funções do próprio SIG (tais como possibilidade de mudança de escala ou funções de análise estatística). O espaço de apresentação pode também ser expandido pelo uso do conceito de hipertexto, em que as âncoras conectam dados semanticamente relacionados, ligando tanto textos descritivos quanto mapas. Em alguns casos, as âncoras ligam não apenas dados estáticos, mas permitem ativar funções que calculam correlações espaciais entre dados geo-referenciados [BK95]. Ainda, o resultado de uma consulta pode compreender diversos espaços geograficamente distintos.

Como visto no capítulo 6, existem várias formas de se realizar consultas em um SIG. O sistema de visualização deve se preocupar não apenas em mostrar resultados, mas também em oferecer ao usuário facilidades para manipular os elementos visualizados de forma a construir novas consultas. Assim, uma interface para SIG deve considerar ao menos 3 níveis de visualização [LG93]:

- visualização de metadados;
- visualização dos componentes usados para construir consultas (tanto resultados de consultas anteriores quanto operadores e funções que podem ser aplicados a tais resultados);
- visualização do resultado propriamente dito, com resoluções múltiplas.

A visualização de metadados visa facilitar a construção de consultas e permitir estabelecer correlações entre conjuntos de dados em um nível mais abstrato (por exemplo, permitindo relacionar classes de geo-objetos e de geo-campos). Novas propostas de visualização destes relacionamentos envolvem grafos em duas ou três dimensões [ea93, BFG93]. Nos SIGs atuais, os metadados são em geral limitados ao esquema do banco de dados geográfico e são via de regra apresentados de forma tabular.

A visualização com resoluções múltiplas pode requerer apresentação de informação contextual que varie conforme a resolução. Considere, por exemplo, a seguinte consulta: “Onde está localizada a UNICAMP?”. Como visto nos capítulos anteriores, o processamento desta consulta corresponde a encontrar a geo-região onde está localizado o geo-objeto UNICAMP.

A figura 7.2.1 mostra três dentre várias apresentações possíveis. A primeira indica que a UNICAMP está no Estado de São Paulo; a segunda usa a cidade de Campinas como referência

dentro do Estado; a terceira utiliza um mapa esquematizado da cidade. Estas apresentações diferem pela escala, formas, legendas, ícones e pela informação contextual fornecida. Na terceira apresentação, por exemplo, são incluídas vias de acesso e o aeroporto de Viracopos, embora estes dados não houvessem sido solicitados.

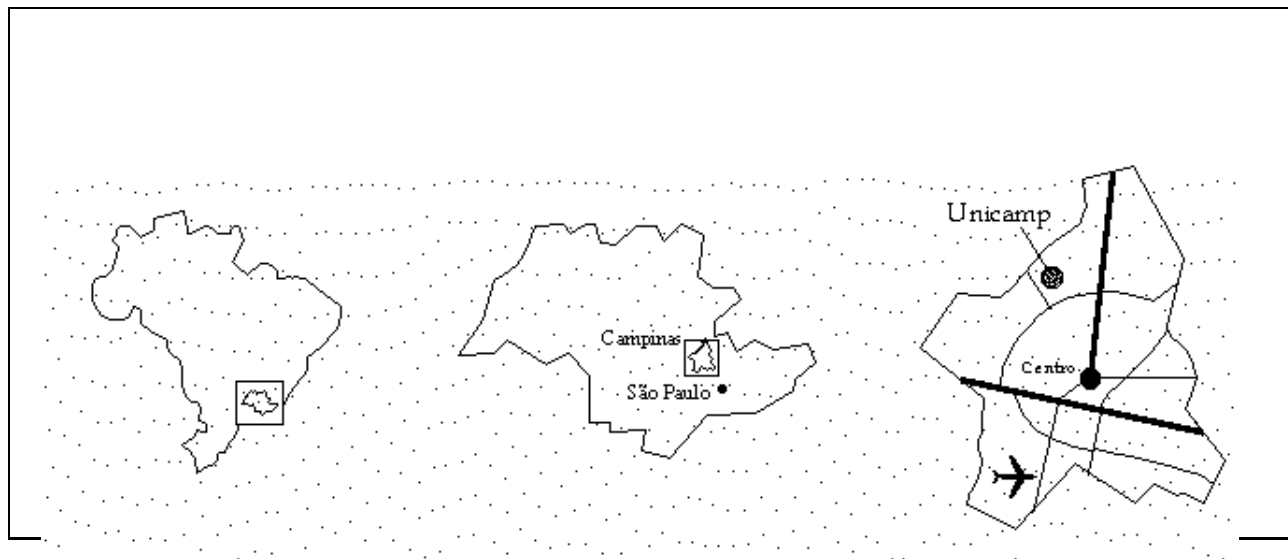


Figura 7.1: Apresentações de uma mesma consulta em resoluções diferentes

A visualização de dados georeferenciados é complicada pelas questões de *representação* e o tratamento de *relacionamentos entre os dados*. Em aplicações tradicionais, representações múltiplas inexistem e os relacionamentos entre os dados são estabelecidos durante a modelagem de dados. No caso de sistemas georeferenciados, uma determinada representação pode ser apresentada de diversas formas a fim de atender aos requisitos do usuário (quer por exigências formais do domínio, quer por necessidades informais ligadas a aspectos cognitivos ou culturais). Além disto, a maioria dos relacionamentos espaciais é calculada a partir do contexto e da representação adotados. Como visto no capítulo 4, o resultado de uma consulta depende da representação sobre o qual é calculada, que por sua vez determina como e quais relacionamentos espaciais podem ser calculados.

A complexidade do problema pode ser exemplificada a partir da figura 7.2.1, retirada de [Arn93], que retrata duas formas de apresentar concentrações populacionais em uma região urbana, cada uma das quais permitindo ao usuário deduzir relacionamentos espaciais particulares para uma mesma representação subjacente. A parte superior de cada figura mostra a distribuição do número de habitantes por quarteirão na zona coberta pela linha AB; a parte inferior mostra a distribuição da densidade populacional. No primeiro caso, a apresentação é feita sob forma de mapa temático, classificando as regiões por valor de densidade; no segundo caso, a apresentação utiliza isolinhas.

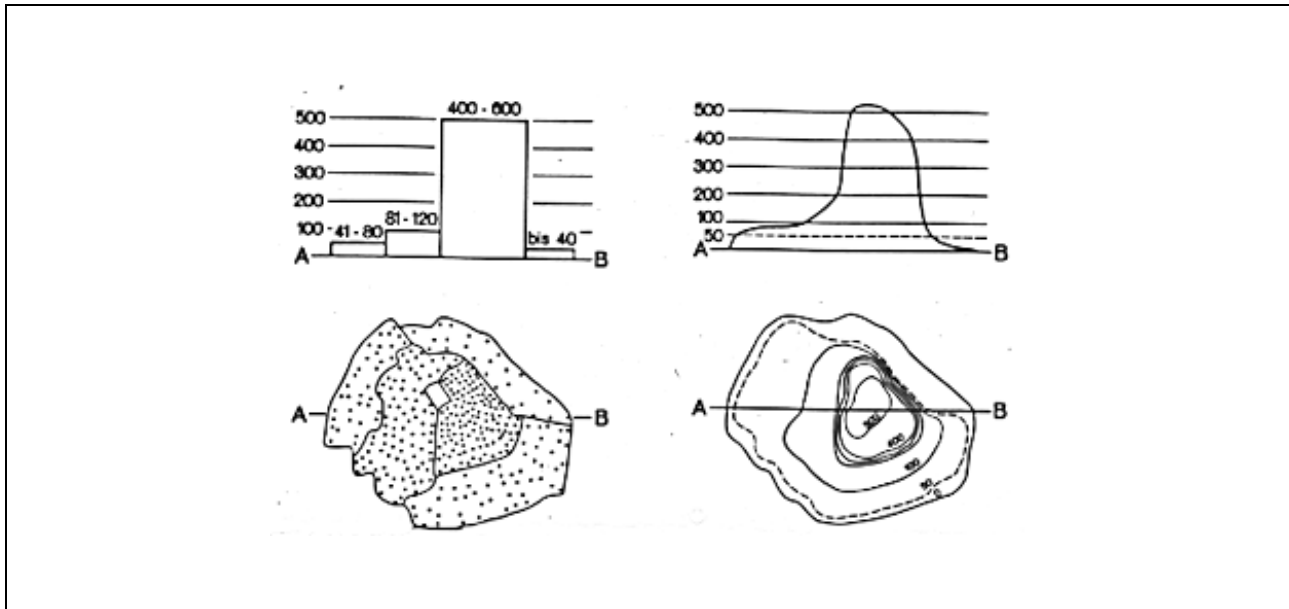


Figura 7.2: Apresentações distintas de uma mesma representação

### 7.2.2 Produção Cartográfica

A apresentação de mapas é uma função importante em um SIG. De fato, muitas aplicações têm por objetivo produzir cartas eletrônicas [Goo93, BP94]. A produção cartográfica digital adapta os procedimentos manuais de geração de mapas em papel à produção de mapas por computador. A automatização destes procedimentos manuais não é simples. Como ressaltado por [Oli95a], estudos recentes de produção cartográfica digital mostram que cores e símbolos utilizados nas cartas em papel não são diretamente transportáveis para o meio digital. Esta automatização é efetuada através de funções de análise e estatística espacial, envolvendo técnicas de computação gráfica, processamento de imagens e, para casos mais sofisticados, morfologia matemática [Ber93, FR94]. Os algoritmos de apresentação precisam levar em conta também fatores como a resolução do dispositivo de saída e as distorções provocadas por operações como rotação ou translação.

Estas observações são resumidas pela tabela 7.1 e que compara as facilidades de apresentação cartográfica usando SIGs às facilidades da cartografia tradicional:

Como analisado por Monmonier [Mon91], mapas são construídos a partir de três elementos básicos – escala, projeção e símbolos – que são fontes de distorção. A apresentação deve levar em consideração tais distorções para tentar reduzir o erro na produção de um mapa. De fato, as regras de produção cartográfica variam com os objetivos da aplicação. Cartas terrestres, por exemplo, têm características distintas de cartas náuticas. Estas últimas, por sua vez, diferem na apresentação do mesmo conjunto de fenômenos: cartas para operações com minas, por exemplo, exigem detalhamento do fundo do mar e do relevo da plataforma continental, enquanto que cartas para operações anfíbias requerem detalhamento das praias, com gradiente da inclinação e natureza do litoral [Oli95a].

Cartografia tradicional	Cartografia usando SIG
Cartas limitadas a aspectos gráficos	Cartas com recursos multimeios
Imagem estática	Imagem dinâmica
Documento único	Múltiplos documentos interligados
Carta destinada a um público grande	Carta destinada a necessidades individuais
Carta destinada a comunicar um conhecimento	Carta utilizada para descobrir novas informações, usando mecanismos de visualização e exploração
Carta usada para leitura	Carta usada para comunicação interativa
Carta produzida por cartógrafo	Carta produzida por usuário, não necessariamente cartógrafo

Tabela 7.1: Comparação entre cartografia tradicional e cartografia usando SIG

[Ber93] identifica quatro categorias principais de operações primitivas para produção cartográfica: reclassificação de categorias, sobreposição (*overlay*) de dois ou mais mapas, cálculo de vizinhança e medidas de distância e conectividade. Estas operações correspondem a grosso modo às operações de mesmo nome sobre geo-campos e geo-objetos do seção 5.3, onde cada operação descrita naquele capítulo seria a seguir complementada por uma função de visualização do resultado.

Operações de *reclassificação* reorganizam os dados de um mapa de forma a criar um novo mapa, onde cada faixa de valores (classe) é visualizada variando, na apresentação, atributos de cor, textura e outros. A determinação dos valores destes atributos de visualização é função não apenas das faixas estabelecidas pela consulta, mas também de sua posição, contigüidade e geometria. Em outras palavras, não se trata apenas de atribuir diferentes cores às distintas faixas, mas de fazê-lo de forma a permitir ao usuário distinguir as diferenças entre elas.

Operações de *sobreposição* envolvem vários mapas e criam novas regiões que resultam da combinação, para cada posição geográfica, dos valores dos mapas sobrepostos naquela posição, que são distinguidos por modificações de cor e textura. Os limites das regiões podem ser indicados explicitamente (através de linhas poligonais) ou deduzidos por diferença de cor. Processo semelhante de apresentação ocorre em operações de *vizinhança*. Estas geram mapas que indicam zonas de influência de algum tipo de fenômeno, indicadas através de variação de cores ou uso de polígonos.

Operações de *distância e conectividade* geram mapas que em geral apresentam traçado de rotas. A indicação de rotas de emergência ou o planejamento de vias de escoamento de produtos são exemplos deste tipo de situação, em que as rotas resultantes da consulta são apresentadas em destaque – por exemplo, usando cor – sobre um pano de fundo que é a malha viária da região.

A linguagem cartográfica utiliza 7 variáveis visuais [BP94]: forma, tamanho, cor, valor, textura, grão e orientação. Forma, tamanho e cor são os fatores cuja variabilidade melhor permite ao usuário entender o resultado de uma consulta.

Para informações qualitativas sobre objetos pontuais ou lineares, a apresentação deve ser feita variando a forma, a orientação ou a cor; se as informações são sobre objetos poligonais, varia-se a cor e o grão. Se, por outro lado, a informação é quantitativa, varia-se o tamanho se o objeto é pontual ou linear, enquanto que para objetos poligonais varia-se o tamanho, o valor, a cor ou o grão.

Há uma infinidade de *formas* disponíveis na cartografia eletrônica, tanto geométricas quanto simbólicas. A variável forma só permite ao usuário determinar informação qualitativa, precisando estar associada a outras variáveis (cor, tamanho, textura) para permitir derivar informação quantitativa. Em uma carta náutica, por exemplo, o estilo e espessura de uma linha permitem determinar o tipo de fenômeno apresentado: batimetria, região de visibilidade de um farol, contorno de ilhas [Oli95a]. Formas podem estar associadas a pontos (ícones), linhas ou polígonos. Neste último caso, o termo *forma* é substituído por *textura*.

A variação do *tamanho* dos componentes de uma carta permite avaliar suas grandezas relativas e absolutas. Na prática, esta variação é limitada pelas dimensões físicas observáveis pelo olho humano (por exemplo, o menor círculo perceptível tem um diâmetro de 0.2 mm [BP94]) e pelo conjunto de dados apresentados (por exemplo, havendo um grande conjunto de fenômenos a serem apresentados, o uso do tamanho para avaliar a ordem de grandeza relativa de um fenômeno pode inviabilizar a apresentação de outros fenômenos na região). Se um conjunto de cidades é apresentado por meio de pontos (forma) em um mapa, a variação do tamanho destes pontos permite ao usuário visualizar sua importância relativa em termos de população.

Dentre as sete variáveis visuais, a *cor* é a que melhor permite separar os fenômenos de natureza diferente. Todas as cartas de inventário (pedológicas, climáticas e outras) se baseiam na cor como meio de transmitir informação qualitativa. A variação de tom e contrastes de cores possibilita ao usuário determinar diferenças relativas entre os valores de um fenômeno. Por exemplo, em cartas náuticas, diferenças de tons de azul indicam a variação na profundidade; em mapas de temperatura, cores mais próximas do vermelho denotam temperaturas mais quentes. A apresentação exige também que haja um contraste suficiente entre cores de regiões contíguas, para permitir determinar seus limites.

Um *valor* pode ser apresentado de forma textual ou gráfica (combinando forma, cor e tamanho). Neste último caso, a informação obtida é relativa ao contexto da carta. Tramas diferentes (por exemplo, hachuras de larguras distintas) são usadas para passar ao usuário a noção de gradação de valores.

*Legendas alfanuméricas* ajudam a mostrar as diferenças relativas entre os dados, conforme o tipo, tamanho e outras características do fonte tipográfico. Além de permitir comparação entre as dimensões de fenômenos a orientação e inclinação do fonte de uma legenda tornam possível visualizar a geometria de um objeto. Os problemas de colocação automática de legendas e de texto em cartas digitais são descritos em [DF92].

Como a apresentação em escala 1:1 é em geral inviável, qualquer apresentação cartográfica corresponde a uma redução da realidade, efetuada a partir das chamadas operações de *generalização*. Recordando, o processo de generalização consiste em extrair e resumir as informações desejadas de forma a destacar em uma carta os pontos de interesse, em função dos objetivos, tema, escala e particularidades da região retratada. A generalização cartográfica é erroneamente considerada como implementável de forma automática a partir de operações matemáticas de mudança de escala. Muito pelo contrário, a automação completa de procedimentos de generalização é raramente bem sucedida, pois estes exigem envolvimento do usuário quanto a decisões de apresentação. A figura 7.2.2 mostra exemplos de operações de eliminação de detalhes como parte dos procedimentos de generalização: a parte (a) indica a generalização de uma edificação, enquanto a parte (b) mostra a generalização de um curso d'água. Em nenhum dos dois casos há redução de escala, mas no entanto o usuário responsável pela produção decidiu eliminar tais detalhes (por exemplo, por razões de clareza).

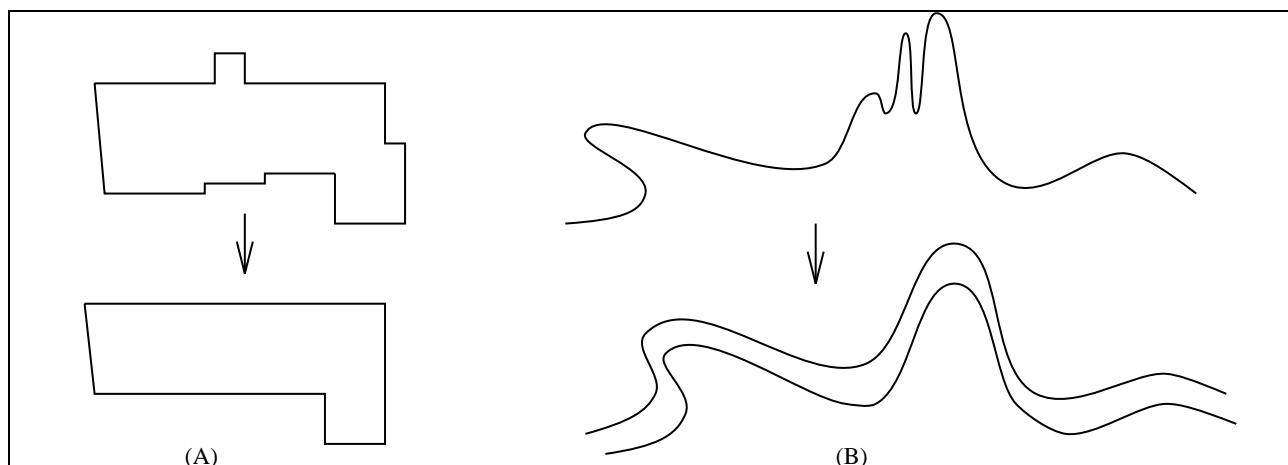


Figura 7.3: Exemplos de generalização

A generalização se aplica em dois contextos: no processamento numérico dos dados (definindo valores derivados e agregando dados) e na apresentação (relacionando visualmente as características relevantes). As principais operações envolvidas são [BP94]: seleção, esquematização e harmonização. A *seleção* consiste em escolher os elementos que comporão a carta, que serão transformados em um “pano de fundo” gráfico para permitir visualização do contexto (por exemplo, topografia, limites administrativos e outros). A *esquematização* corresponde a simplificar e eliminar detalhes da apresentação, como por exemplo na figura 7.2.2. Segundo [BP94], a esquematização pode ser estrutural (correspondendo a uma simplificação gráfica usando algoritmos predefinidos) ou conceitual (ligada à semântica da carta). A primeira pode ser automatizada usando operações como interpolação e eliminação de valores, enquanto que a segunda exige intervenção humana. A *harmonização* consiste no conjunto de etapas iterativas que combinam seleção e esquematização para produzir o resultado final.

### 7.2.3 Apresentação de Dados em LEGAL

O capítulo 6 descreveu de uma forma geral linguagens de consulta para SIGs, terminando com o exemplo da linguagem LEGAL, que satisfaz vários dos requisitos enunciados. Esta linguagem foi descrita em alto nível, do ponto de vista de especificação de esquemas e formulação de consultas. Um outro componente de LEGAL se refere à *visualização* dos dados de uma consulta. LEGAL permite definir as variáveis visuais cor, valor, textura, forma e tamanho, cinco das sete variáveis possíveis.

Neste contexto, pode-se considerar a formulação de uma consulta LEGAL como composta de duas partes: especificação do resultado desejado e especificação de sua apresentação. A linguagem de apresentação de LEGAL permite definir mecanismos para controle da visualização dos dados, a partir da cláusula **defina visual**, cujos parâmetros variam de acordo com a classe de geo-campos ou geo-objetos apresentada. Os parâmetros incluem fatores como cor, estilo e padrão de preenchimento. Diferentes formas e símbolos podem igualmente ser associados aos geo-objetos selecionados em uma consulta. Além disso, resultados de consultas distintas podem ser combinados em uma única apresentação utilizando, na definição da apresentação, o parâmetro **overlay** [Cam95].

A definição dos atributos de uma apresentação pode ser feita de duas formas (complementares). Uma delas consiste em definir características default de apresentação para todos os objetos de uma geo-classe, tais como padrão, cor, textura e símbolo para todos os objetos de uma geo-classe usando a cláusula **defina visual**. Neste caso, a cláusula **show** permite visualizar os objetos dessas classes de acordo com o especificado. Alternativamente, o usuário pode modificar interativamente algumas destas características da visualização.

Suponha por exemplo que a apresentação dos geo-objetos da classe **Município** do capítulo 6 tenha sido definida por

```
DEFINA VISUAL PARA CLASSE Município
    COR azul
    PADRAO hachura
    HACHURA 30
    TEXTO AUTOMATICO
FIM
```

Dada a consulta já vista no capítulo 6, “Selecione os nomes de todos os municípios a menos de 50km da cidade de Belo Horizonte”, o resultado da consulta será apresentado usando estes parâmetros – polígonos preenchidos por hachuras em azul, inclinadas a 30 graus, com o nome de cada polígono (nome do município) colocado automaticamente.

Alternativamente, o resultado pode ser armazenado em alguma classe temporária através de atribuição:

```
mun_minas:=  
    SELECT m.nome  
    FROM m Municipio, c Cidade  
    WHERE c.nome = "Belo Horizonte"  
    and distance(m,c) < 50
```

e a apresentação ser solicitada posteriormente, usando a cláusula **show**:

```
SHOW mun_minas  
AS SHADING  
WHERE SHADING_COLOUR = amarelo
```

Neste caso, os municípios que satisfazem a restrição da consulta dada serão mostrados como polígonos (parâmetro **shading**), com hachuras inclinadas a 30 graus, porém na cor amarela. Os outros objetos da classe **Municipio**, no entanto, serão visualizados na cor azul (default).

## 7.3 Projeto de Interfaces

A primeira parte do capítulo discutiu problemas de apresentação. Estes problemas estão, naturalmente, intimamente ligados a questões de projeto da interface. Há ainda poucos trabalhos voltados ao projeto de interfaces para SIGs, apesar da importância da interface na aceitação de um produto. Isto se deve basicamente a dois fatores: problemas intrínsecos à natureza dos dados geográficos e problemas relativos ao projeto de interface em SIGs. Por um lado, as aplicações geográficas apresentam desafios que não são respondidos adequadamente pelas propostas de interfaces existentes (como, por exemplo, a visualização cartográfica e a apresentação de representações múltiplas de um mesmo objeto); por outro lado, a própria área de projeto e desenvolvimento de interfaces envolve inúmeros problemas em aberto, sobre os quais ainda não existe consenso.

Os principais resultados na área de interfaces para SIGs começaram a aparecer na década de 90, quando problemas nos níveis inferiores da arquitetura de um SIG (manipulação e armazenamento de dados, vide capítulo 8) passaram a ser melhor equacionados. Nesses estudos, a interface é considerada como uma camada que entre o usuário e o SIG subjacente. Um tutorial sobre trabalhos na área [Oli95b] identificou três tipos de enfoque para projeto de interfaces em SIG: arquitetura, linguagem e fatores humanos, mostrados na figura 7.3.1.

O enfoque relativo a arquitetura se ocupa em definir a interface funcionalmente, como um conjunto de blocos que se comunicam entre si e com o SIG subjacente. Cada bloco é responsável por um conjunto de tarefas (por exemplo, operações de gerenciamento de janelas ou de apresentação de dados).

O enfoque em linguagens se preocupa em definir e implementar uma linguagem utilizada pelo usuário para definição e manipulação dos dados. A interface possui, neste caso, mecanismos que



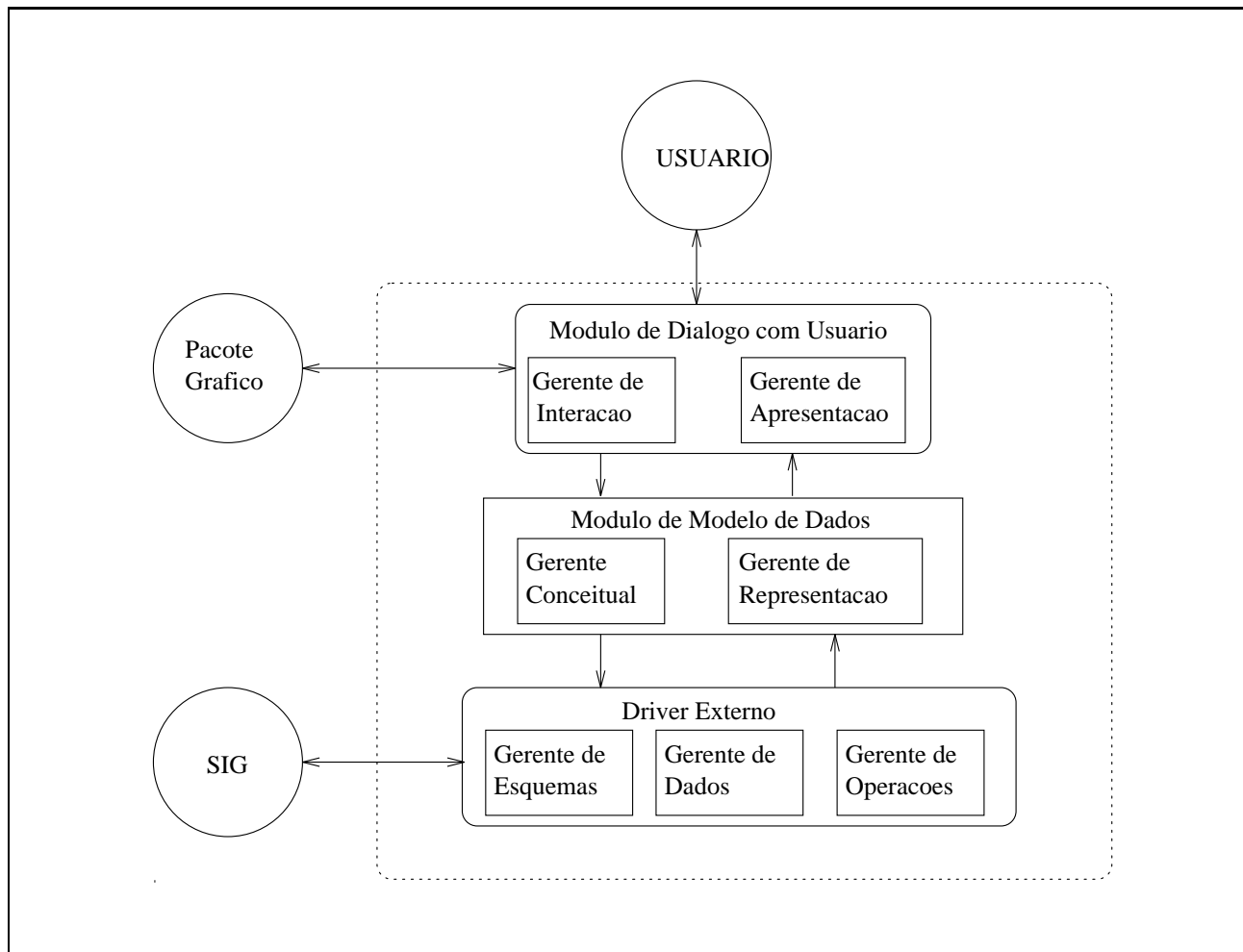


Figura 7.4: Exemplo de arquitetura de uma interface aberta para SIG

traduzem comandos do usuário na linguagem proposta (na figura,  $L_e$ ) em comandos para o SIG subjacente ( $L_i$ ). A linguagem LEGAL abordada no capítulo 6 é um exemplo de  $L_e$ , enquanto que alguns detalhes de  $L_i$  serão abordados no contexto de processamento de otimização de consultas (vide capítulo 10).

O enfoque em fatores humanos estuda o modelo mental do usuário, para, a partir deste, especificar o modelo de interface apropriado, buscando minimizar o esforço do usuário em definir e manipular dados georeferenciados.

A seguir, cada um destes enfoques é discutido em mais detalhes.

### 7.3.1 Arquitetura

Este enfoque se preocupa principalmente com dois itens:

- especificação dos blocos que compõem a arquitetura da interface, suas funções, suas responsabilidades e suas interfaces;
- definição de mapeamentos para otimizar a comunicação entre a interface e o SIG subjacente (traduzindo o modelo da interface para o modelo do SIG e vice-versa).

A arquitetura da interface deve ser considerada sob dois pontos de vista: sua comunicação com o resto do sistema (no caso, o SIG) e a interação entre os módulos que compõem a interface propriamente dita. Neste último ponto de vista, o sistema de interface pode ser baseado em arquiteturas diretamente implementáveis ou arquiteturas de “toolbox” (vide por exemplo, [Voi91, Voi94]). Cada conjunto de funções é agrupado em um módulo: visualização, manipulação de mapas, interação com o banco de dados e outros. Nosso modelo de dados, descrito no capítulo 4, serve de base à arquitetura de interface de [OM95], mostrada na figura 7.3. Esta arquitetura prevê três níveis de blocos funcionais: diálogo com o usuário, modelo de dados e mapeamento. O primeiro nível é responsável pela visualização e interação com o usuário. O nível intermediário permite ao usuário utilizar as noções multinível do modelo, considerando geo-classes e comportando múltiplas representações. O nível de mapeamento é responsável pela comunicação entre a interface e o SIG subjacente, correspondendo à linguagem  $L_i$ .

O capítulo 8 discute as camadas subjacentes à interface. Esta última corresponde no capítulo aos chamados SV (Subsistema de Visualização) e SM (Subsistema de Manipulação); o SV trata da apresentação dos dados, enquanto o SM trata das funções para definição e manipulação destes objetos (ou seja, a linguagem  $L_e$ ). A comunicação entre esta camada mais externa e o restante do SIG segue três tipos de filosofia básicos: *unificada*, *difusa* ou *focalizada* [LG93, AYA<sup>+</sup>92]. Modelos unificados são baseados em uma API (Applications Programming Interface) que esconde do projetista de interfaces os detalhes das estruturas físicas dos dados. Modelos difusos aceitam distintas APIs, cada uma das quais baseada em estruturas de dados diferentes. Modelos focalizados eliminam a existência da API, obrigando o projetista de interfaces a manusear diretamente as estruturas de dados, sendo a interface parte integrante do SIG. Neste caso, a linguagem de comunicação  $L_e$  é altamente dependente do SIG.

Os modelos unificado e difuso são considerados *abertos*, ou seja, o usuário entende a interface como um subsistema à parte. Nestes casos, a interface “enxerga” o SIG como um módulo independente com o qual se comunica através de protocolos de comunicação e conversão de dados. O modelo unificado é pouco extensível, pois está dirigido a um conjunto básico de estruturas providas pela API. O modelo difuso é mais flexível, mas apresenta dificuldades de programação, pois cada API precisa ser programada em separado. O modelo focalizado facilita a otimização de estruturas, mas obriga o programador a conhecer detalhes de estruturas físicas.

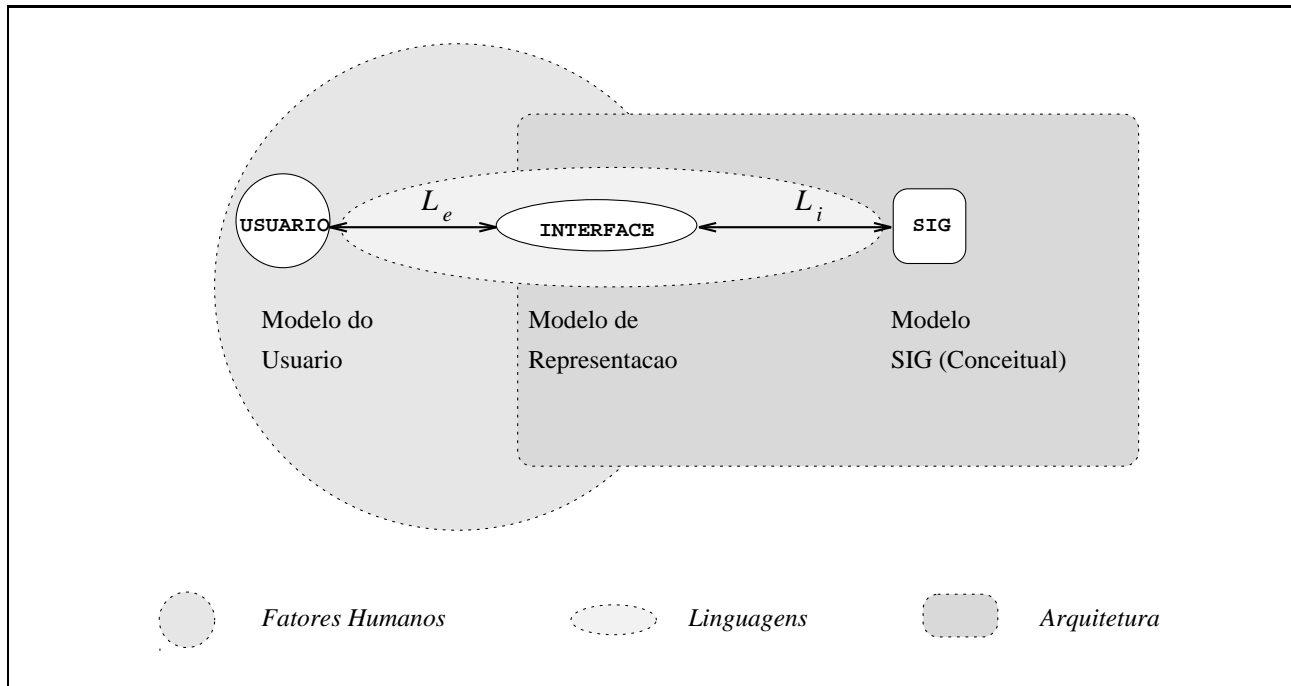


Figura 7.5: Enfoques no projeto de interface

### 7.3.2 Linguagens

Como visto no capítulo 6, linguagens de consulta para SIGs podem ser textuais, visuais ou multimodais. Em geral, não há preocupação com aspectos de visualização no projeto destas linguagens, que apenas especificam quais resultados devem ser apresentados, sem descrever a sintaxe da apresentação. Exceções são [Ege94, Cam95], que especificam, como parte da linguagem, os comandos de apresentação.

Em geral, a linguagem de apresentação é tratada à parte, quer dentro do código de desenvolvimento da aplicação, quer através da criação no banco de dados de entidades específicas que definam os detalhes da visualização. Em ambos os casos, não se trata na verdade de uma linguagem propriamente dita, mas de informações passadas ao SIG através de programas ou menus.

### 7.3.3 Modelo do Usuário

O estudo dos aspectos envolvendo cognição espacial formam a terceira dimensão do projeto de interfaces para SIGs. Estes aspectos dependem do perfil do usuário e necessidades das aplicações, resultando em diferentes *modelos mentais* do usuário [Tve93]. O objetivo da pesquisa nesta área é entender como o ser humano percebe o espaço e, a partir desta percepção, desenvolver um modelo que deve guiar as operações da interface. São igualmente sugeridas formas de prover ao usuário diferentes metáforas de interação, adaptadas a seu modelo cognitivo. As dificuldades começam na modelagem do conhecimento espacial, passam pelo armazenamento de tal conhecimento e vão até a visualização. Esta deve variar conforme as características culturais do usuário e levar em consideração inclusive seu posicionamento em relação aos cenários visualizados (observador externo versus participante).

A construção de um modelo mental deve considerar três fatores [Kuh91, LS92, Kuh93, Gou93, Woo93]: a informação desejada pelo usuário, a informação por ele previamente conhecida e a capacidade do usuário em inferir novos fatos a partir dos resultados apresentados, associando-os ao conhecimento prévio. Modelos mentais são usados como base para definir ordens de execução de comandos e regras de apresentação, validando diferentes tipos de interfaces. Os estudos realizados levam em consideração tanto aspectos cognitivos (qualitativos) quanto físicos (mensuráveis). Exemplos de estudos quantitativos são a cronometragem das ações de um ser humano no manuseio de mapas de papel, ou o rastreamento dos movimentos do olho humano ao examinar um mapa [HCS93]. Tais estudos direcionam a especificação do posicionamento dos elementos apresentados em uma tela.

Vale ressaltar que muitos dos aspectos cognitivos explorados são comuns a trabalhos em outras áreas (vide, por exemplo, [Pet95]). Sob este ponto de vista, uma grande quantidade de estudos se ocupa do uso de cores e de posicionamento relativo dos objetos de uma apresentação. Por exemplo, se duas cores são adjacentes em uma apresentação, a intensidade de uma pode modificar a percepção que o usuário tem da outra (devido ao estímulo da retina), o que o fará associar importâncias relativas aos dados apresentados. Cores “quentes” são mais perceptíveis do que cores “frias”, o que geralmente faz com que apresentações relacionem tons de vermelho e amarelo a valores positivos em mapas [BP94].

Os padrões de percepção de mapas e memorização de seus componentes também ajudam a compor o modelo mental [HCS93]. Certas pessoas classificam dados mentalmente sob forma de hierarquias, enquanto que outras criam grafos mentais que lhes permitem deduzir distâncias e correlações espaciais. A interface, portanto, deve priorizar uma ou outra maneira de apresentação, conforme o usuário. Isto envolve, inclusive, discussões sobre modos de navegação [Nye93], percepção do espaço e formas de apresentação de relacionamentos espaciais em 2D e 3D [Her94].

Ressaltamos finalmente algumas observações relativas ao problema de apresentação. Em primeiro lugar, o uso de interfaces gráficas não garante a clareza ou qualidade dos resultados, pois os usuários “enxergam” o resultado de uma consulta não apenas vendo o que é mostrado, mas também em função de alguma notação gráfica implícita – transmitida, por exemplo, por

noções de adjacência, cor e outras, particulares ao perfil cognitivo do usuário. Em segundo lugar, esta notação gráfica implícita é derivada da experiência do usuário com o domínio da aplicação e seu conhecimento pessoal da região geográfica estudada – ou seja, a familiaridade do usuário com o problema deveria influenciar a estratégia de apresentação. Finalmente, especialistas em um domínio têm necessidades de apresentação distintas daquelas de novatos. Todos estes fatores, no limite, levam à conclusão que um SIG ideal deveria ter uma interface diferente para cada usuário, o que seria bastante complexo. A recomendação mais próxima de tal solução é a criação de um banco de regras que descrevam o perfil do usuário, que seja consultado para direcionar a apresentação [Gou93].

## PARTE III



# Capítulo 8

## Arquiteturas para SIG

### 8.1 Introdução

Este capítulo discute arquiteturas alternativas para SIG, balizando-se pelo paradigma que requer o desenvolvimento de tecnologia para gerenciamento de grandes bancos de dados georeferenciados. A seção 8.2 descreve sucintamente uma arquitetura em camadas para um SIG, seguindo os conceitos introduzidos no capítulo 4 e separando claramente as funções de armazenamento, manipulação e visualização. A seção 8.3 especifica ainda diferentes estratégias de implementação, compatíveis com a arquitetura em camadas apresentada, que permitem classificar, de maneira coerente, os SIG disponíveis comercialmente ou propostos como protótipos. Finalmente, a seção 8.4 estende a discussão para arquiteturas distribuídas para SIG.

Questões genéricas sobre arquitetura de SIG são tratadas em [PMP93, CHF93a, Cam93, Cam94a]. Este capítulo baseia-se principalmente em [Toc95, Cam95], onde o leitor encontrará discussões bastante detalhadas sobre arquiteturas para SIG. Experiências com sistemas de gerência de bancos de dados geográficos são reportadas em [HC91, vOV91, Bat92] para o caso de sistemas relacionais estendidos, em [DRS93b, GR93, MMS93, SV92] para sistemas orientados a objetos e em [SRD<sup>+</sup>91, SA93, LL93, Web90] para o caso do uso de técnicas de inteligência artificial. Exemplos de projetos objetivando a construção de protótipos de SIG incluem Sequoia 2000 [SFGM93], Goose [AA93], Amusing [FT93] e GeO<sub>2</sub> [DRS93a], os dois primeiros ligados ao problema de mudança global.

### 8.2 Uma Arquitetura em Camadas

Uma estratégia comumente empregada para descrever sistemas computacionais consiste em dividi-los em camadas funcionalmente cada vez mais complexas e, ortogonalmente, em subsistemas especializados em determinada tarefa ou em determinado tipo de dados.

Esta seção segue esta estratégia descrevendo sucintamente uma arquitetura em camadas para um SIG que claramente separa as funções de armazenamento, manipulação e visualização.



A arquitetura reflete as classes de objetos e os níveis de abstração introduzidos no capítulo 4. A figura 8.1 indica as camadas e subsistemas propostos para organizar um SIG.

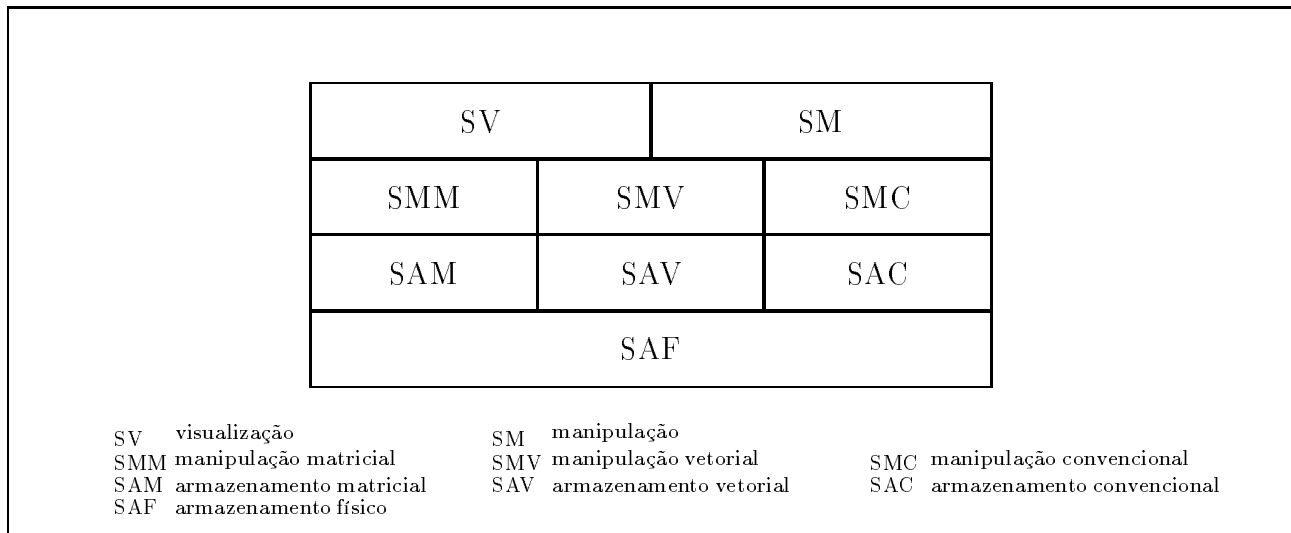


Figura 8.1: Arquitetura genérica em camadas para um SIG.

No nível superior, não especificado pela arquitetura, encontram-se as aplicações de geoprocessamento e módulos de interface com o usuário, discutidos no capítulo 7.

A primeira camada distingue as questões de visualização dos problemas de manipulação e corresponde ao nível conceitual do modelo. O *Subsistema de Visualização* (SV) oferece funções básicas para visualização de objetos tradicionais e georeferenciados, enquanto que o *Subsistema de Manipulação* (SM) oferece funções para definição e manipulação destes objetos.

A segunda camada corresponde ao nível de representação do modelo. Ela oferece separadamente serviços de manipulação de alto nível para atributos convencionais, representações matriciais e representações vetoriais, através do *Subsistema de Manipulação Convencional* (SMC) *Subsistema de Manipulação Matricial* (SMM) e do *Subsistema de Manipulação Vetorial* (SMV), respectivamente.

A terceira camada engloba os subsistemas que oferecem serviços de armazenamento e manipulação elementar para atributos convencionais, representações matriciais e representações vetoriais. Esta camada é composta do *Subsistema de Armazenamento Convencional* (SAC) *Subsistema de Armazenamento Matricial* (SAM) e do *Subsistema de Armazenamento Vetorial* (SAV). Estes subsistemas fazem uso do serviço de armazenamento de páginas físicas disponibilizado pelo *Subsistema de Armazenamento Físico* (SAF), que compõe a quarta camada.

## 8.3 Estratégias de Implementação

Esta seção apresenta diferentes estratégias de implementação para a arquitetura em camadas, baseadas em sistemas de gerência de bancos de dados com grau crescente de funcionalidade. As estratégias de implementação permitem classificar, de maneira coerente, vários SIGs disponíveis comercialmente ou propostos como protótipos.

A tabela 8.1 apresenta uma comparação das vantagens e desvantagens de cada uma das estratégias de implementação discutidas nesta seção. Vale notar que a comparação apresentada é esquemática e esconde nuances. Por exemplo, as vantagens completas da última estratégia discutida só ocorrem quando o SGBD utilizado é completamente extensível.

<i>Estratégia</i>	<i>Vantagens</i>	<i>Problemas</i>
Dual	SGBD de mercado Independência do SGBD	Controle de integridade Otimização de consultas
Campos Longos	Controle de integridade	Dependência do SGBD Otimização de consultas
Extensível	Controle de integridade Otimização de consultas e métodos de indexação	Dependência do SGBD Falta de padronização aceita para ling. de consulta

Tabela 8.1: Comparação entre as estratégias de implementação

### Estratégia Dual

Em linhas gerais, um SIG implementado segundo a estratégia dual utiliza um SGBD relacional para armazenar em tabelas a componente convencional de todos os objetos, e arquivos normais para a componente espacial dos objetos.

Em termos da arquitetura em camadas proposta, esta estratégia significa implementar o SMC e o SAC inteiramente sobre um SGBD relacional e implementar o SMM, SAM, SMV e SAV sobre o sistema de arquivos do sistema operacional, para a componente espacial das representações, e sobre o SGBD, para a componente convencional das representações. Contudo, a maior parte das soluções disponíveis comercialmente para Geoprocessamento que seguem esta estratégia leva a dualidade até as camadas superiores. Exemplos de sistemas comerciais seguindo esta estratégia são o ARC/INFO [Mor92], o MGE [Int90] e o SPRING (vide capítulo 11), onde apenas este último de fato mascara completamente a dualidade para o usuário. De certa forma, o sistema GRASS [U.S92], de domínio público, também segue esta estratégia, embora não utilize um SGBD relacional.

A principal vantagem da estratégia dual está em utilizar SGBDs relacionais de mercado. Por outro lado, como as componentes espaciais estão fora do controle do SGBD, a estratégia dificulta o equacionamento das questões de otimização de consultas, gerência de transações e controle de integridade e de concorrência. Estes problemas só podem ser resolvidos através de

implementações sofisticadas do SMM, SMV e do próprio SM para que operem coordenadamente com o SGBD convencional. A estratégia dual também exige a construção do SAM e do SAV a partir apenas das facilidades oferecidas pelo sistemas de arquivos do sistema operacional, que em geral são muito mais pobres do que aquelas oferecidas por um SGBD.

A execução de uma consulta em um ambiente dual tipicamente segue um plano bem rígido: a componente convencional da consulta é processada pelo SGBD relacional, em separado das restrições espaciais, que são deixadas a cargo do SMM e do SMV; os resultados parciais são posteriormente combinados pelo SM para gerar o resultado final da consulta. Esta rigidez na geração do plano de execução, embora conveniente para a implementação do SIG, geralmente leva a planos sub-ótimos.

Como um segundo exemplo, é possível que uma representação permaneça registrada no banco de dados convencional, embora o arquivo onde estava armazenada a sua componente espacial tenha sido removido, quer inadvertidamente, quer por interrupção indevida do processamento. Para resolver completamente este problema de controle da integridade, é necessário implementar as camadas inferiores da arquitetura genérica de forma inacessível ao sistema operacional (por exemplo, em UNIX, os dados seriam colocados numa “raw partition” – partição da qual o UNIX não conhece a estrutura interna), deixando para o SAF a tarefa de guardar todas informações sobre blocos e trilhas de disco onde estão armazenadas as representações.

### Estratégia baseada em Campos Longos

Esta estratégia baseia-se no uso de SGBDS relacionais com suporte para campos longos, nos quais são armazenadas as componentes espaciais dos objetos. Em outras palavras, em termos da arquitetura em camadas, tanto os subsistemas para manipulação e armazenamento dos atributos convencionais - SMC e SAC - quanto os subsistemas para armazenamento matricial e vetorial - SAM e SAV - passam a ser implementados através do SGBD relacional, pelo menos em parte. Porém, os subsistemas para manipulação matricial e vetorial - SMM e SMV - continuam fora do SGBD. O SYSTEM 9 [vEU89] é um exemplo de um sistema comercial implementado segundo esta estratégia.

Em mais detalhe, do ponto de vista conceitual, um *campo longo* é apenas uma cadeia binária, onde se pode armazenar qualquer tipo de dado, como dados gráficos, numéricos ou pictóricos. Fisicamente, o SGBD permite que um campo longo possua tamanho considerável (em alguns casos, da ordem de gigabytes). As funções oferecidas por um SGBD para tratamento de campos longos são tipicamente limitadas ao acesso a fragmentos do conteúdo do campo, especificando a posição de início e o tamanho do fragmento. Em alguns casos, engloba também mecanismos de integridade física (tipicamente *shadowing*).

A principal vantagem desta estratégia está em armazenar todos os objetos de um banco de dados geográficos através de um único SGBD, evitando os problemas de gerência de transações e de controle de integridade e de concorrência, mais complicados de resolver na estratégia dual. Porém, como o SGBD trata um campo longo como uma cadeia binária, não conhece a semântica do seu conteúdo e, portanto, não possui mecanismos satisfatórios para seu

tratamento. Assim, a implementação do SIG ainda deve incluir métodos de acesso espacial e um otimizador de consultas que prepare planos de execução que façam uso destes métodos. Além disto, embora o padrão da linguagem SQL, principal interface oferecida pelos SGBD relacionais, defina funções básicas para tratamento de campos longos, há pequenas variações nas facilidades de fato disponibilizadas. Como consequência, uma implementação seguindo esta estratégia pode ficar dependente do SGBD escolhido.

### **Estratégia baseada em Mecanismos de Extensão**

Esta estratégia depende da adoção de um SGBD extensível que disponha de mecanismos que permitam implementar o tratamento das componentes espaciais através de extensões ao seu ambiente. Exemplos de SGBD extensíveis são os assim chamados sistemas pós-relacionais, como o POSTGRES [Uni93] e o STARBURST [HC91], e os sistemas orientados a objetos, como o O<sub>2</sub> [BDK92] e o ObjectStore.

No nível da linguagem de definição e manipulação de dados, ou seja da sua interface com o usuário, um SGBD extensível tipicamente permite a definição de novos tipos de objetos (dentre os quais, aqueles introduzidos no modelo apresentados no capítulo 4). Note que a especificação de um novo tipo inclui a definição dos atributos dos objetos e dos métodos que atuam sobre eles. Portanto, esta facilidade vai muito além de campos longos, pois permite capturar a semântica dos objetos.

Para facilitar a implementação de um SIG, um SGBD deve estar aberto também a extensões nos níveis mais internos. Ele deve permitir a definição de novos métodos de armazenamento e acesso, a adição de novas regras ao otimizador de consultas para fazer bom uso dos novos métodos e até a modificação da política de gerência de transações para adaptar a sua funcionalidade às necessidades das aplicações de Geoprocessamento. Em última instância, deve ser possível implementar o SMM, SAM, SMV e o SAV como extensões do SGBD. Naturalmente, esta estratégia apresenta dificuldades técnicas, embora menores do que os casos anteriores, mas aumenta muito a funcionalidade e possivelmente o desempenho do SIG resultante.

Na prática, os SIGs comercialmente disponíveis baseados em estratégias extensíveis (dos quais o mais importante é o SMALLWORLD GIS) desenvolveram seus próprios SGBD, por não disporem de alternativas aceitáveis no mercado à época [Bat92]. Quatro projetos de pesquisa merecem menção no contexto desta estratégia: os sistemas GEO<sup>++</sup>, Sequoia 2000, Geo<sub>2</sub> e Paradise. Os sistemas GEO<sup>++</sup> [vOV91, VvO91] e

Sequoia 2000 [Sto93] foram desenvolvidos sobre o sistema POSTGRES. Este sistema é uma extensão ampla da tecnologia relacional e dispõe de algumas ferramentas para manuseio de dados gráficos, incluindo suporte para árvores-R, estrutura de indexação espacial com bom desempenho e tipos de dados espaciais - ponto, segmento de linha, linhas poligonais e retângulo envolvente ("box"). O sistema GEO<sup>++</sup> foi escrito em C<sup>++</sup> e estende os tipos de dados geométricos do POSTGRES, além de rotinas de acesso aos dados. Sequoia 2000 é um projeto de longo prazo com o propósito de dar suporte a estudos de mudanças globais e envolve aspectos de armazenamento, redes de comunicação de dados de alto desempenho e visualização de dados

georeferenciados [Gar93].

O GeO<sub>2</sub> baseia-se no sistema O<sub>2</sub>, um SGBD orientado a objetos disponível comercialmente [DRS93b]. O GeO<sub>2</sub> utiliza os mecanismos de criação de novos tipos do O<sub>2</sub> para implementar um modelo de dados adequado às aplicações de Geoprocessamento, limitado porém às representações vetoriais. Foi desenvolvido pelo IGN (“Institut Geographique National”) da França para dar apoio à Cartografia Digital francesa, em várias escalas e para vários temas.

O projeto Paradise (*Parallel Data Information System*) da Universidade de Wisconsin [DDY93, ea94] tem como objetivo aplicar as tecnologias de orientação a objetos e bancos de dados paralelos ao desenvolvimento de um SIG capaz de gerenciar grandes bases de dados geográficas. Como gerenciador persistente de objetos, Paradise utiliza o sistema Shore [M. 94], também desenvolvido pelo grupo da Universidade de Wisconsin.

## 8.4 Arquiteturas Distribuídas

### 8.4.1 Arquiteturas Cliente-Multiservidor

Na arquitetura para sistemas de gerência de bancos de dados distribuídos (SGBDD, abreviadamente), o armazenamento dos dados nos vários processadores é guiado por um *critério de distribuição*, que pode ou não incluir replicação de fragmentos dos dados. O ciclo de execução de uma consulta ou atualização passa a incluir, então, execução de operações locais, transferências de dados e composição de resultados parciais.

Já as arquiteturas cliente-servidor mais recentes para bancos de dados preocupam-se em separar a funcionalidade do SGBD entre um *servidor* e um ou mais *clientes*. O servidor oferece serviços de armazenamento e recuperação, que podem ser no nível de arquivos (como o NFS do sistema UNIX), de páginas físicas ou de objetos (no caso de alguns sistemas orientados a objetos). Cada cliente implementa as camadas superiores do sistema até o nível da aplicação. Esta discussão pode ser unificada com a questão de SGBDD considerando-se um conjunto de servidores, em lugar de um único, gerando uma arquitetura *cliente-multiservidor*.

Reportando-se à arquitetura em camadas descrita na seção 8.2, obtêm-se diferentes arquiteturas cliente-multiservidor variando-se a camada em que se introduz o critério de distribuição dos dados e o corte entre cliente e servidor [Toc95]. As arquiteturas resultantes distinguem-se entre si pelo grau de abstração dos dados fragmentados e pelas operações realizadas no cliente e nos servidores. A camada onde é aplicado o corte passa a ser responsável por mascarar para a camada superior a distribuição do armazenamento e manipulação dos dados, controlando a localização dos fragmentos dos dados, bem como o processamento distribuído das operações oferecidas naquela camada.

Considere inicialmente que a distribuição se dá na primeira camada, ou seja, no nível do SV e SM (ver figura 8.1). Neste caso, pode-se definir diversos critérios de distribuição para objetos tradicionais e georeferenciados de forma semelhante aos usados em bancos de dados convencionais. O corte da arquitetura nesta camada resulta em um banco de dados distribuído

fracamente acoplado, onde cada servidor conhece a semântica dos dados e implementa operações de alto nível, tornando os servidores independentes entre si.

A distribuição no nível da segunda camada facilita a definição de critérios de distribuição para a componente espacial de forma completamente independente dos critérios para a componente convencional dos objetos. Contudo, cada um dos servidores ainda conhece a semântica das componentes e pode operar sobre elas de forma independente. Uma forma de distribuição baseada na fragmentação das componentes espaciais oferece maior flexibilidade para paralelizar operações espaciais, já que permite armazenar fragmentos de componentes espaciais em servidores distintos.

Os dois últimos pontos de corte possíveis são no nível das camadas de armazenamento. Cortando no penúltimo nível, apenas operações básicas sobre representações vetoriais (no SAV), sobre representações matriciais (no SAM) ou sobre atributos convencionais (no SAC) são paralelizadas. Os servidores que armazenam os fragmentos destes objetos são completamente dependentes dos processos que implementam as camadas superiores. Já o corte no nível do SAF gera uma arquitetura cliente-multiservidor com distribuição em nível de páginas físicas, onde os servidores desconhecem completamente a semântica dos objetos.

### 8.4.2 Arquiteturas Orientadas-a-Serviços

Em uma arquitetura cliente-servidor tradicional, conforme discutido na seção anterior, cada cliente recebe comandos do usuário e executa aplicações, que requisitam dados a um ou mais servidores. Já em uma arquitetura orientada-a-serviços, a hierarquia cliente-servidor tradicional é quebrada em três componentes: clientes, servidores e uma *malha de interoperabilidade* (*request broker*), conforme ilustrado na figura 8.2. Os sistemas EOSDIS [EMM94] e METVIEW [RND<sup>+</sup>95] são exemplos desta arquitetura no caso de SIG.

O coração desta arquitetura é a malha de interoperabilidade, responsável pela comunicação entre os clientes e os servidores. Esta malha realiza vários tipos de operações: anúncio dos serviços disponíveis para os clientes; adequação do pedido dos usuários aos serviços disponíveis. Em muitos casos, uma solicitação do usuário pode envolver vários serviços disponíveis no sistema; tradução e transferência de parâmetros e dados entre máquinas, processos e espaços de endereçamento distintos.

Além disto, em uma arquitetura orientada-a-serviços, os servidores são tipicamente especializados para tarefas bem definidas. Isto permite que cada servidor tenha apenas a informação necessária para sua operação, sem sobrecarga desnecessária.

Os princípios genéricos enumerados acima para arquiteturas orientadas-a-serviços podem ser aplicados à arquitetura em camadas descrita na seção 8.2, gerando diferentes arquiteturas orientadas-a-serviços para SIG. Contudo, os critérios para divisão dos serviços entre os clientes e servidores serão aqui considerados mais rígidos, seguindo as limitações listadas abaixo:

- os pontos de corte coincidem sempre com as interfaces entre as camadas, facilitando a definição da malha de interoperabilidade;

- o critério de distribuição do banco de dados entre os servidores será apenas por tipo de objeto, não se considerando a possibilidade de fragmentar objetos.

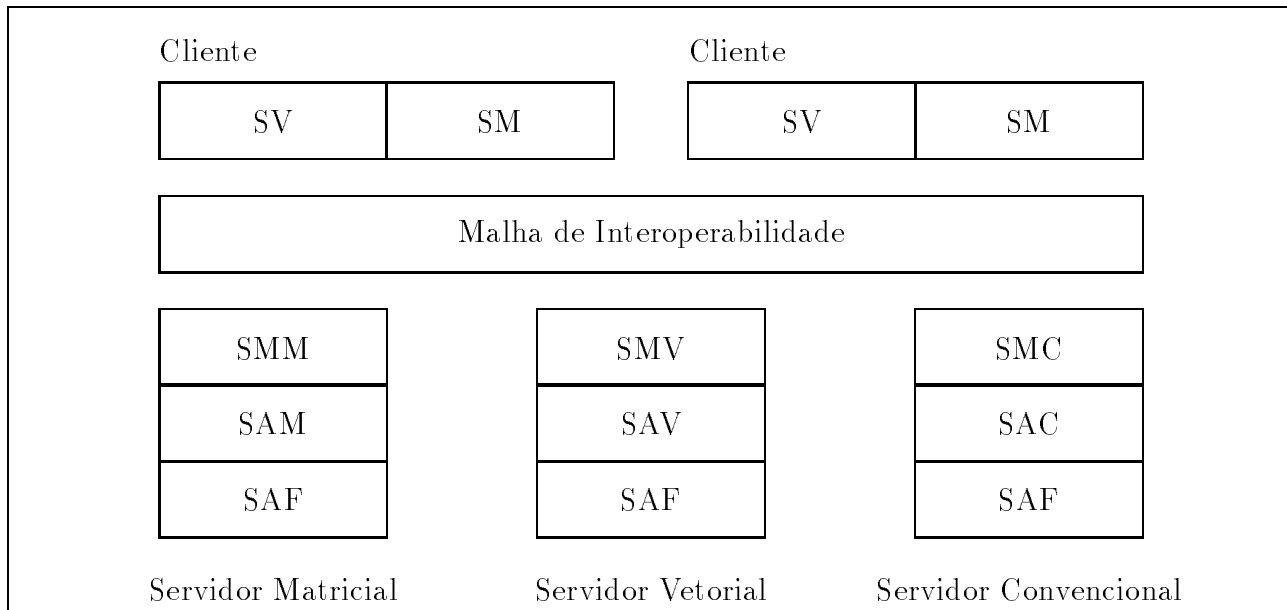


Figura 8.2: Arquitetura orientada-a-serviços para SIG.

O resto desta seção discute alternativas dentro destas limitações.

A primeira alternativa consiste em implementar os servidores de armazenamento de forma separada (e por conseguinte dividir o SAF em três). Cada servidor passa a ser especializado portanto no armazenamento de um tipo de objeto - matricial, vetorial ou convencional. A implementação das camadas superiores fica a cargo dos clientes. Esta alternativa é bastante conveniente quando o volume de dados é substancial. No caso extremo, cada servidor pode ser substituído por um conjunto de servidores, formando um *cluster*, ou ligados por uma rede local ou mesmo por uma rede de longo alcance.

Os servidores especializados não conhecem a organização lógica do banco de dados geográfico e são responsáveis apenas por serviços elementares de armazenamento persistente e recuperação e pelos serviços locais de gerência de transações, controle de concorrência e controle de integridade. A malha de interoperabilidade trabalha neste caso em um nível de serviço bastante elementar. O serviço de otimização de consultas, e os serviços globais de gerência de transações, controle de concorrência e controle de integridade permanecem sob a responsabilidade das camadas superiores, implementadas (replicadamente) em cada cliente. Em particular, quando uma consulta é submetida a partir de um cliente, a implementação local do SM deverá estabelecer um plano ótimo de execução e coordenar a sua execução. Cada plano consistirá na execução de subconsultas sobre representações matriciais pelo SMM, sobre representações vetoriais pelo SMV e sobre atributos convencionais pelo SMC, que eventualmente chamam os servidores de armazenamento apropriados através da malha de interoperabilidade.

Evoluindo para uma camada superior, a segunda alternativa seria implementar em servidores separados os pares de subsistemas de manipulação/armazenamento para cada tipo de objeto. O subsistema responsável pela visualização de representações, em princípio, permaneceria sob responsabilidade do cliente, embora do ponto de vista lógico nada impeça a criação de um servidor especializado em visualizações. Os servidores especializados neste caso oferecem serviços sofisticados, incluindo recuperação de dados através de consultas de alto nível. As subconsultas geradas pelo plano de execução de uma consulta seriam então diretamente submetidas ao servidores especializados.

Em resumo, as arquiteturas orientadas-a-serviços para SIG discutidas nesta seção baseiam-se em servidores especializados por tipo de objeto, sugerindo implementações próximas do esquema tradicional de sistemas de gerência de bancos de dados distribuídos heterogêneos com acoplamento fraco entre os nós. As alternativas apresentadas diferem essencialmente pelo nível dos serviços oferecidos pelos servidores. Este enfoque justifica-se pelas enormes diferenças de comportamento e volume de dados entre os diversos tipos de representações, geo-objetos e geo-campos manipulados pelas aplicações de Geoprocessamento.

Por fim, observamos que as decisões mais importantes sobre a implementação do processo que gerência a malha de interoperabilidade são o modelo de objetos a ser utilizado e o protocolo de comunicação com os demais processos. Alternativas possíveis incluem [Cam95]:

- utilização de um padrão de mercado, onde as opções incluem SOM [Cam94b] e OLE [Bro94];
- implementação de uma “casca” adicional;
- adoção de protocolo dedicado, de rápida implementação, semelhante ao disponível para o sistema METVIEW [RND<sup>+</sup>95].

Um dos problemas com os padrões emergentes na área é a incerteza relativa quanto ao seu futuro, o que aumenta em muito o risco de qualquer opção. Adicionalmente, embora especificações como CORBA [Obj94] definam um padrão de interoperabilidade em sistemas distribuídos, algumas de suas implementações (como o SOM) operam em um único espaço de endereçamento [Cam94b].

O aspecto talvez mais importante é o vício de origem. Padrões como SOM e OLE têm como objetivo maior o suporte a documentos compostos, típicos de aplicações de automação de escritório, necessitando adaptações para o manuseio de dados geográficos. O padrão OGIS, discutido na seção 4.3, propõe-se exatamente a adaptar um padrão de interoperabilidade tipo CORBA para aplicações geográficas.





# Capítulo 9

## Armazenamento de Dados Geográficos

### 9.1 Introdução

Este capítulo discute as questões de armazenamento e indexação de dados geográficos, a partir da arquitetura e conceitos apresentados no capítulo 8.

Há uma considerável quantidade de trabalhos publicados sobre bancos de dados geográficos abordando métodos de indexação espacial [Sam89, GB90, HS92b, Cox91], [Kri91, HS92b, Rot91, SFGM93, KBS93, MCD94], sem que haja um consenso sobre qual é o melhor [GB90]. Um estudo comparativo sobre o desempenho de diferentes tipos de árvores-R pode ser encontrado, por exemplo, em [Cox91].

### 9.2 Estratégia Elementar de Armazenamento

De acordo com a discussão da seção 8.3, as implementações mais simples da arquitetura em camadas seguem a estratégia dual ou a estratégia baseada em campos longos. Resumidamente, estas estratégias:

- separam a componente convencional da componente espacial dos objetos;
- armazenam a componente convencional através de um SGBD;
- armazenam a componente espacial em arquivos convencionais, ou em campos longos, se o SGBD oferecer esta facilidade.

As virtudes e problemas destas estratégias já foram comentadas na seção 8.3. Recordando alguns pontos, estas estratégias permitem construir, de forma simples e imediata, um SIG sobre um SGBD usufruindo dos serviços por este oferecidos. Porém, por não levar em conta a natureza da componente espacial dos objetos, elas oferecem poucas oportunidades para otimizar o processamento das operações espaciais. O resto desta seção discute algumas formas elementares de otimização, abstraindo até onde for possível a forma exata de implementação.

A sequência típica de processamento neste contexto consiste em selecionar um conjunto de objetos através de uma consulta convencional, trazer da memória secundária para a memória principal as suas componentes espaciais, ou apenas parte delas, e executar operações espaciais sobre estas componentes.

Há assim três oportunidades para otimização. A primeira oportunidade consiste em facilitar a seleção dos objetos, armazenando no SGBD, sob forma de atributos convencionais, valores pré-computados a partir das suas componentes espaciais. Por exemplo, para um objeto cuja componente espacial seja um polígono, armazena-se a área, a descrição do perímetro, e uma aproximação do polígono, tal como o seu retângulo envolvente mínimo. O usuário, ou automaticamente o sistema, tem neste caso a oportunidade de selecionar objetos pelas características do seu polígono, utilizando estes novos atributos, juntamente com os convencionais. Para um objeto cuja componente espacial seja uma imagem, computa-se vários descritores, como por exemplo o nível médio de cinza e uma medida de textura da imagem, ou mesmo cria-se um pequeno ícone re-amostrando-se drasticamente a imagem. Há, por exemplo, sistemas que oferecem uma interface para o usuário selecionar um conjunto de imagens através de seus descritores e refinar a escolha folheando os ícones das imagens selecionadas (ver também a seção 6.2.4).

A segunda oportunidade de otimização ocorre no nível de armazenamento físico. De fato, se a componente espacial de um objeto ocupar mais de uma página física, o subsistema de armazenamento deverá armazenar todas estas páginas contiguamente em memória secundária para minimizar o tempo de leitura das páginas. Além disto, uma vez solicitada a recuperação da componente espacial, o subsistema de armazenamento deverá reconstruir o valor do atributo em um bloco contíguo de memória principal e passá-lo para a camada superior, ou implementar um esquema mais sofisticado de paginação que dê a ilusão para as camadas superiores de que o valor da componente reside completamente em memória principal. Note que esta forma de otimização é completamente independente do tipo da componente espacial e da operação que o usuário deseja realizar.

A terceira oportunidade de otimização, por outro lado, depende completamente do tipo da componente espacial e consiste em escolher uma estratégia de armazenamento que otimize as operações tipicamente executadas sobre esta componente. Neste caso, os polígonos deverão ter seus vértices armazenados em uma lista ordenada por valor crescente da coordenada usada na varredura. Esta estratégia torna-se interessante principalmente quando o subsistema armazena esta lista em várias páginas físicas e as transfere uma a uma para memória principal. Assim, se o espaço for varrido apenas uma vez, cada página será transferida também apenas uma vez da memória secundária para a memória principal (exceto se o número de regiões envolvidas for tão grande que não seja possível manter em memória principal pelo menos uma página para cada região).

O restante deste capítulo descreve métodos especiais de armazenamento e indexação que levam em consideração o tipo da componente espacial para acelerar operações típicas sobre ela.

## 9.3 Armazenamento de Representações Matriciais

Recordando, uma representação matricial nada mais é do que uma matriz, possivelmente de grande dimensão. O método de armazenamento para matrizes mais simples consiste em armazenar, nas páginas físicas em memória secundária, uma linha após a outra (ou uma coluna após a outra). Este método é adequado para processar operações que exigem visitar os elementos da matriz linha a linha, ou sem nenhuma ordem preferencial como, por exemplo, no cálculo de um histograma. Ele não é apropriado, no entanto, quando a operação exige visitar elementos em uma ordem que não seja aquela usada no armazenamento, ou visitar um elemento e seus vizinhos nas duas dimensões, ou mesmo recuperar os elementos da matriz cobertos por uma janela definida por um intervalo de índices em cada dimensão.

É possível minimizar este problema através de um método de armazenamento que divida cada linha (ou coluna) em *segmentos* de comprimento fixo, submúltiplo do tamanho da linha, e que permita transferir tais segmentos da memória secundária para a memória principal e vice-versa. Este método seria complementado com um mecanismo que permitisse trazer concomitantemente para memória principal vários segmentos de linha. Note que, no caso de armazenamento linha a linha, operações percorrendo a matriz coluna a coluna ainda seriam processadas ineficientemente.

Um terceiro método de armazenamento para matrizes consiste em dividir cada matriz em pequenos blocos, digamos de dimensão  $256 \times 256$ , armazenados sequencialmente em memória secundária segundo alguma ordem, digamos na direção das linhas. Este método, complementado por mecanismo de *buffer pool*, permite otimizar o processamento de operações que não trabalham linha a linha (ou coluna a coluna).

Em todos estes métodos, dado um elemento ou uma janela de elementos, é possível identificar os segmentos ou blocos envolvidos e computar diretamente as páginas físicas em memória secundária que os contêm. Este mapeamento simples é quebrado se cada linha, segmento ou bloco sofrer compactação (sem perda) antes de ser armazenado. A inclusão de compactação diretamente no método de armazenamento é bastante interessante no caso da representação de imagens de sensoriamento remoto e de mapas temáticos, onde a redundância é bastante alta. Porém, cada linha, segmento ou bloco, depois de compactado, passa a ter comprimento variável, não havendo mais um mapeamento simples entre cada entrada na matriz e sua posição em uma página em memória secundária. Para contornar este novo problema, a estrutura de armazenamento da matriz deve ser acompanhada de um diretório auxiliar indicando em que posição de que página física começa cada linha (segmento ou bloco) compactado e qual o seu comprimento depois de compactado. Este diretório será tipicamente pequeno, podendo residir em memória principal após a matriz ter sido aberta para processamento.

## 9.4 Armazenamento e Indexação de Representações Vetoriais

Nesta seção trataremos sucessivamente do armazenamento e indexação espacial de conjuntos de pontos, retângulos, linhas e regiões definidos em  $\mathcal{R}^2$ . Em seguida, discutiremos o armazenamento de representações vetoriais mais complexas. No que se segue, pressupomos o uso de um sistema de coordenadas planas, o que nos permite mencionar o eixo dos  $X$  e o eixo dos  $Y$ .

### 9.4.1 Pontos

Nesta seção descreveremos dois métodos de armazenamento e indexação espacial a conjuntos de pontos, os arquivos em grade (*grid files* [NHS84]) e as árvores KD (*kd-trees* [BER85]).

Uma *grade* particiona a região de interesse em *células* através de retas paralelas aos eixos e espaçadas irregularmente, conforme ilustrado na figura 9.1. Um *arquivo em grade* consiste de duas *escalas*, uma para cada eixo, e um *diretório*. A escala associada ao eixo  $X$  é uma lista contendo as abcissas em que as retas paralelas ao eixo  $Y$  cruzam o eixo  $X$  (e similarmente para a outra escala). O diretório é uma matriz bidimensional, com as mesmas dimensões da grade, tal que cada elemento do diretório corresponde a uma célula e aponta para a página física (ou *bucket*) que contém todas as coordenadas dos pontos localizados na célula. Vários elementos podem apontar para a mesma página física. As escalas e o próprio diretório, se forem estruturas de dados pequenas, poderão residir em memória principal; caso contrário deverão ser por sua vez paginados.

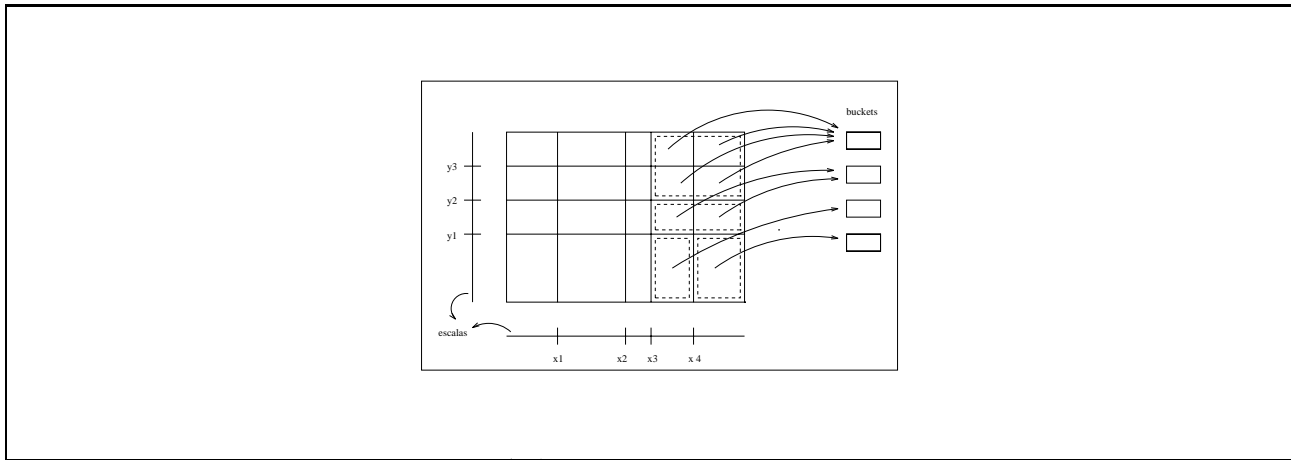


Figura 9.1: Estrutura de um arquivo em grade.

Suponha, por exemplo, que as escalas e o diretório residam em memória principal e que se deseje recuperar todos os pontos contidos em uma dada região. Neste caso, utiliza-se inicialmente as escalas para computar que células cobrem a região; em seguida, através do diretório, obtém-se os endereços das páginas físicas associadas às células; finalmente, lê-se as páginas

físicas contendo pontos e seleciona-se aqueles que de fato estão contidos na região dada. A referência [NHS84] descreve em detalhe as operações sobre arquivos em grade.

Uma *árvore kd* também particiona o espaço em células, mas de uma forma diferente dos arquivos em grade, conforme ilustrado na figura 9.2. A árvore kd é uma árvore de busca binária, residindo em memória principal, tal que os nós interiores em cada nível contêm valores referentes a um único eixo,  $X$  ou  $Y$ , alternadamente, e as folhas apontam para páginas físicas. Várias folhas podem apontar para a mesma página física. Assim, o valor armazenado na raiz divide o espaço em dois subespaços através de uma reta perpendicular ao eixo dos  $X$ , digamos; o valor armazenado no filho à esquerda (ou direita) por sua vez divide o subespaço à esquerda (ou direita) em dois subespaços através de uma reta perpendicular ao eixo dos  $Y$ ; e assim por diante, alternando as dimensões. O processo de subdivisão pára quando a célula contém um conjunto de pontos que possa ser armazenado em uma única página física.

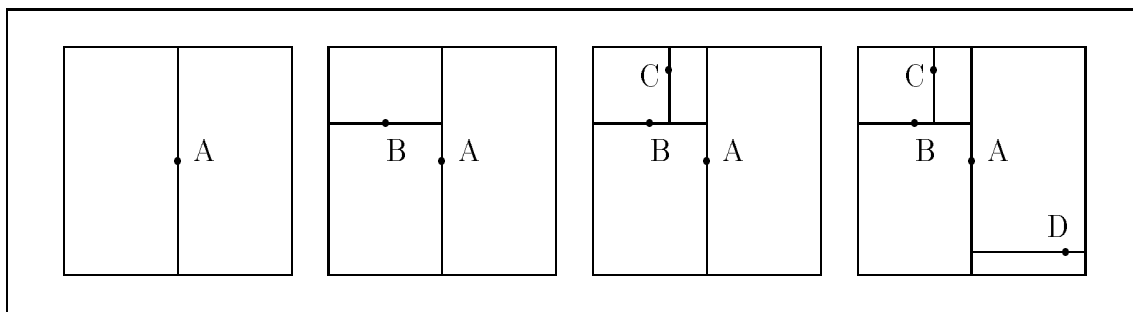


Figura 9.2: Particionamento do espaço determinado por uma árvore kd

Tanto os arquivos em grade quanto as árvores kd podem ser generalizados para espaços de dimensão superior a 2 (de fato o próprio nome desta segunda estrutura sugere um espaço  $k$ -dimensional). Há um número substancial de variantes de árvores kd reportadas na literatura, como as árvores KDB [Rob81], que generalizam as árvores B, as árvores LSD [AHW89] (para *local split decision*), que permitem escolher a cada nível qual dimensão será dividida. Outros métodos de armazenamento e indexação de conjuntos de pontos são, por exemplo, *EXCELL* [Tam82], *buddy hash trees* [SK90], *BANG Files* [Fre87] e as *hB-trees* (*hole brick trees*) [LS89].

### 9.4.2 Retângulos

Nesta seção abordaremos três enfoques para o armazenamento e indexação espacial de conjuntos de retângulos: transformação de retângulos em pontos; decomposição do espaço com superposição; e retalhamento de retângulos. Maior ênfase será dada ao segundo enfoque.

O primeiro enfoque consiste em transformar um retângulo de duas dimensões em uma quádrupla (um ponto de 4 dimensões) e utilizar um método de armazenamento e indexação espacial de quádruplas, como arquivos em grade e árvores kd quadri-dimensionais. De fato, um retângulo está perfeitamente definido por quádruplas indicando a abscissa e ordenada dos dois pontos extremos de uma diagonal, ou a base, altura e o seu ponto central. Este enfoque,

embora elegante e simples, apresenta problemas de desempenho sérios, discutidos em detalhe em [PST93].

O segundo enfoque abandona a idéia de decompor o espaço em regiões que não se superpõem e utiliza a noção de retângulo envolvente. O *retângulo envolvente mínimo* (r.e.m.) de um conjunto de objetos no  $\mathcal{R}^2$  é o menor retângulo com lados paralelos aos eixos  $X$  e  $Y$  que contém todos os objetos no conjunto.

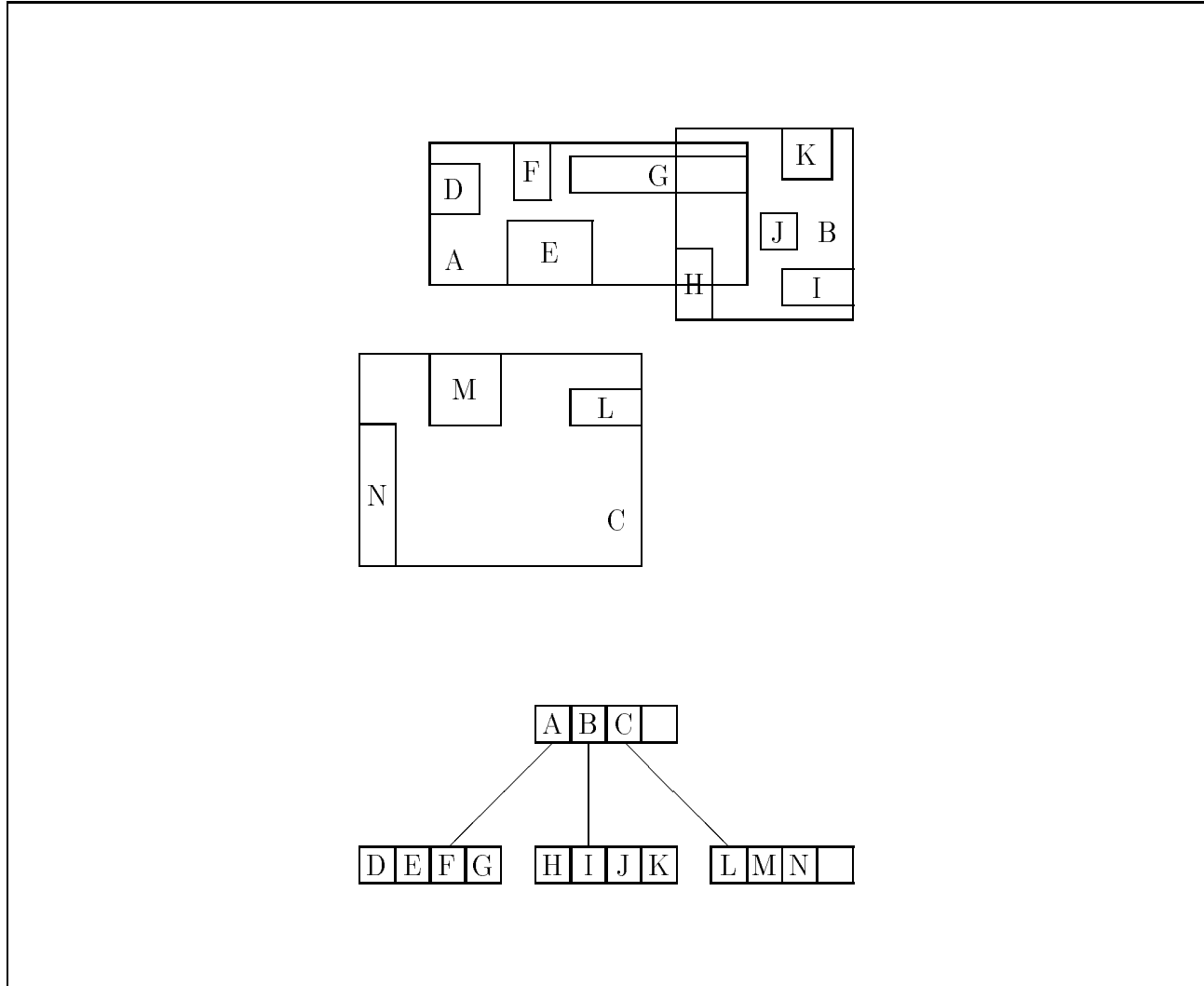


Figura 9.3: Exemplo de uma árvore R

O melhor exemplo deste enfoque são as *árvores R* [Gut84], ilustradas na figura 9.3. Dados dois inteiros  $m > 1$  e  $n > 1$ , uma *árvore R de ordem*  $(m, n)$  é uma árvore de busca balanceada tal que:

- todas as folhas estão no mesmo nível;

- cada folha  $N$  contém entre  $\lceil n/2 \rceil$  e  $n$  retângulos. O r.e.m. de  $N$  é o r.e.m. do conjunto de todos os retângulos armazenados em  $N$ .
- cada nó interior contém entre  $\lceil m/2 \rceil$  e  $m$  filhos, exceto a raiz, que contém entre 2 e  $m$  filhos.
- cada nó interior  $N$  contém uma entrada para cada filho  $M$ , consistindo de um ponteiro para  $M$  e o r.e.m. de  $M$ . O r.e.m. de um nó interior  $N$  é o r.e.m. do conjunto de todos os r.e.m. associados a entradas de  $N$ .

Todos os ramos de uma árvore  $R$  possuem o mesmo comprimento, ou seja, todas as folhas estão no mesmo nível, como numa árvore de busca balanceada. Esta propriedade é mantida através de estratégias especiais de inserção e remoção, semelhantes às de árvores  $B$ . Resumidamente, a inserção de uma nova entrada em um nó  $N$  que já possua o número máximo de entradas provoca a sua divisão, ficando  $N$  com metade (ou metade mais uma) das entradas, e criando-se um novo nó  $N'$  com a outra metade; uma entrada referente ao novo nó  $N'$  é inserida recursivamente no pai de  $N$ . A remoção segue o processo inverso, podendo dois nós serem combinados, recursivamente.

O processo de busca é guiado pelos retângulos armazenados nos nós. Por exemplo, voltando à figura 9.3, para se determinar quais retângulos contêm o ponto  $P$ , procede-se da seguinte forma. Observa-se inicialmente que os retângulos  $A$  e  $B$ , armazenados na raiz, contêm  $P$ ; estes retângulos correspondem ao primeiro e ao segundo filho da raiz. Em seguida, repete-se o processo descendo-se por estes nós. Assim, descendo-se pelo primeiro filho da raiz, observa-se que o retângulo  $G$  contém  $P$  e acrescenta-se este retângulo à resposta. Descendo-se pelo segundo filho da raiz, observa-se que nenhum retângulo aí armazenado contém  $P$ .

Cada retângulo do conjunto original está armazenado em apenas uma folha da árvore  $R$ . Porém, para cada nível da árvore, os retângulos contidos nos nós daquele nível podem se superpor. Na figura, por exemplo, os retângulos  $A$  e  $B$  contidos na raiz se superpõem. Isto significa que uma busca pode ser obrigada a descer por mais de um ramo da árvore. De fato, os retângulos armazenados em uma árvore  $R$  não induzem uma partição do espaço pois é possível ocorrer tanto superposição de retângulos quanto haver regiões do espaço não cobertas por nenhum retângulo. Além disto, a decomposição do espaço é dinâmica, podendo ser modificada a cada inserção ou remoção de um retângulo. Assim, para otimizar o processo de pesquisa, torna-se importante minimizar a superposição dos retângulos armazenados nos nós interiores. Das várias estratégias encontradas na literatura, a proposta em [BKSS90], gerando as chamadas árvores  $R^*$ , aparece como a mais eficiente.

Por fim, o terceiro enfoque mantém a idéia de decompor o espaço em regiões que não se superpõem. Porém, cada retângulo ao ser inserido é retalhado em um conjunto mínimo de retângulos tal que: os retângulos não se superpõem; a união dos retângulos reconstrói o retângulo original; cada retângulo intercepta exatamente uma das regiões em que se decompõe o espaço.

Este enfoque permite adaptar, usando retângulos, qualquer método desenvolvido para arma-



zenar e acessar pontos. Ele permite ainda utilizar uma só estrutura de dados para armazenar, ao mesmo tempo, pontos e retângulos. Por outro lado, resulta na replicação de objetos na estrutura, pois a inserção de um único retângulo acarreta a inserção de uma entrada para cada região que intercepta. Isto complica os processos de inserção e remoção, além de implicar em mais espaço de armazenamento.

As árvores  $R^+$  [SRF87] são uma variante de árvores R em que, ao se inserir uma nova entrada, se o seu retângulo intercepta os retângulos de dois ou mais nós, o retângulo é recortado como anteriormente descrito e cada um dos novos retângulos é inserido nos nós apropriados, como ilustrado na figura 9.4.

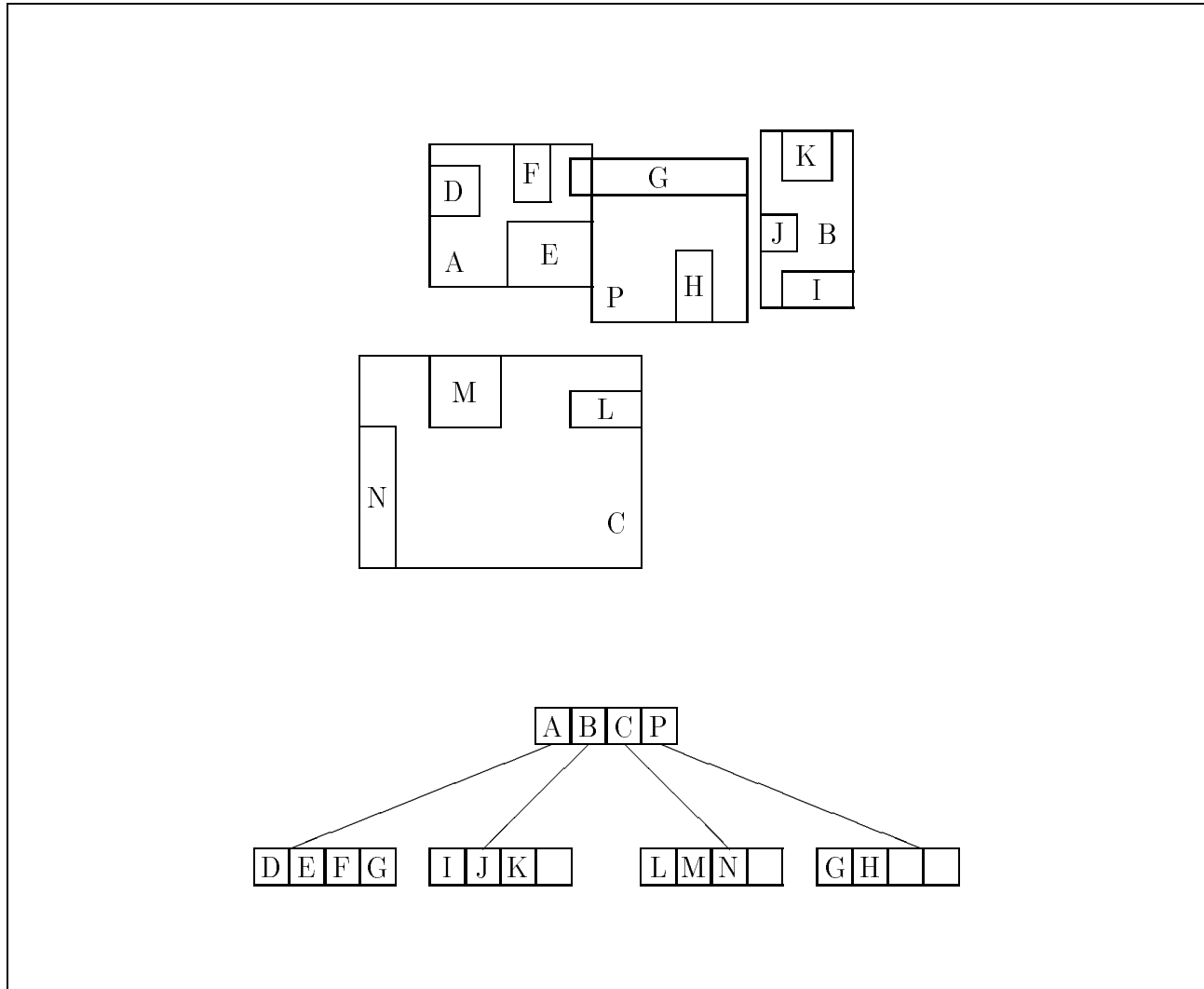


Figura 9.4: Exemplo de uma árvore  $R^+$

A inserção e a remoção de entradas é um processo bem mais complexo nas árvores  $R^+$  do que nas árvores R. Os retângulos inseridos tendem a se dividir para evitar superposição, aumentando o tamanho da árvore. De fato, ao contrário das árvores R, a inserção de uma nova entrada nesta

estrutura pode desencadear recursivamente uma multiplicidade de inserções. Comparações de desempenho entre árvores  $R^+$  e árvores  $R$  podem ser encontradas em [CFR87].

### 9.4.3 Linhas Poligonais

O armazenamento e indexação espacial de conjuntos de linhas pode ser abordado adaptando-se os métodos utilizados para pontos e retângulos.

Dentro da estratégia de decompor o espaço em regiões que não se superpõem, cada linha é quebrada em um conjunto mínimo de novas linhas tal que: as novas linhas não se superpõem; a união das novas linhas resulta na linha original; cada nova linha intercepta exatamente uma das regiões em que se decompõe o espaço. Este enfoque permite adaptar para linhas qualquer método desenvolvido para armazenar e acessar pontos. Pode gerar, porém, para cada linha, múltiplas entradas na estrutura de dados adotada, com as mesmas conseqüências para os processos de inserção e remoção, e desperdício de espaço de armazenamento, já observados para retângulos.

Dado um conjunto de linhas, um segundo enfoque consiste em aproximar cada linha pelo seu retângulo envolvente mínimo e utilizar um dos métodos discutidos anteriormente para armazenar e acessar os retângulos assim gerados. Por exemplo, adotando-se árvores  $R$ , cada folha passa a conter como entradas um apontador para uma linha e seu retângulo envolvente mínimo. Este enfoque não é apropriado, porém, para o caso de linhas muito longas ou, em geral, quando o retângulo envolvente mínimo não resulta em uma boa aproximação para as linhas. Este último problema é resolvido utilizando-se as *árvores  $V$  e  $VR$* , descritas no resto desta seção.

Dados dois inteiros  $m > 1$  e  $n > 1$ , uma *árvore  $V$  de ordem  $(m, n)$*  para armazenar uma linha  $L$  é construída da seguinte forma. A linha  $L$  é quebrada em fragmentos consecutivos, de tamanho entre  $n/2$  e  $n$ , que são aproximados pelo seu retângulo envolvente mínimo. (Esta flexibilidade no número de pontos por fragmento permite remover e adicionar pontos posteriormente, mantendo as características da estrutura resultante). Durante o processo de quebrar a linha em fragmentos, o último ponto de cada fragmento torna-se o primeiro do próximo fragmento de tal forma que os retângulos envolventes cobrem a linha completamente. Uma *árvore  $R$  de ordem  $(m, n)$*  é então construída a partir dos retângulos envolventes mínimos dos fragmentos e considerando que suas folhas armazenam os próprios fragmentos. A estrutura de dados resultante será então a *árvore  $V$  de ordem  $(m, n)$*  para  $L$ .

Um conjunto de linhas é armazenado da seguinte forma. Cada linha é armazenada em uma *árvore  $V$*  separada e aproximada pelo seu retângulo envolvente mínimo; o conjunto destes retângulos é então armazenado em uma *árvore  $R$* , cujas folhas apontam para as raízes das *árvores  $V$* . A estrutura resultante, composta desta *árvore  $R$*  e das várias *árvores  $V$* , forma então uma *árvore  $VR$* .

Ao fragmentar as linhas, este método resolve o problema de armazenar conjuntos de linhas muito longas, facilitando o acesso a partes das linhas, bem como mudanças para escalas menores

(por exemplo, de 1:250,000 para 1:1,000,000). Uma análise mais detalhada do método pode ser encontrada em [MCD94].

#### 9.4.4 Polígonos

O armazenamento e indexação de polígonos pode ser tratado, de forma simplista, como um caso particular do tratamento de linhas. Esta última parte do capítulo aprofunda a discussão abordando métodos de armazenamento e indexação de polígonos que se baseiam no uso de aproximações para a geometria dos polígonos e na decomposição dos polígonos em polígonos simples. O uso de aproximações evita testes geométricos complexos, enquanto que as técnicas de decomposição simplificam e aceleram tais testes.

Para que uma aproximação da geometria de um polígono seja útil ao processamento de consultas, ela deve ser conservativa, ou seja, deve conter completamente o polígono. O retângulo envolvente mínimo (r.e.m.) de um polígono, por exemplo, oferece uma aproximação bastante simples e cômoda. De fato, dado um conjunto de polígonos, podemos criar os r.e.m.s dos polígonos e utilizar quaisquer dos métodos de armazenamento e indexação de retângulos descritos anteriormente para tratar o conjunto de polígonos. Assim, qualquer consulta espacial  $Q$  ao conjunto de polígonos é inicialmente transformada em uma consulta espacial  $Q'$  ao conjunto de r.e.m.s de tal forma que o resultado de  $Q'$  identifique todos os polígonos que possivelmente satisfazem  $Q$ ; estes polígonos devem então sofrer um pós-processamento para separar aqueles que realmente satisfazem  $Q$ .

Decompor um polígono significa dividi-lo em várias componentes simples - triângulos, trapézios, e outros - de tal forma que, durante o processamento de uma consulta espacial, apenas um pequeno número destas componentes precise ser examinado. Assim, cada polígono sofre inicialmente uma decomposição em elementos geométricos mais simples, que são então armazenados através de um método apropriado. Note que as árvores  $V$  discutidas na seção anterior de fato aplicam às linhas esta idéia de decomposição.

Para completar esta seção, descreveremos o *método de decomposição por grades* que combina diretamente as idéias de aproximação e decomposição. Considere o espaço dividido hierarquicamente em células regulares. As células geradas a cada nível podem ser colocadas em *z-ordem* [Ore86] através do processo de *intercalação de bits*, proposto originalmente por Morton [Mor66]. Na figura 9.5, o diagrama à esquerda indica a *z-ordem* imposta aos 4 quadrantes no primeiro nível da divisão hierárquica do espaço. O diagrama à direita ilustra o nível seguinte: em cada quadrante, as células são conectadas em *z-ordem* e depois os grupos de células de cada um dos quatro quadrantes são novamente conectados em *z-ordem*. Cada célula em cada nível da hierarquia possui um endereço dado por uma cadeia de bits cujo comprimento corresponde ao nível ao qual a célula pertence. Por exemplo, o endereço '1110' da célula hachurada à direita da figura foi obtido intercalando-se os bits da sua abcissa '11' com os bits da sua ordenada '10'. A *z-ordem* das células tende a manter próximas células que são também próximas espacialmente.

Cada polígono é aproximado pelo menor conjunto de células, de qualquer nível, que cobre completamente o polígono, até um certo nível de subdivisão definido pela precisão desejada,

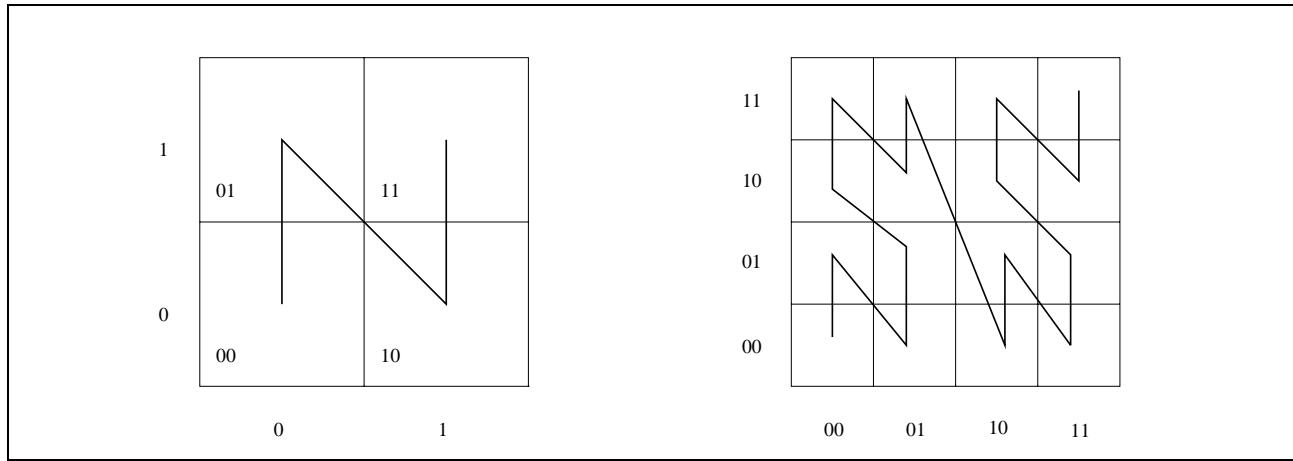


Figura 9.5: Exemplo de z-ordem

conforme ilustrado na figura 9.6. Os endereços das células assim obtidas formam o conjunto de *z-elementos* associado ao polígono [Ore86]. Um conjunto de polígonos é então representado por um conjunto de conjuntos de z-elementos, armazenados em um índice convencional, como por exemplo uma árvore B. Este índice convencional pode então ser utilizado para otimizar várias operações espaciais sobre o conjunto original de polígonos [Ore89].

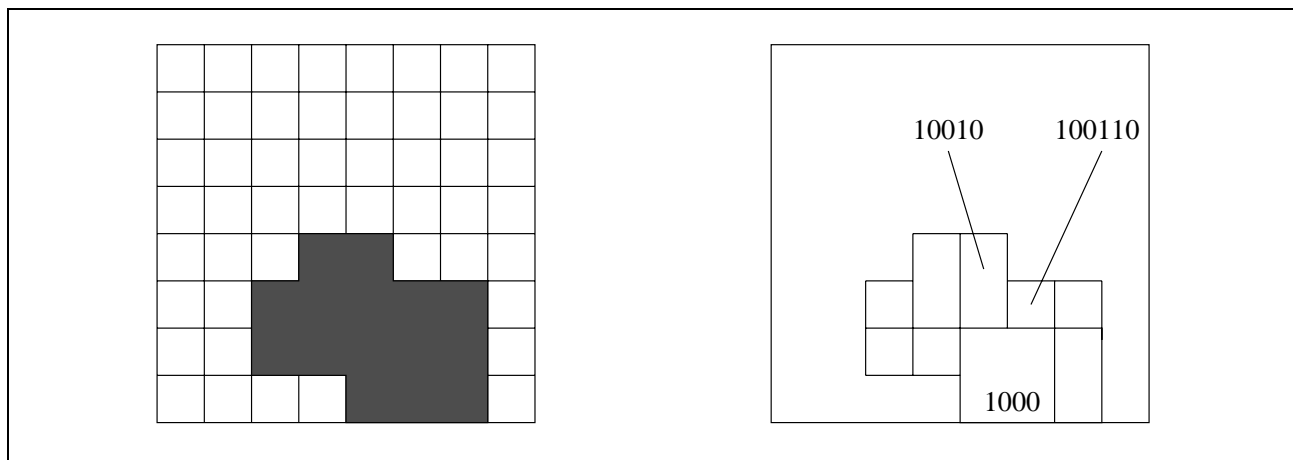


Figura 9.6: Exemplo da decomposição de um polígono em células.

### 9.4.5 Representações Vetoriais Complexas

Todos os métodos de armazenamento e indexação discutidos até o momento cobrem apenas os objetos vetoriais mais elementares - pontos, retângulos, linhas e polígonos. Esta seção discute como compor tais métodos para armazenar representações vetoriais complexas, utilizando como exemplo mapas temáticos. Maiores detalhes podem ser encontrados em [BHKS93, Med95].

Recorde que a geometria de uma representação por sub-divisão planar é definida por um

conjunto de polígonos que particionam o espaço de interesse, ou seja, que não se sobrepõem e que cobrem completamente o espaço de interesse. Por simplicidade, introduz-se uma *região exterior* fictícia que circunda completamente o espaço de interesse. Este conjunto de polígonos pode ser completamente definido por um conjunto de pontos, chamados de *vértices*, e por um conjunto de linhas, chamadas de *arestas*, tais que:

- cada linha começa e termina em um vértice (que pode ser o mesmo, no caso de linhas fechadas);
- duas linhas só se tocam nos vértices;
- cada linha possui uma *região à esquerda* e uma *região à direita*, uma das quais podendo ser a região exterior.

As operações típicas sobre uma representação por sub-divisão planar exigem que se determine, dado um ponto arbitrário, a que região ele pertence e, dada uma linha arbitrária, que arestas ela cruza e em que pontos. Estas operações e outras podem ser otimizadas escolhendo-se métodos de indexação apropriados para os polígonos, vértices e arestas que compõem o mapa temático. Para otimizar ainda mais o processamento, as entradas nas várias estruturas podem ser modificadas para incluírem referências cruzadas entre si. Por exemplo, uma entrada correspondente a um vértice apontaria também para as entradas (na estrutura armazenando arestas) correspondentes às arestas que começam ou terminam no vértice.

Esta descrição sucinta não torna clara, porém, a questão do armazenamento em si das representações exatas dos objetos espaciais.

Na seção 10.3.2, discute-se que o esquema de armazenamento em memória secundária deve equacionar tanto o problema de espaço de armazenamento dos objetos espaciais quanto o problema de contigüidade física. No caso de representações por sub-divisão planar estes dois problemas podem ser resolvidos, por exemplo, armazenando os vértices e arestas, com toda informação complementar, em páginas físicas separadamente dos índices. Cada página física conteria então tanto vértices quanto arestas. Para garantir contigüidade física, pode-se indexar o *pool* de páginas físicas através de uma outra árvore *R*, aproximando cada página pelo retângulo envolvente mínimo de todos os objetos nela armazenados, e determinando em que página um objeto irá residir através de uma pesquisa na árvore. Se o critério de divisão dos nós da árvore *R* for cuidadosamente definido, objetos geograficamente próximos serão direcionados para páginas próximas quando da sua inserção.

A referência [Med95] contém um tratamento detalhado do armazenamento de representações por sub-divisão planar que se preocupa com estes problemas.

# Capítulo 10

## Processamento de Transações e Consultas

### 10.1 Introdução

Este capítulo aborda dois aspectos da implementação de um SIG: gerência de transações e otimização de consultas. Estes dois assuntos foram escolhidos por necessitarem de soluções distintas daquelas empregadas em sistemas convencionais. De fato, sistemas convencionais tratam apenas de transações que têm curta duração, quando usuários trabalham isoladamente. No caso de aplicações de Geoprocessamento, a interação com o banco de dados é comumente mais longa exigindo uma revisão dos mecanismos de controle de transações, principalmente a criação de mecanismos para versionamento, atualização cooperativa e bloqueio parcial de objetos.

Há bem poucas referências sobre gerência de transações para o caso específico de aplicações de Geoprocessamento [BK91]. Porém, o padrão de transações para estas aplicações é semelhante às chamadas *transações longas e aninhadas*, típicas de ambientes de desenvolvimento de software [VM90, PS90], ambientes de projeto de VLSI [BK85, CK86] e hipertexto [SCR93], entre outros. A seção 10.2 baseia-se em parte nas referências [DGM95, SCR93].

Os problemas de otimização de consultas a bancos de dados geográficos possuem alguma semelhança com aqueles estudados no contexto de bancos de dados científicos [Gra93] ou bancos de dados orientados a objetos [Cou92, JBG93]. Políticas específicas para otimização de consultas a bancos de dados geográficos podem ser encontradas em [AS91, Ooi90], que tratam separadamente a componente espacial da parte convencional das consultas. A referência [Kri91] discute o efeito de várias técnicas de filtragem sobre o processamento de consultas. Já a referência [HS92b] trata de consultas sobre segmentos de linhas e a referência [Rot91] aborda métodos de junção espacial. [SFGM93, Cif95] propõem *benchmarks* para consultas espaciais e [KBS93] analisa a influência do efeito da forma do polígono envolvente na filtragem. As referências [FK93, KF92, BKS93, BKSS94, HS94, WH92] tratam do uso de estruturas distribuídas em ambientes paralelos para otimização de consultas.

## 10.2 Gerência de Transações

### 10.2.1 Transações em SIG

Em aplicações convencionais de banco de dados, um usuário modifica os dados através de uma seqüência de operações elementares que devem ser executadas como um todo. As operações são tipicamente curtas e envolvem um volume pequeno de dados. Por exemplo, uma transferência de fundos é implementada através de duas atualizações, em contas bancárias distintas, que devem ser ambas executadas ou ambas canceladas, para evitar erros. Esta situação configura o que se chama de uma *transação*, ou seja, uma seqüência de operações que o sistema de gerência de banco de dados deve processar até o fim, ou garantir que o banco de dados não reflète nenhuma delas, desfazendo aquelas que tenham sido total ou parcialmente executadas. Além disto, o sistema deve processar as operações sem interferência de outras transações e garantir que, se as operações começam em um estado consistente do banco de dados, elas terminam em um estado também consistente. Mecanismos para garantir estas propriedades foram amplamente investigados no contexto de aplicações convencionais.

Os usuários de aplicações de Geoprocessamento também executam transações desta natureza, principalmente ao modificar os atributos convencionais de objetos. Porém, eles podem apresentar um padrão de comportamento muito diferente, mais próximo do encontrado em ambientes de desenvolvimento de software, projeto de circuitos VLSI e criação de documentos hipermídia, entre outros. Nestes ambientes, uma equipe de usuários trabalha cooperativamente para produzir uma nova configuração dos objetos. Cada usuário cria uma parte de um todo em várias etapas, possivelmente trabalhando sobre uma situação já existente, que não pode ser congelada. Além disto, a equipe pode criar configurações alternativas dos objetos, ou seja, modificações que não necessariamente serão efetivadas.

O conjunto de modificações criado pela equipe de usuários configura uma *transação longa e aninhada*. Neste contexto, o termo *sessão de trabalho* é freqüentemente utilizado como sinônimo de transação. A transação é longa pois normalmente o trabalho da equipe se desenvolve por muito tempo, possivelmente semanas. Ela é cooperativa porque envolve modificações criadas por usuários distintos, de forma coordenada. Finalmente, a transação pode conter coleções completas de modificações organizadas em *subtransações aninhadas*. Informalmente, uma transação  $T$  contém uma subtransação aninhada  $T'$  quando uma parte das operações de  $T$  é agrupada e tratada como uma transação  $T'$  de tal forma que as operações em  $T'$  possam ser desfeitas ou refeitas pelo sistema como um todo, ou tornadas condicionalmente visíveis a outros usuários.

Por exemplo, considere um banco de dados geográfico contendo dados sobre topografia, vegetação, áreas de preservação ambiental e rede viária de uma região. Suponha que uma equipe de dois engenheiros esteja desenvolvendo o traçado de um novo oleoduto, cada um trabalhando em trechos diferentes e contíguos do oleoduto. Este projeto envolve então trabalho cooperativo, de longa duração, sobre os objetos no banco, configurado como uma única transação longa, contendo duas subtransações aninhadas, uma para cada membro da equipe. Este exemplo será elaborado de forma um pouco diferente na seção 10.2.3.

### 10.2.2 Mecanismos para Implementação de Transações

Esta seção discute alguns mecanismos genéricos propostos para implementação de transações com as características descritas na seção anterior, incluindo tratamento de versões. Estes mecanismos são implementados no Subsistema de Manipulação, discutido na seção 8.2, e ativados através de operações específicas pertencentes à interface deste subsistema. A descrição segue [DGM95, SCR93]. Esta seção é opcional, podendo o leitor passar diretamente para a seção 10.2.3.

Paralelamente aos mecanismos sofisticados para manipulação de dados, descritos abaixo, o usuário poderá ainda manipular diretamente tais dados, para cobrir casos simples como, por exemplo, atualização do valor de um atributo convencional.

Em geral, um ambiente para trabalho cooperativo deve permitir tanto compartilhamento de conjuntos de dados entre usuários quanto a definição de conjuntos privativos a um grupo de usuários. Para tanto, supomos que os dados estão organizados em uma hierarquia de bancos de dados e que cada dado só pertence a um destes bancos. Se  $B'$  é filho de  $B$  na hierarquia, dizemos que  $B'$  é um banco *subordinado* a  $B$ . (Note que a hierarquia possui um número arbitrário de níveis). Cada banco está associado a um grupo de usuários com permissão para realizar um certo conjunto de operações sobre os dados no banco ou em bancos subordinados a ele. Esta hierarquia de bancos reflete o fato de transações conterem outras transações aninhadas. Assim, um grupo de usuários poderá criar um banco de dados para conter os dados com que irá trabalhar e, recursivamente, criar outros bancos subordinados a este, associando-os a transações aninhadas conduzidas por subgrupos de usuários.

Ortogonalmente a estes conceitos, consideramos que um dado pode estar em um de três estados: *pronto*, *em trabalho* ou *obsoleto* (respectivamente, *committed*, *uncommitted* ou *obsolete*). Um dado  $d$  pode ainda estar associado a um conjunto de outros dados, tratados como suas *versões*, e organizados sob forma de uma árvore, onde  $d$  é a raiz. Esta árvore é mantida de tal forma que, se uma versão está pronta, todos os seus ancestrais na árvore também estão prontos; e, se uma versão é obsoleta, todos os seus descendentes também são obsoletos. Os dados são manipulados através do elenco de operações descritas a seguir.

A operação *update* permite atualizar um dado em trabalho, sem alterar o seu estado.

A operação *commit* transforma em pronto um dado em trabalho.

A operação *delete* remove um dado  $d$  de um banco  $B$ . Se  $d$  for um dado em trabalho, ele é efetivamente destruído; se  $d$  for um dado pronto, ele é tornado obsoleto. O usuário também pode remover um banco  $B$ , provocando a remoção de todos os dados e bancos subordinados a  $B$ , recursivamente. O sistema se encarregará de manter dados obsoletos apenas enquanto eles forem referenciados por outros dados.

Há três operações para criação de dados: *create* cria um dado  $d'$  completamente novo em um banco  $B'$ ; *create-version* cria uma nova versão  $d'$  de um dado  $d$  pronto ou em trabalho no mesmo banco de dados  $B'$  que contém  $d$ ; *check-out* cria um novo dado  $d'$  em  $B'$  como uma versão de um dado pronto  $d$  existente em  $B$ , onde  $B'$  é um banco subordinado a  $B$ . O usuário



também poderá utilizar esta operação para criar em bloco versões de um conjunto de dados. Note que o usuário não pode mover  $d$  de  $B$  para  $B'$ , mas apenas usar a operação *check-out* para criar uma versão  $d'$  de  $d$  em  $B'$ . Em todos estes três casos,  $d'$  é considerado um dado em trabalho após a sua criação.

A operação *check-in* permite mover um dado pronto  $d'$  de um banco  $B'$  para o banco  $B$  a que  $B'$  se subordina. O usuário também poderá utilizar esta operação para mover em bloco um conjunto de dados de um banco para outro. Há duas variantes desta operação. A primeira delas, *check-in de adição*, de fato move  $d'$  de  $B'$  para  $B$ , exceto se  $d'$  for uma versão de um dado  $d$  existente em  $B$  e  $d'$  não tiver sido modificado, caso em que  $d'$  é descartado, evitando assim a criação de uma réplica de  $d$  em  $B$ . A segunda, *check-in por substituição*, move  $d'$  de  $B'$  para  $B$ , se  $d'$  for um novo dado, ou move  $d'$  de  $B'$  para  $B$  e remove o dado  $d$  existente em  $B$  (através da operação *delete*), se  $d'$  for uma versão de  $d$  e  $d'$  tiver sido modificado. Esta última operação corresponde portanto à tradicional operação de atualização.

Estas operações mantêm as árvores de versões consistentes através de ações colaterais. Assim, a operação *commit* aplicada a um dado se propaga recursivamente para todos os seus ancestrais na árvore, transformando-os também de dados em trabalho para prontos. A operação *delete* aplicada a um dado se propaga recursivamente para todos os seus descendentes na árvore, destruindo-os ou tornando-os obsoletos. Este efeito pode ser provocado também por uma operação de *check-in por substituição* ao chamar implicitamente uma operação *delete*.

Note que as árvores de versões cruzam a hierarquia de bancos de dados. Portanto, quando um usuário executa uma operação de *commit* sobre um dado em um banco  $B$ , ele poderá afetar um dado que pertence ao banco a que  $B$  se subordina, e assim recursivamente. Assim, o usuário deverá possuir permissão para executar *commit* também sobre dados nestes outros bancos. Porém, a operação *delete* não causa problemas. Por definição, se o usuário possui permissão para aplicar a operação de *delete* sobre dados de  $B$ , ele também possui permissão para aplicar *delete* a dados em bancos subordinados a  $B$ , recursivamente. A mesma observação se aplica à operação de *check-in por substituição*. Assim, como efeito colateral de uma operação sobre um dado  $d$ , um usuário poderá interferir com bancos de dados de outros usuários que contêm versões de  $d$ .

Para minimizar este problema, podemos lançar mão de mecanismos de controle de acesso, permitindo o bloqueio de dados em três modos: *compartilhado*, *exclusivo* e *versão-exclusivo*. Quando um usuário (ou grupo de usuários) bloqueia um dado em modo compartilhado, ele indica uma potencial intenção de substituí-lo por outra versão; em modo exclusivo, ele proíbe outros usuários de substituí-lo por outra versão; em modo versão-exclusivo, ele proíbe outros usuários de criar versões do dado.

Este recurso se integra às operações descritas anteriormente da seguinte forma. Em uma operação de *check-out* que cria um novo dado  $d'$  em  $B'$  como uma versão de um dado  $d$  em  $B$ ,  $d$  é bloqueado em modo compartilhado para o usuário que executa a operação. Ao executar um *check-in por substituição* sobre  $d'$ , se  $d'$  tiver sido modificado, o bloqueio de  $d$  é escalonado para o modo exclusivo e os outros usuários que possuíam bloqueios compartilhados sobre  $d$  têm seus bloqueios descartados. Isto os impede de retornar versões de  $d$ , forçando-os a refazer o

*check-out*, agora sobre  $d'$  (dado que substituiu  $d$ ).

Esta discussão é simplificada pois não explora em detalhe as referências cruzadas entre dados, quer seja porque um dado é uma componente de outro, quer seja porque há um relacionamento explícito entre ambos. A referência [DGM95] descreve como este problema é resolvido no contexto do sistema SAGRE (abordado no capítulo 12). No entanto, ela tampouco considera a criação de versões ou o bloqueio de partes de um conjunto de dados. Esta extensão é particularmente importante pois geo-objetos ou geo-campos frequentemente cobrem áreas geográficas maiores do que a região de interesse do usuário. A seção seguinte apresenta um exemplo de modificação de um banco de dados geográfico que utiliza extensões das operações acima descritas cobrindo este ponto.

### 10.2.3 Exemplo

Considere um banco de dados geográfico  $B$  contendo, para o Estado do Rio de Janeiro, um geo-campo  $T$  capturando a sua topografia, uma coleção de geo-objetos, chamada de ÁREAS-DE-PROTEÇÃO, cujos elementos descrevem as áreas de proteção, representadas em uma representação por subdivisão planar  $P$ , e duas coleções de geo-objetos, REDE-VIÁRIA e OLEODUTOS, com a descrição das estradas e oleodutos, cujas localizações encontram-se em uma representação complexa  $V$ .

Suponha agora que uma equipe tenha sido designada para elaborar uma proposta para um oleoduto do terminal da Petrobrás no Município de Angra dos Reis para o Porto de Sepetiba, sem cruzar áreas de proteção ambiental e aproveitando ao máximo os cortes já feitos para as estradas. O trabalho da equipe configura uma transação longa e cooperativa, construída interativamente da seguinte forma.

Inicialmente a equipe define uma janela  $R$  cobrindo toda a região de trabalho. Em seguida, define o banco de dados privado  $B'$  e cria, através da operação *check-out*, versões  $T_R$  e  $V_R$  dos objetos  $T$  e  $V$  apenas para a região  $R$  e uma versão  $OL$  de OLEODUTOS. Suponha que a equipe tenha permissão apenas para consultar as coleções ÁREAS-DE-PROTEÇÃO e REDE-VIÁRIA e a representação  $P$ .

Em seguida, a equipe define o oleoduto, inserindo um novo geo-objeto  $O$  em  $OL$ , e alterando  $V_R$  para incluir a representação de  $O$ . A equipe também modifica o geo-campo  $T_R$  para indicar os novos cortes necessários à passagem do oleoduto (mas não modifica  $P$ , por restrições de projeto).

Ao término, a equipe introduz o resultado do projeto no banco original  $B$  através da operação de *check-in por substituição*, estendida para tratar de versões que se referem a apenas uma parte da área geográfica coberta pelos objetos.

Em paralelo, uma segunda equipe poderá estar trabalhando no traçado de uma nova estrada no Município de Angra dos Reis, com restrições semelhantes às anteriores: a estrada não poderá cruzar áreas de proteção ambiental e deverá aproveitar ao máximo os cortes já feitos para oleodutos.

Novamente, a equipe define uma janela  $S$  cobrindo sua região de trabalho e define o banco de dados privado  $E$ , criando através da operação *check-out* versões  $T_S$  e  $V_S$  dos objetos  $T$  e  $V$  apenas para a região  $S$  e uma versão  $RV$  de REDE-VIÁRIA.

De forma semelhante, esta equipe define a estrada, inserindo um novo geo-objeto  $E$  na coleção  $RV$ , e alterando  $V_S$  para incluir a representação de  $E$ , e o geo-campo  $T_S$  para indicar novos cortes necessários à passagem da nova estrada.

Ao término, a equipe tentará introduzir o resultado deste segundo projeto no banco original  $B$  através da operação de *check-in por substituição*.

Duas situações poderão ocorrer. Se as regiões  $R$  e  $S$  forem disjuntas, então o trabalho de uma equipe não afeta diretamente o da outra e a alteração do banco  $B$  ocorrerá normalmente. Porém, a implementação da operação de *check-in por substituição* deverá ser suficientemente sofisticada para alterar  $V$  e  $T$  apenas para a região  $S$  de tal forma que não destrua as modificações efetuadas nestes campos pela equipe que criou o oleoduto.

Já se as regiões  $R$  e  $S$  não forem disjuntas, o trabalho de uma equipe potencialmente afeta o da outra. À guisa de ilustração, façamos uma análise mais detalhada desta situação. A primeira equipe altera  $V$  acrescentando a representação de  $O$  e a segunda acrescentando a representação de  $E$ . Portanto, se o sistema versionar as componentes de  $V$ , e não  $V$  como um todo, será simples implementar a operação de *check-in por substituição* de tal forma que o trabalho de uma equipe não interfira com o da outra do ponto de vista de  $V$ : basta reconhecer que ambas as equipes simplesmente criaram novas componentes de  $V$ . Argumento semelhante se aplica a  $T$ , exceto que, neste caso, como  $T$  é um geo-campo será necessário analisar a representação utilizada para  $T$ . Por exemplo, se a representação for por isolinhas, será necessário versionar as isolinhas em si; assim uma equipe não interferirá com a outra se alterarem isolinhas distintas.

Naturalmente, toda esta discussão sobre a operação de *check-in por substituição* é afetada pelos mecanismos de controle de acesso, que também deverão ser revistos para acomodar os refinamentos baseados no uso de regiões ou de componentes.

## 10.3 Processamento de Consultas

### 10.3.1 Tipos de Consultas

A eficácia das estratégias de otimização de consultas e dos métodos de armazenamento e indexação das componentes espaciais dos dados geográficos depende fundamentalmente do tipo e frequência das consultas e operações em geral submetidas a bancos de dados geográfico. Embora estes fatores dependam da aplicação em questão, esta seção sugere uma classificação das consultas e operações bastante útil para o estudo do problema, seguindo [BHKS93].

Podemos separar de um lado as operações de atualização - inserção, remoção e modificação - submetidas ao banco de dados geográfico, e de outro lado as consultas espaciais em si. Dentre os tipos de consultas espaciais, destacamos quatro:

**seleção convencional** : dado um conjunto de dados  $D$  e definida uma restrição  $\rho$  sobre atributos convencionais dos dados em  $D$ , determine todos os dados em  $D$  que satisfazem  $\rho$ .

**seleção pontual** : dado um ponto  $P$  e um conjunto de dados  $D$ , determine todos os dados em  $D$  que contêm  $P$ .

**seleção por região** : dada uma região  $R$  e um conjunto de dados  $D$ , determine todos os dados em  $D$  que interceptam ou estão contidos em  $R$ . Um subtipo simples e importante é a chamada *seleção por janela* em que a região é um retângulo com os lados paralelos aos eixos.

**junção espacial** : dados dois conjuntos de dados,  $D$  e  $D'$ , e definido um relacionamento espacial  $\theta$  entre pares de dados (por exemplo “estar contido em”), determine todos os pares de dados  $(d, d') \in D \times D'$  tais que  $d$  e  $d'$  satisfazem  $\theta$ .

A seleção convencional foi incluída na lista por uma questão de completude, já que não envolve a componente espacial. Note que o resultado de consultas de quaisquer um destes tipos retorna um conjunto de dados, ou um conjunto de pares de objetos, retirados do banco de dados geográfico. Existe no entanto uma variante importante da junção espacial que não retorna simplesmente pares de objetos:

**superposição de representações** : dados dois conjuntos de dados  $D$  e  $D'$  e um critério de superposição  $\sigma$ , determine todos os pares de dados  $(d, d') \in D \times D'$  tais que  $d$  e  $d'$  se superpõem e crie um novo dado para cada par conforme o critério de superposição  $\sigma$ . O resultado da operação consiste nestes novos dados.

### 10.3.2 Estratégia para Processamento de Consultas

Conforme a discussão do capítulo 8, dada uma consulta sobre um banco de dados geográfico, o Subsistema de Manipulação (SM) contém um otimizador de consultas que produz um *plano de execução* para a consulta consistindo de comandos que serão executados pelos Subsistemas de Manipulação Convencional (SMC), Vetorial (SMV) e Matricial (SMM). O otimizador contém um componente capaz de gerar planos alternativos para execução de consultas e uma função de custo associando um custo estimado a cada plano. O gerador de planos incorpora ainda heurísticas que limitam o espaço de busca, evitando uma explosão combinatorial de planos alternativos.

Tipicamente, a parte do plano responsável pelo processamento da componente espacial da consulta contém três fases [Ore89, Kri91]:

#### *Fase 1: Filtragem*

Um banco de dados geográfico normalmente contém estruturas de dados auxiliares, chamadas de *índices espaciais* ou *estruturas de indexação espacial* que facilitam a localização dos

dados cuja componente espacial satisfaz à qualificação de uma consulta. Reservamos o termo *método de indexação espacial* para designar uma família de estruturas de indexação espacial. De fato, diversos métodos de indexação espacial foram discutidos no capítulo 9.

O desempenho deste passo depende da distribuição espacial dos dados e da forma como a estrutura decompõe o espaço, sendo que em geral é baixo o fator de seletividade [CHF93b].

Dada a complexidade da componente espacial e da semântica dos próprios operadores espaciais, em geral os índices trabalham apenas com aproximações das componentes espaciais (por exemplo, o retângulo mínimo envolvente). Assim, o uso de índices para resolver a parte espacial de uma consulta apenas filtra, dos conjuntos de dados armazenados no banco de dados, aqueles que certamente não satisfazem à qualificação da consulta, sendo necessário um segundo passo.

### *Fase 2: Refinamento*

Uma vez identificados os dados que se candidatam a satisfazer a qualificação da consulta, as geometrias exatas das componentes espaciais dos dados são trazidas para memória principal. É sobre estas geometrias que são então executados os operadores espaciais especificados na qualificação. As estruturas de armazenamento e indexação devem preferencialmente permitir que seja trazida para memória principal apenas a parte das componentes espaciais necessária à execução dos operadores.

O desempenho deste passo depende da quantidade de dados recuperados, da sua complexidade e do custo dos operadores a aplicar, que em geral é elevado.

### *Fase 3: Pós-Processamento*

Depois de identificados os dados que satisfazem à qualificação da consulta, usualmente é necessário passá-los para as camadas superiores da arquitetura onde sofrerão processamento posterior ou serão visualizados. Assim, o sistema deve oferecer um mecanismo eficiente para passar a geometria das componentes espaciais dos dados, na sua totalidade ou em parte, para as camadas superiores.

Cabe ao Subsistema de Armazenamento Físico (SAF) implementar os *métodos de armazenamento*, que governam a utilização das páginas físicas em memória secundária, onde residem tanto as componentes espaciais dos dados quanto os próprios índices espaciais. Os Subsistemas de Armazenamento Matricial (SAM) e de Armazenamento Vetorial (SAV) implementam métodos de indexação espacial e métodos de armazenamento específicos para as componentes espaciais.

Estes subsistemas, em conjunto, devem equacionar dois problemas:

**espaço de armazenamento:** a representação interna de uma componente espacial pode necessitar um espaço substancial, ocupando mais de uma página física;

**contiguidade física:** as páginas físicas de uma mesma componente espacial ou de componentes contíguas no espaço devem ser vizinhas em memória secundária, diminuindo o custo de acesso.

Uma estratégia para resolver estes problemas consiste em armazenar a geometria exata das

componentes espaciais dos dados fora dos índices, digamos, em um *pool* comum de páginas físicas. Esta forma de armazenamento leva a índices mais compactos e não impõe limitações ao tamanho das componentes, resolvendo assim o problema de espaço de armazenamento. Porém, para endereçar o problema de contigüidade física, é necessário definir uma estratégia de gerência para o *pool* de páginas físicas de tal forma que componentes vizinhas no espaço geográfico tenham as suas geometrias armazenadas em páginas próximas e, idealmente, que permita armazenar nas mesmas páginas diferentes tipos de geometrias.

### 10.3.3 Exemplos de Otimização de Consultas

Esta seção aborda, por meio de exemplos, o problema de otimização de consultas espaciais, à luz dos comentários feitos na seção 10.3.2.

#### Exemplo de Seleção por Região

Considere um banco de dados geográfico sobre cidades brasileiras consistindo de apenas um conjunto de geo-objetos CIDADES. Para cada cidade (do mundo real), há um objeto em CIDADES, cujos atributos convencionais correspondem aos dados convencionais da cidade e cuja localização é dada por um par ordenado indicando a latitude e longitude da cidade; ou seja, a localização destes geo-objetos é representada por um ponto em coordenadas geográficas. No que se segue, suporemos que estas coordenadas estão armazenadas junto com o próprio geo-objeto, embora todas as operações sobre elas sejam implementadas pelo SMV e qualquer índice espacial sobre CIDADES seja de responsabilidade do SAV.

Considere agora a consulta  $Q$ , formulada informalmente como:

$Q$ . *Selecione o nome e população de todas as cidades a menos de 50km da cidade de Belo Horizonte e com mais de 50 mil habitantes.*

Em LEGAL, esta consulta seria definida através do comando:

```
SELECT d.nome, d.populacao
FROM d Cidade, c Cidade
WHERE c.nome = "Belo Horizonte"
      and distance(d,c) < 50
      and d.populacao > 50.000
```

Esta consulta exemplifica uma seleção por região, definida por um círculo de raio 50km e centro na localização geográfica da cidade de Belo Horizonte. Há essencialmente três planos de execução mais plausíveis para esta consulta (entre outros possíveis), discutidos a seguir.

Considere inicialmente o plano de execução  $P_1$ :

$P_{11}$ . *Determine a posição  $b$  de Belo Horizonte;*

$P_{12}$ . *Determine o conjunto  $C'$  de todas as cidades a 50km de  $b$ ;*

$P_{13}$ . *Determine o subconjunto  $C$  de  $C'$  de cidades com mais de 50 mil habitantes.*

As subconsultas  $P_{11}$  e  $P_{13}$  não envolvem nenhum operador espacial e, portanto, são subconsultas convencionais de  $Q$  para este plano. Já a subconsulta  $P_{12}$  envolve o operador de distância e é a única subconsulta espacial de  $Q$  em  $P_1$ .

O processamento de  $P_1$  inicia-se então com a execução, pelo SMC, da subconsulta convencional  $P_{11}$ , que retorna a localização da cidade de Belo Horizonte, o ponto  $b$ .

Suponha que o banco de dados inclua um índice espacial sobre CIDADES, que chamaremos de ID-CIDADES, invisível aos usuários, que facilite identificar todos os objetos de CIDADES cuja localização esteja a uma certa distância de um dado ponto. Suponha ainda que o índice espacial seja impreciso para este caso, no sentido de também retornar objetos que não estão à distância especificada do ponto. Esta imprecisão é típica de métodos de indexação espacial, conforme ilustrado na seção 9.4.1, e contrasta com os métodos de indexação tradicionais, que são tipicamente precisos.

O processamento da subconsulta  $P_{12}$ , de responsabilidade do SMV, segue em três passos:

$P_{121}$ . *Através de ID-CIDADES, determine que objetos em CIDADES podem estar a menos de 50km de  $b$ , criando o conjunto  $I$ ;*

$P_{122}$ . *Leia da memória secundária para a memória principal a representação interna dos objetos indicados por  $I$ , criando o conjunto  $C''$ ;*

$P_{123}$ . *Determine que elementos de  $C''$  estão de fato a menos de 50km de  $b$ , criando o conjunto  $C'$ .*

Para utilizar o índice ID-CIDADES, no passo  $P_{121}$ , e para ler a representação interna dos objetos, no passo  $P_{122}$ , o SMV invoca o SAV. Os elementos de  $I$  são apontadores para a localização em memória secundária da representação interna de objetos em CIDADES. Além disto, como supomos que o índice não é exato,  $I$  poderá apontar para objetos a mais de 50km de  $b$ . Já os elementos de  $C'$  e  $C''$  são representações internas de objetos em CIDADES, armazenadas em memória principal. Assim, o passo  $P_{121}$  corresponde à fase de filtragem e  $P_{122}$  e  $P_{123}$ , à fase de leitura e refinamento.

O processamento da subconsulta  $P_{13}$  é agora simples, bastando extrair de cada elemento de  $C'$  os valores dos atributos de interesse e eliminando aqueles que não possuem mais de 50 mil habitantes, criando assim o conjunto  $C$ , que é a resposta da consulta  $Q$ . Os elementos de  $C$  serão pares de valores, correspondendo ao nome e à população de uma cidade. O processamento de  $P_{121}$ ,  $P_{122}$ ,  $P_{123}$  e  $P_3$  poderá ser concatenado (*pipelined*), ou seja, cada elemento identificado em  $P_{121}$  é imediatamente processado por  $P_{122}$  e filtrado por  $P_{123}$ , eventualmente gerando um par em  $C$ , ao ser tratado em  $P_{13}$ . Isto evita a criação explícita dos resultados intermediários  $I$ ,  $C'$  e  $C''$ , levando a uma razoável economia no processamento global da consulta.

O segundo plano,  $P_2$ , inverte a ordem de execução das restrições da consulta  $Q$ :

$P_{21}$ . *Determine a posição  $b$  de Belo Horizonte;*

$P_{22}$ . *Determine o conjunto  $C'$  de todas as cidades com mais de 50 mil habitantes;*

$P_{23}$ . *Determine o subconjunto  $C$  de  $C'$  de cidades a 50km de  $b$ .*

Os elementos de  $C'$  serão triplas em memória principal indicando o nome, população e localização de objetos em CIDADES com mais de 50 mil habitantes. O processamento da subconsulta convencional  $P_{22}$  pelo SMC será particularmente eficiente se partirmos do pressuposto que o banco de dados inclui um índice convencional sobre CIDADES, que chamaremos de ID-POP, invisível aos usuários, que permita identificar todos os objetos de CIDADES cuja população esteja em uma faixa de valores. Tal índice será provavelmente preciso, no sentido de retornar exatamente o subconjunto de cidades cuja população está na faixa de valores dada. Porém, note que o conjunto  $C'$  corresponderá a todas as cidades brasileiras com mais de 50 mil habitantes. Neste caso, a execução de  $P_{22}$  dá-se da seguinte forma:

$P_{221}$ . *Através de ID-POP, determine que objetos em CIDADES possuem população com mais de 50 mil habitantes, criando o conjunto  $J$ ;*

$P_{222}$ . *Leia da memória secundária para a memória principal a representação interna dos objetos indicados por  $J$ , selecionando seu nome, população e localização e criando o conjunto  $C'$ .*

O processamento da subconsulta  $P_{23}$ , de responsabilidade do SMV, é então simplificado, bastando selecionar do conjunto  $C'$  (suposto já em memória principal) aqueles elementos que estão de fato a menos de 50km de  $b$ , criando o conjunto  $C$ . Novamente, o processamento de  $P_{221}$ ,  $P_{222}$  e  $P_{23}$  poderá ser sincronizado, evitando a criação de  $J$  e  $C'$ .

O terceiro plano,  $P_3$ , constrói dois conjuntos independentemente para, em seguida, computar a sua interseção:

$P_{31}$ . *Determine a posição  $b$  de Belo Horizonte;*

$P_{32}$ . *Determine o conjunto  $C'$  de todas as cidades com mais de 50 mil habitantes;*

$P_{33}$ . *Determine o conjunto  $C''$  de cidades a 50km de  $b$ ;*

$P_{34}$ . *Determine a interseção  $C$  de  $C'$  e  $C''$ .*

A execução detalhada deste plano é uma combinação da discutida para os dois planos anteriores.

Até este ponto, descrevemos apenas o detalhamento de três planos plausíveis para execução da consulta. A estimativa do custo de cada plano depende essencialmente de estimar o número de objetos identificados pelo índice ID-CIDADES e pelo índice ID-POP e está fora do escopo deste texto. Por exemplo, o plano  $P_1$  será mais vantajoso do que  $P_2$  se ID-CIDADES for razoavelmente eficiente, ou seja, não retornar muitas cidades que não estão a menos de 50km de Belo Horizonte, e se o número de cidades brasileiras com mais de 50 mil habitantes for muito grande. Em outras palavras, se a construção da resposta intermediária registrada no conjunto  $I$  for mais barata do que a registrada no conjunto  $J$ .



### Exemplo de Junção Espacial

Suponha agora que o banco de dados geográfico contém também um conjunto de geo-objetos AEROPORTOS tal que, para cada aeroporto brasileiro, há um objeto em AEROPORTOS cujos atributos convencionais correspondem aos dados convencionais do aeroporto e cuja localização é dada novamente pela latitude e longitude do aeroporto. Suponha ainda que o banco inclua um índice espacial sobre AEROPORTOS chamado ID-AEROPORTOS, que permita identificar todos os aeroportos cuja localização esteja a uma certa distância de um dado ponto.

Considere a seguinte consulta “Selecione todas as cidades que são sede de municípios e que são atendidas por aeroportos pavimentados”. Esta consulta exemplifica uma junção espacial. Para torná-la mais precisa, suponha que uma cidade é atendida por um aeroporto se a cidade dista menos de 50km do aeroporto. A consulta reformulada seria então:

*Q. Selecione o nome de cada cidade e o nome de cada aeroporto tais que a cidade é sede de município, o aeroporto é pavimentado e a cidade dista menos de 50km do aeroporto.*

Novamente há vários planos de execução possíveis para esta consulta, sendo os principais discutidos a seguir. Todos os planos, sob certo aspecto, reduzem o processamento da junção espacial ao processamento de várias seleções por região.

Considere o plano  $P_1$  inicialmente:

$P_{11}$ . *Determine o conjunto  $R$  contendo o nome  $n$  e localização  $l$  de cada aeroporto pavimentado;*

$P_{12}$ . *Inicialize o conjunto  $C$  como vazio;*

$P_{13}$ . *Para cada elemento  $(n, l) \in R$ , faça:*

$P_{131}$ . *Determine o conjunto dos nomes  $c$  de cidades que são sede de município e que estão a menos de 50km de  $l$ , acrescentando o par  $(c, n)$  a  $C$ .*

O processamento de  $P_1$  inicia-se com a execução, pelo SMC, da subconsulta convencional  $P_{11}$ , que retorna o nome e a localização de todos os aeroportos pavimentados. Para cada um destes, executa-se uma seleção por região para determinar as sedes dos municípios que estão a menos de 50km do aeroporto. O processamento destas seleções segue então de acordo com uma das estratégias discutidas no exemplo anterior.

O segundo plano,  $P_2$ , é simétrico a este, recuperando primeiro o conjunto de todas as cidades que são sede de município para depois determinar quais aeroportos pavimentados estão a menos de 50km de cada cidade. Como supusemos que há um índice espacial sobre o conjunto de aeroportos, este segundo plano compete com o primeiro. A determinação de qual dos dois planos é mais vantajoso depende de qual conjunto for menos numeroso: os aeroportos pavimentados ou as cidades que são sede de município.

O terceiro plano,  $P_3$ , constrói dois conjuntos independentemente para, em seguida, computar a sua interseção:

$P_{31}$ . Determine o conjunto  $R$  contendo o nome  $n$  e localização  $l$  de cada aeroporto pavimentado;

$P_{32}$ . Determine o conjunto  $S$  contendo o nome  $c$  e localização  $s$  de cada sede de município;

$P_{33}$ . Determine o conjunto  $C$  tal que  $(c, l) \in C$  sse  $(n, l) \in R$  e  $(c, s) \in S$  e a distância de  $l$  a  $s$  é menor do que 50km.

O passo  $P_{33}$  esconde uma iteração sobre os elementos de  $R$  e sobre os elementos de  $S$ . Em um caso, por exemplo, a cada elemento de  $R$ , todos os elementos de  $S$  seriam visitados para determinar quais satisfazem a condição imposta. O plano  $P_3$  nada mais seria então do que uma reformulação do plano  $P_1$ .

É possível, no entanto, implementar outras formas de computar o passo  $P_{33}$  que tornam o plano  $P_3$  diferente dos demais. Discutiremos a seguir, brevemente, como aplicar o *método de varredura do espaço* para computar este passo. Intuitivamente, este método consiste em varrer o espaço em uma dada direção processando todos os objetos encontrados. No caso abaixo, usaremos um meridiano, varrendo o plano em ordem crescente de valor de longitude:

$P_{331}$ . Ordene inicialmente os elementos de  $R$  e  $S$  em valor crescente da longitude das suas localizações;

$P_{332}$ . Para cada elemento de menor longitude ainda não processado, faça:

$P_{3321}$ . Suponha que este elemento seja um par  $(n, l) \in R$  (se for um par  $(c, s) \in S$ , o processamento é inteiramente semelhante);

$P_{3322}$ . Determine todos os pares  $(c, s) \in S$  tais que a distância de  $l$  a  $s$  é menor do que 50km;

$P_{3323}$ . Considere o par  $(n, l)$  e todos os pares  $(c, s)$  encontrados como processados.

Novamente a escolha entre os planos depende de uma estimativa do custo de cada um, discussão que está fora do escopo deste texto.

### Exemplo de Superposição de Representações

Considere agora que o banco de dados geográfico contém dois geo-campos temáticos, SOLOS e VEGETAÇÃO, indicando os tipos de solo e de vegetação do Estado de São Paulo. Suponha que os geo-campos possuam representações por subdivisão planar. Suponha ainda, por simplicidade, que as duas representações utilizem a mesma escala e projeção de tal forma que seja possível operar sobre elas sem conversão.

Considere agora a consulta  $Q$ , formulada como:

$Q$ . Crie um geo-campo temático com o remanescente de mata atlântica em latossolo roxo na região  $B$

onde  $B$  é uma *janela* – um retângulo definido por um par  $(ie, sd)$ , no mesmo sistema de coordenadas cartográficas das representações, indicando o canto inferior esquerdo e superior direito do retângulo. Esta consulta difere das anteriores por criar um campo temático a partir de dois outros armazenados no banco de dados, tendo uma restrição dada por um retângulo.

Discutiremos a execução de apenas um plano,  $P_1$ , para esta consulta:

- $P_{11}$ . Utilizando SOLOS, crie o campo temático SOLOS-B indicando os tipos de solo em  $B$ ;
- $P_{12}$ . Utilizando VEGETAÇÃO, crie o campo temático VEGETAÇÃO-B indicando os tipos de vegetação em  $B$ ;
- $P_{13}$ . Crie o campo temático VEGET-REM, resposta da consulta, aplicando uma variante da operação de superposição de campos a SOLOS-B e VEGETAÇÃO-B;

Como ambos os campos possuem representação vetorial, a execução de  $P_1$  é de inteira responsabilidade do SMV, com auxílio do SAV.

Ao contrário dos exemplos anteriores em que a otimização do plano dependia da existência de índices espaciais, o processamento será facilitado aqui se supusermos a existência de estruturas de armazenamento eficientes para as representações que facilitem recuperar as parcelas das representações que cobrem o retângulo  $B$ . Suponha então que o SAV armazene as representações envolvidas em estruturas que permitam trazer para memória principal apenas as páginas que contêm os dados de vegetação ou solos na região  $B$  ou, pelo menos, em uma região  $B'$  que cubra  $B$  e seja significativamente menor do que a região abrangida pelas representações. Neste segundo caso, os dados devem sofrer um processo de filtragem no SAV antes de passar para o SMV.

Em uma implementação mais simples, o SAV recuperará as páginas apropriadas, filtrando-as se necessário, e materializará as representações vetoriais dos campos SOLOS-B e VEGETAÇÃO-B. Sobre estes campos intermediários, o SMV aplicará então a operação de superposição de representações apropriada para criar o campo temático VEGET-REM, resposta da consulta. Já em uma implementação mais sofisticada, o SAV tratará os campos SOLOS-B e VEGETAÇÃO-B como *visões* não materializadas dos campos originais. Mais explicitamente, quando o SMV executar a operação de superposição de representações, o SAV recuperará, para memória principal, as páginas contendo os dados de vegetação ou solos à medida que se tornem necessárias para a operação, sendo de fato criada apenas a representação do campo temático VEGET-REM.

## PARTE IV



# Capítulo 11

## O Sistema SPRING

### 11.1 Introdução

Este capítulo descreve as características principais do SPRING ("Sistema para Processamento de Informações Georeferenciadas"), desenvolvido no INPE, com ênfase na versão 2.0 do sistema.

A versão 1.0 do SPRING, lançada em 1993, concretizava o esforço, levado a cabo no INPE, para desenvolvimento de uma nova geração de sistemas de Processamento de Imagens e Geoprocessamento. A motivação básica para esta iniciativa foi integrar as tecnologias de Sensoriamento Remoto e Sistemas de Informação Geográfica, utilizar um modelo de dados orientado a objetos e fornecer ao usuário um ambiente interativo para visualizar, manipular e editar dados geográficos [CSFP93].

A versão 2.0 do SPRING foi concebida para ser um SIG de segunda geração, trabalhando em um ambiente cliente-servidor e funcionando acoplado a SGBDs. Os objetivos desta versão foram: dar suporte a um banco de dados geográficos de grande porte, sem estar limitado pelo recorte de projeções cartográficas; aprimorar a integração de dados geográficos, com a introdução do conceito de modelagem de geo-campos e geo-objetos; obter escalabilidade, sendo o sistema capaz de operar com funcionalidade plena em microcomputadores pessoais e em estações de trabalho; e prover uma interface que combine aplicações comandadas por menus e uma linguagem de consulta e manipulação espacial.

O SPRING oferece interfaces de janela nos padrões Motif, Windows e SVT ("Spring Virtual Toolkit"), sendo este último um padrão proprietário implementado como uma classe virtual que permite a definição de elementos de interface. Além desses, está disponível também um módulo gerador de interfaces, que faz uso do aplicativo X-Designer.

Além do SPRING, o INPE também disponibiliza uma biblioteca de classes de uso geral, comum para diversas plataformas de mercado, e um ambiente para desenvolvimento e teste de novas tecnologias em SIG. A biblioteca de classes, chamada SPRINGlib, possui código na linguagem C++ (multiplataforma), implementando os objetos do modelo conceitual e de representação como classes C++. A fim de garantir seu uso em diversas plataformas, há aplicativos

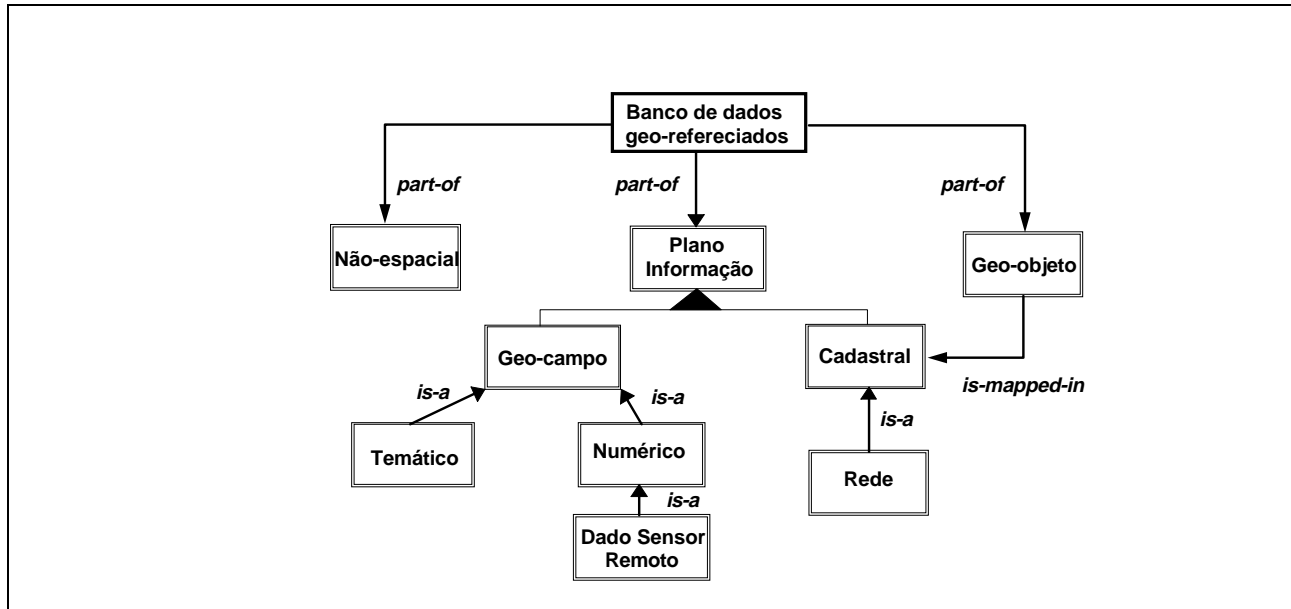


Figura 11.1: Nível conceitual do modelo de dados

independentes do sistema de janelas adotado e classes que permitem a independência do SGBD adotado.

Do ponto de vista do SGBD empregado, o SPRING v2.0 oferece uma interface seguindo o padrão SQL, além de interfaces com o CODEBASE e com o POSTGRES. O acesso ao banco de dados é encapsulado em uma classe específica.

## 11.2 O Modelo de Dados do SPRING

### 11.2.1 Nível Conceitual

O modelo de dados do SPRING v2.0 baseia-se em uma versão preliminar do modelo descrito no capítulo 4. A Figura 11.1 resume as classes básicas, a nível conceitual, do modelo.

As classes GEOCAMPO, TEMÁTICO, NUMÉRICO, DADOSENSORREMOTO, GEOOBJETO e OBJETO NÃO ESPACIAL são equivalentes às classes homônimas introduzidas no capítulo 4.

A classe CADASTRAL foi introduzida para permitir a associação de representações múltiplas de um geo-objeto e armazenar as relações de topologia entre os geo-objetos representados num mesmo mapa. As instâncias desta classe definem um mapeamento de geo-objetos para uma determinada região geográfica e projeção cartográfica.

A classe PLANOINFORMAÇÃO generaliza as classes GEOCAMPO e CADASTRAL.

Para ilustrar o caso em que mapas cadastrais representam a localização de geo-objetos, considere o exemplo de um mapa de lotes de uma cidade com duas classes: LOTES, especialização de GEOOBJETO, e MAPALOTES, especialização de CADASTRAL. No caso dos lotes, a informa-

geoid	num_cad	área	endereço	IPTU (R\$)
154	195689	1500	Serimbura, 60	350

Figura 11.2: Exemplo de Atributos para LOTES

geoid	número	região	escala	ano
345	273	Jardim Esplanada	2000	1986

Figura 11.3: Exemplo de Atributos para MAPALOTES

ção associada seria o número do cadastro na Prefeitura, endereço, área construída e valor do IPTU, conforme a figura 11.2. Para os elementos da classe MAPALOTES, os atributos típicos seriam as características do mapa – número do mapa, região da cidade, escala do mapa, data do levantamento – como ilustrado na figura 11.3. Por fim, para estabelecer a ligação entre os geo-objetos e suas representações (no exemplo, entre lotes e suas localizações num mapa de lotes) é preciso associar os polígonos (ou arcos) do mapa cadastral aos geo-objetos descritos anteriormente, como mostra a figura 11.4.

Recorde que o modelo de dados contempla ainda a classe OBJETONÃOESPACIAL, que engloba qualquer tipo de informação que não seja georeferenciada, e que se queira agregar a um SIG. O exemplo, na figura 11.5 mostra o uso deste conceito em uma aplicação de cadastro rural. Neste caso, tem-se os geo-objetos da classe FAZENDAS (que estão localizados num mapa) e se deseja estabelecer a ligação entre estes geo-objetos e a informação alfanumérica constante em um cadastro de propriedades já existente. Estas informações de cadastro são consideradas objetos não espaciais.

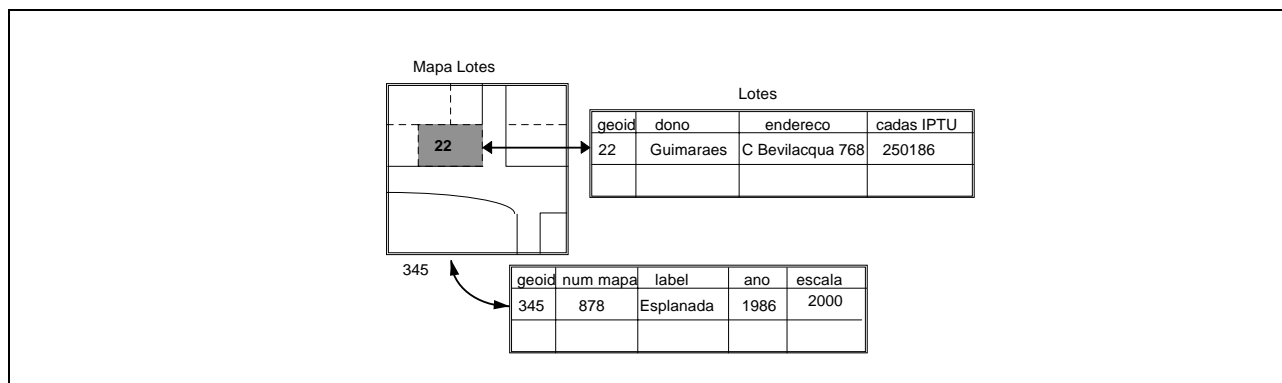


Figura 11.4: Associação de geo-objeto a mapa cadastral



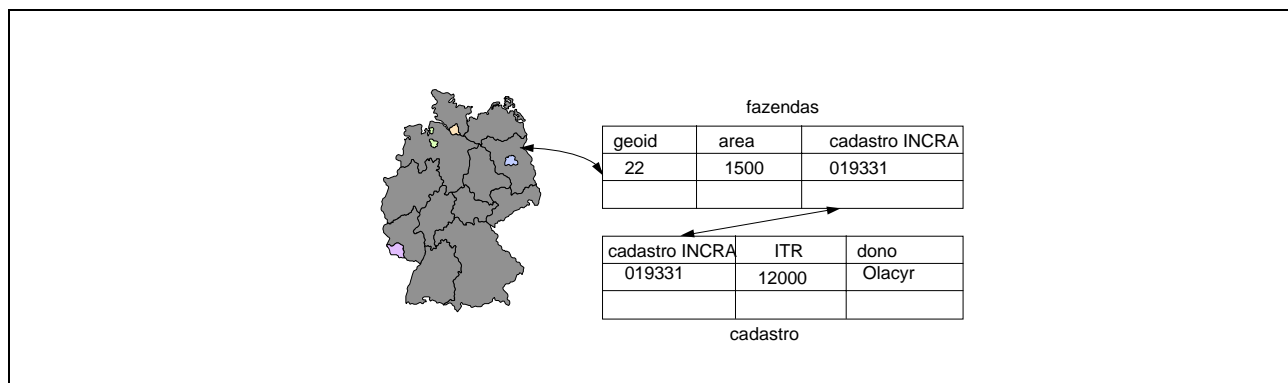


Figura 11.5: Ligação entre geo-objeto e objeto não espacial

### 11.2.2 Nível de Representação

No nível de representação, o SPRING possui duas grandes classes, REPRESENTAÇÃO VETORIAL e REPRESENTAÇÃO MATRICIAL. O SPRING permite que tanto as representações matriciais quanto as vetoriais estejam associadas a um mesmo plano de informação.

O sistema distingue as seguintes especializações da classe REPRESENTAÇÃO VETORIAL:

- SUBDIVISÃO PLANAR: conjunto de arcos, nós e polígonos formando um recobrimento completo (utilizado para mapas temáticos no formato vetorial);
- GRAFO ORIENTADO: conjunto de arcos e nós conectados sob forma de grafo orientado;
- ISOLINHAS: conjunto de linhas, que não se cruzam, associadas a uma cota (utilizado para modelos numéricos de terreno);
- AMOSTRAS 3D: conjunto de pontos esparsos, armazenando coordenadas 3D (utilizado em modelos numéricos de terreno);
- GRADE TRIANGULAR: representação vetorial formada por triângulos, cujos nós possuem coordenadas 3D.

As especializações da classe REPRESENTAÇÃO MATRICIAL são:

- GRADE REGULAR 2D: uma matriz onde cada elemento está associado a um valor real;
- IMAGEM TONS CINZA: imagem representada através de uma matriz cujos valores correspondem à quantização de uma grandeza numérica. Esta representação é utilizada tanto para dados de sensores remotos como para apresentações pictóricas de modelos numéricos de terreno;
- IMAGEM TEMÁTICA: representação matricial de um geo-campo da classe TEMÁTICO, dada por uma matriz onde cada elemento é o índice em uma tabela que contém a descrição dos temas do mapa.

A tabela 11.1 mostra como um mesmo dado geográfico pode ter diferentes representações.

Dado Geográfico	Nível Conceitual	Nível de Representação
Mapa de aptidão agrícola	Geo-campo Temático	Matriz temática Representação planar (polígonos)
Mapa altimétrico	Geo-campo numérico	Grade regular 2D Grade triangular Amostras 3D
Lote urbano	Geo-objeto	(em um mapa de lotes)
Mapa de lotes	Cadastral	Representação vetorial (arco-nó-polígono)
Rede elétrica	Rede	Grafo orientado (arcos orientados + nós)

Tabela 11.1: Relação entre níveis conceitual e de representação

## 11.3 Interface Interativa e Linguagem de Consulta

O SPRING v2.0 oferece uma interface de usuário dirigida por menus, especialmente útil à definição de esquemas conceituais.

O processo de definir o esquema conceitual de um banco de dados geográfico no ambiente SPRING consiste em estender a hierarquia de classes, definida pelo modelo de dados, criando classes derivadas das classes básicas. Na terminologia adotada no SPRING, uma especialização de uma geo-classe é chamada de *categoria*, podendo ser esta uma classe básica ou não. Ao definir uma categoria, o usuário estará criando uma classe derivada das classes básicas, herdando as suas propriedades.

Como exemplo, considere a seguinte definição do esquema conceitual de um banco de dados geográficos para cadastro rural, ilustrada na figura 11.6. A categoria FAZENDAS é uma especialização da classe GEOOBJETO. A categoria MAPAPROPRIEDADES é uma especialização da classe CADASTRAL, que inclui uma representação geométrica para os geo-objetos da classe FAZENDAS. A categoria MAPASOLOS é uma especialização da classe TEMÁTICO, cujas instâncias armazenam os tipos de solos para as áreas de estudo. As categorias ALTIMETRIA e DECLIVIDADE são especializações da classe NUMÉRICO, cujas instâncias guardam a topografia e a declividade da área de estudo, respectivamente. A categoria DADOSTM é uma especialização da classe DADOSSENSORREMOTO, cujas instâncias contêm imagens do satélite Landsat sobre a região de estudo.

O SPRING v2.0 inclui uma implementação parcial da linguagem LEGAL, já discutida na seção

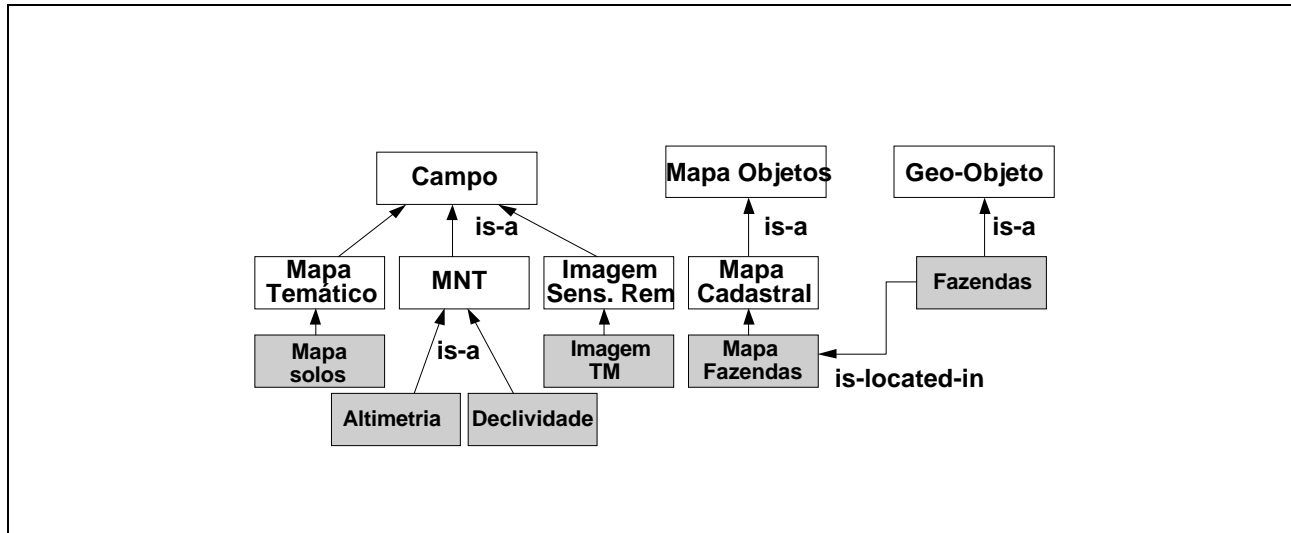


Figura 11.6: Exemplo de definição de esquema conceitual no spring

6.3. Recordando, **LEGAL** baseia-se em uma álgebra geográfica, que permite expressar operações específicas para cada tipo de dado, bem como explicitar operações que transformam um tipo de dado em outro.

Para indicar como **LEGAL** deverá operar em sua implementação plena, considere o exemplo de um estudo para selecionar uma área para depósito de lixo na cidade de São José dos Campos, baseando-se na localização dos postos de saúde e em mapas de solos e de declividade. As condições ideais de localização de um depósito de lixo são: estar a mais de 5 km de postos de saúde, em solo não-permeável e com declividade menor que 5%. As áreas com valor aceitável são marcadas “adequadas para um depósito de lixo”. A operação requer uma consulta espacial (“selecione todos os postos de saúde localizados na região da Vila Paiva”), uma transformação entre geo-objetos e geo-campo (“calcule um mapa de distâncias a partir dos postos de saúde”), uma fórmula numérica (“determine a declividade”) e uma análise booleana. A figura 11.7 apresenta os comandos em **LEGAL** que implementam o exemplo descrito.

## 11.4 Funcionalidade

A funcionalidade completa do **SPRING v2.0** não será aqui descrita em detalhes, sendo abordadas apenas algumas operações e facilidades oferecidas pelo sistema, ressaltando aquelas incorporadas nesta nova versão. Para uma descrição mais detalhada sobre as funções citadas, o leitor deve se referir ao manual do sistema.

**Funções sobre Imagens.** O **SPRING** oferece um amplo leque de operações sobre imagens, incluindo operações de contraste, filtragem, classificação, registro, segmentação, operações aritméticas, transformação IHS-RGB, operações estatísticas, entre outras. A classificação de imagens por regiões oferece funções de classificação não supervisionada [BEF93], su-

```

// Definicao dos Dados
Tematico solo (SOLO), adeq (ADEQUACAO), dist (DISTANCIA);
Cadastral  mapa_SJC (MAPA_CIDADE);
Numerico   topo(TOPOGRAFIA), decl (DECLIVIDADE);
Tabela     tab_dist (FATIAMENTO);
Colecao    postos (PSAUDE);

// Recuperacao dos Dados
topo=(SELECT * FROM TOPOGRAFIA WHERE Nome = 'Top92');
solo=(SELECT * FROM SOLO WHERE Nome = 'Solo92');
mapa_SJC=(SELECT * FROM MAPA_CIDADE WHERE Nome = 'SJC Campos');

// Postos de Saude no Bairro
postos = (SELECT posto
          FROM posto IN PSAUDE ON MAP mapa_SJC,
          bairro IN BAIRRO ON MAP mapa_SJC,
          WHERE posto INSIDE bairro.nome = 'Vila Paiva');

// Calculo do Mapa de Distancias
mapa_dist= NOVO ( Nome = 'Dist', Representacao = Matricial,
                 ResX = 100, ResY= 100, Escala = 25000);
tab_dist:= NOVO (TIPO = FATIAMENTO,
                 'mais de 5 km' : $>$ 5km, 'perto': DEFAULT);
mapa_dist= FATIA (DISTANCIA (postos, tab_fatia));

// Calculo do mapa de declividade
decl= NOVO (Nome='Declividade', Representacao = Matricial,
           ResX = 100, ResY= 100);
decl := DECLIVIDADE (mapa_topo);

// Calculo do Mapa de Adequacao
adeq= NOVO (Nome = 'DepositoLixo', Representacao = Matricial,
           ResX = 100, ResY= 100, Escala = 25000);
mapa_adeq: = Atribua ( 'adequado': solo.tema = 'Le'
                      E dist.tema='mais de 5 km'
                      E mapa_decl $<$ 0.05;
                      'inadequado': Outros) ;

```

Figura 11.7: Exemplo de implementação em LEGAL

pervisionada e por redes neurais [BML94]. Possui operações para restauração de imagens TM e SPOT. Na versão 2.0, estão sendo incorporadas diversas funções para processamento de imagens de radar, inclusive: correção de padrão de antena, correção geométrica (“slant range – ground range”) e cálculo de parâmetros estatísticos.

**Funções sobre Dados Temáticos e Cadastrais.** As funções sobre dados temáticos e cadastrais incluem operações de tabulação cruzada, lineamentos, mapas de distância, edição com topologia automática (ajuste e poligonalização), operações booleanas (álgebra), mosaico e mascaramento, transformação vetor-varredura e varredura-vetor, entre outras.

**Funções sobre Modelos Numéricos de Terreno.** Oferece operações de cálculo de declividade, fatiamento, geração de imagem, perfil, cálculo de volume, visualização 3D, ponderação (álgebra), mosaico e mascaramento, geração de grade regular e triangular, geração de isolinhas, entre outras. Ressaltamos a operação de triangulação com restrições, onde as linhas de restrição representam linhas ao longo das quais ocorrem quebras de continuidade na superfície modelada. Um modelo de terreno deve incorporar estas linhas se uma representação fiel é necessária. Assim, para o modelo que utiliza grades triangulares, as linhas de restrição devem ser incorporadas e a superfície a ser ajustada a cada retalho triangular deve também considerar a quebra de continuidade.

**Funções sobre Redes.** A versão 2.0 oferecerá operações de caminho ótimo e centros de atendimento (alocação).

**Acoplamento a Banco de Dados.** O SPRING permite o acoplamento a um SGBD relacional mono-usuário e também a SGBDS comerciais com suporte à arquitetura cliente-servidor. Oferece interface compatível com DBASEIV e com SQL. As operações de consulta ao banco de dados incluem seleção de objetos baseada em atributos ou em restrições topológicas (adjacência, cruzamento, pertinência, sobreposição) e métricas (distância).

**Geração Interativa de Cartas.** o SPRING dispõe de um módulo para geração de cartas, que combina uma interface interativa com o acesso a uma biblioteca de símbolos cartográficos.

**Interface com outros SIG.** A fim de garantir a comunicação com SIG mais utilizados no mercado, o sistema importa dados nos formatos SGI, MAXICAD, ASCII, ARC/INFO e DXF. A versão 2.0 exporta dados nos formatos SGI, ASCII, ARC/INFO e DXF.

Além da funcionalidade delineada acima, cabe ressaltar que o SPRING v2.0 permite a execução de forma não interativa de certas operações, tais como importação com remapeamento de imagens, segmentação, classificação e modelagens.

# Capítulo 12

## O Projeto SAGRE

### 12.1 Introdução

O SAGRE (Sistema Automatizado de Gerência de Rede Externa) [Mag94] é um conjunto integrado de software para automatizar os diversos processos relacionados ao cadastro, planejamento, projeto, implantação, operação, manutenção, expansão e gerência da Rede Externa das Empresas Operadoras do Sistema TELEBRÁS.

A Rede Externa é a rede que está do lado de fora das Estações Telefônicas. Ela é composta da rede de canalização (rede formada por conjunto de dutos enterrados conectados a caixas subterrâneas), da rede aérea (rede dos cabos suspensos) e da rede subterrânea (rede dos cabos que passam pela rede de canalização).

O SAGRE é um típico sistema de AM/FM construído sobre um SIG. O SIG é utilizado para representar espacialmente a rede de canalização e a rede aérea sobre um mapeamento urbano. A rede subterrânea é representada em esquemáticos (ou detalhes) – *layouts*. Aplicações deste tipo podem trazer uma série de benefícios para as empresas que os utilizam.

O SAGRE permitirá alcançar, dentre outros benefícios, a redução do tempo de implantação das redes telefônicas, a melhor utilização da rede instalada e a melhoria da qualidade dos serviços. O SAGRE está sendo especificado e desenvolvido de forma cooperativa por Empresas Operadoras do Sistema TELEBRÁS e pelo Centro de Pesquisa e Desenvolvimento da TELEBRÁS (CPQD-TELEBRÁS).

### 12.2 Funcionalidade

O início do SAGRE foi motivado pela necessidade de otimizar os processos de administração das plantas cadastrais do sistema de telecomunicações – atualmente, cerca de 300.000 plantas no total. Este objetivo motivou a criação de um grupo de trabalho multidisciplinar, com a participação de especialistas na área de Rede Externa (planejamento, projeto, implantação e operação) e de Computação provenientes da TELESP, TELERJ, TELEMIG, TELEPAR,

TELEBAHIA, TELEBRASÍLIA e TELEAMAZON para a elaboração do documento preliminar da especificação do SAGRE.

Requisitos do SAGRE exigiram uma plataforma computacional que implementa uma arquitetura cliente-servidor. No lado servidor são utilizadas estações de trabalho RISC de alto desempenho e resolução gráfica e sistema operacional UNIX. Os clientes podem ser estações UNIX ou micro-computadores com sistema operacional MS/Windows. O ambiente computacional é completado com impressoras laser e *plotters*. Os dados são processados através de um SIG comercial. Este SIG difere da maioria dos sistemas geográficos comerciais por utilizar um SGBD relacional comercial para gerenciar não só os dados convencionais mas também os dados espaciais.

O SAGRE está sendo desenvolvido a partir de uma filosofia de sistemas abertos, com facilidade de portabilidade (linguagem de programação C e linguagem de manipulação de banco de dados SQL), adoção de metodologia estruturada e padrão de interface homem-máquina OSF/Motif.

As funções a serem atendidas pelo SAGRE envolvem o cadastramento da rede e mapeamento urbano (correspondente à inserção, no sistema, dos dados e seu georeferenciamento), planejamento, estudos de mercado e demanda, projeto de engenharia e implantação e gerenciamento da operação e manutenção da rede. Suas principais funções estão concentradas em

- Cadastro – conjunto de procedimentos para cadastramento e atualização da rede existente e mapeamento urbano [GEOG94].
- Conversão – metodologia e ferramentas para conversão dos dados e mapas do papel para o banco de dados do SAGRE [MGS<sup>+</sup>94].
- Administração – ambiente de controle de áreas em projeto ou em implantação, controle de usuários, configuração de índices de orçamentação, e gerência do fluxo de trabalho. Esta função permite o trabalho cooperativo entre operadores e projetistas.
- Projeto e Planejamento – funções relativas à elaboração do planejamento para modernização e expansão da Rede Externa [PTO95]. Estas atividades incluem o planejamento de novos centro de fios, cortes de área, entroncamentos e atendimento de grandes clientes, definição da área de tarifa básica, projetos das redes de canalização, alimentação, distribuição e entroncamentos, com previsão de materiais e custo de mão-de-obra envolvidos nos projetos.
- Operação – alocação, reserva, liberação e dedicação de facilidades para serviços de clientes.

Além de reduzir o trabalho manual de verificação da qualidade dos dados, dando mais opções para a atividade de planejamento, o SAGRE permitirá a unificação e integração dos cadastros e gerenciamento automatizado dos recursos existentes, com conseqüente economia de custos e utilização racional da rede. Experiências em outros países mostram que a automatização dos processos de gerenciamento de Rede Externa reduziram os custos das operações de cadastro

em 90%, 20% na elaboração de projetos e 40% nas funções de planejamento, o que resultaria numa redução de custos na média em torno de 50%.

## 12.3 Componentes e Arquitetura

### 12.3.1 Principais Componentes

Geralmente, a rede externa é dividida em rede aérea, rede de canalização e rede subterrânea. Da estação telefônicas, partem as conexões (pares de fios) dentro das canalizações diretamente para concentrações de usuários (e. g. prédios) ou para um centro de distribuição (armário). Do centro de distribuição, os pares são levados por cabos suspensos (às vezes enterrados) até os usuários.

O sistema está organizado em diversos módulos. O módulo de cadastro, já instalado na maioria das operadoras, fornece ferramentas para manutenção, recuperação, geração de relatórios, mapas e conversão de dados de mapas em papel para o meio digital.

No módulo de projeto do ambiente do SAGRE, o usuário (projetista) trabalha nas etapas de criação, modificação e acompanhamento de um projeto. Para iniciar um novo projeto, o usuário precisa definir a região alvo, que está armazenada no banco de dados relacional do SAGRE. Esta escolha de região pode ser feita interativamente, através da seleção de partes de mapas mostrados na tela. Quando a região é selecionada, seus dados são copiados para um banco de dados de trabalho (a chamada atividade de “check-in”), que será utilizado pelo projetista para desenvolver alternativas de projeto. No cadastro, os elementos copiados são marcados como bloqueados com compartilhamento. Ao fim da atividade de projeto, o conteúdo do banco de dados de trabalho será retornado (“check-out”) para o banco de dados do SAGRE, ocorrendo então verificação da consistência. Este processo é controlado por um sistema de versões, descrito ao fim deste capítulo.

Outras funções permitem realizar cálculos de engenharia e de custo de material e mão-de-obra. Atualmente, esta parte é realizada manualmente, sendo responsável por uma grande parcela dos custos de um projeto. Os cálculos de engenharia envolvem operações como atenuação do sinal elétrico, definição da localização de aterramentos, testes de válvulas. O cálculo de custos consome atualmente cerca de 40% do tempo de um projeto e a sua automatização pelo SAGRE deverá reduzir consideravelmente este custo.

Outra função corresponde às operações de análise de redes, que permite calcular e prever fatores como ocupação dos dutos e cabos. Os valores máximo e mínimo de ocupação e as características de uma área (demanda, localização) permitem definir a expansão da rede. Este tipo de atividade pode envolver todas os elementos de uma rede em uma região, ou apenas um elemento específico.

As facilidades de manipulação dos dados incluem edição de atributos convencionais, espaciais e gráficos. A atividade de projeto envolve tanto os elementos da rede quanto os esquemáticos e “layouts” de circuitos. Cada esquemático é associado a uma localização geográfica ou a



outro esquemático mais complexo. Um dos problemas atualmente sendo tratado é a geração automática de esquemáticos a partir de operações do usuário.

### 12.3.2 Arquitetura do SAGRE

A arquitetura do SAGRE pode ser considerada como composta de três camadas funcionais, que executam tarefas bem definidas e funcionalmente independentes: Interface Homem-Máquina (IHM), Aplicação (APL), que permite a transformação dos dados e Encapsulamento da Base de Dados (EBD), conforme ilustra a figura 12.1.

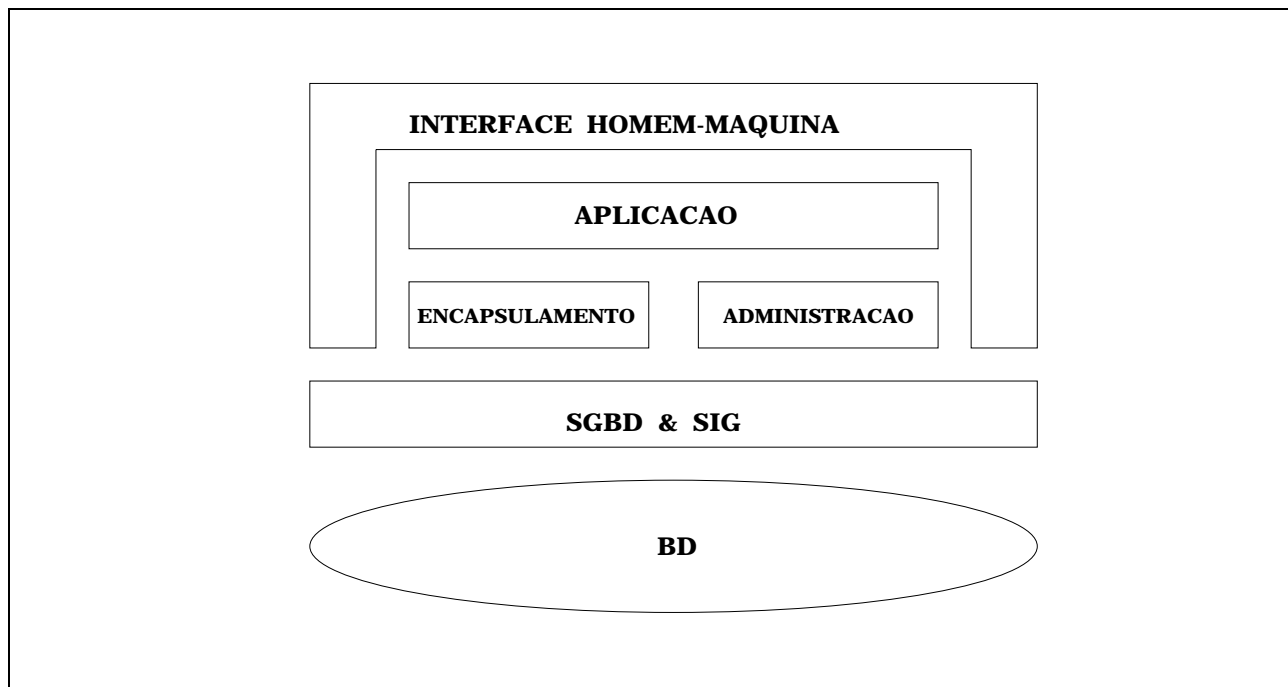


Figura 12.1: Arquitetura do SAGRE

A camada IHM é responsável pela implementação de todos os processos de comunicação com o usuário, incumbindo-se de padronizar a interface. Seus serviços são implementados através de primitivas de abertura/fechamento de telas de atributos, mudança de características dos atributos de uma tela, entrada de informações, confirmação e escolha de opções. A interface é baseada em um conjunto de telas (mapas básicos, definição de atributos, esquemáticos e mensagens). As telas de mapas básicos contêm os detalhes da rede externa, mostrando seus elementos tanto graficamente quanto textualmente. O usuário pode realizar várias operações para visualização, como “zoom”, “pan” e outras.

A camada APL implementa todas as operações de edição (inserção, alteração e remoção) e consulta sobre os objetos do banco de dados, centralizando as tarefas da aplicação propriamente dita. Esta camada é ativada quando o usuário seleciona um objeto através da interface e escolhe uma operação a ser executada. Ela é responsável pelos procedimentos operacionais do sistema,

tais como garantir a consistência dos dados a partir das regras de engenharia e de gerenciamento. A camada é orientada a objetos. Cada objeto manipulado é um elemento da rede, associado a um símbolo para sua representação gráfica.

O módulo EBD reúne todas as rotinas de acesso ao banco de dados, sendo responsável pela padronização de acesso, abstração em relação às tabelas físicas do sistema e segurança na interação dos módulos de IHM e APL com a base de dados, o que se dá através da invocação de primitivas funcionais pré-definidas. Esta camada é responsável pelo banco de dados permanente e pelos bancos de dados de trabalho (para projeto de alternativas), com o controle de versões. Mais importante, esta camada tem uma função básica de prover uma abstração orientados a objetos sobre o sistema relacional utilizado. Dessa forma, a camada APL trabalha com objetos encapsulados, deixando para a camada EBD a tarefa de mapear estes objetos (e, às vezes, seus componentes) nas tabelas do esquema relacional.

A comunicação entre as camadas é gerenciada pelo módulo de administração, que controla todos os projetos em andamento para cada área geográfica coberta pelo banco de dados. Os mecanismos de controle de versão [DGM95] são implementados aqui.

O primeiro módulo a ser automatizado foi o de manutenção do cadastro de rede externa, denominado SAGRECad, cujo objetivo é a atualização de dados, desde o nível macro (rede) até o nível de esquemáticos, através de funções de edição de objetos, edição de esquemáticos e associação de imagens *raster*.

## 12.4 Elementos Armazenados e Modelo de Dados

Um conceito fundamental na modelagem de sistemas para aplicações de concessionárias de serviços públicos é o MUB – *Mapa Urbano Básico*. Este modela os elementos que formam a base do planejamento urbano. Em princípio, todas aplicações relativas a serviços utilitários (telefonia, eletricidade e outros) usam o MUB como base, porém os elementos utilizados variam conforme a aplicação. Exemplos de elementos que fazem parte do MUB são divisas de lote, linhas centrais, imóveis (numeração predial), obras de arte e edificações de destaque.

No caso do SAGRE, os elementos do MUB incluem: Arruamento (Quadra, Meio Fio, Logradouro), Loteamento (Testada de Lote, Indicação de Lote, Numeração), Aspectos Geográficos (Hidrografia, Acidentes Geográficos), Obras Públicas (Rodovia, Ferrovia, Edificações de Destaque). Os elementos do MUB são armazenados no banco de dados (camada EBD) fazendo parte do cadastro armazenado. Do ponto de vista do nosso modelo, descrito no capítulo 4, estes elementos constituem geo-objetos.

No MUB do SAGRE, um logradouro é representado por um ou vários segmentos de *linha central*, conceito fundamental para permitir desenvolver as funções relativas ao georeferenciamento. A linha central é, na verdade, um conjunto de segmentos de reta que passam ao longo do eixo dos logradouros e são delimitadas por pontos de interseção (cruzamento de ruas ou esquinas) com outros logradouros.

Nos pontos de interseção, os segmentos de linha central coincidem nas extremidades formando uma rede de conectividade física.

Além dos elementos do MUB o banco de dados armazena vários outros componentes, relativos aos dados da rede externa. Exemplos são dados da *rede aérea* (bloqueio, caixas terminais, lances de cabo, subida de lateral, caixas prediais, emendas, armários, aterramento), da *canalização subterrânea* (lance de duto, duto lateral, cano interno, amarração das caixas) e da *rede subterrânea* (caixa subterrânea, lance de cabo subterrâneo, subida de lateral). Estes dados sofrem o chamado processo de *conversão* para serem armazenados no banco de dados, de acordo com um padrão patentado pelo CPQD-TELEBRAS, discutido ao fim deste capítulo.

No banco de dados do SAGRE existe um conjunto de estruturas topológicas de redes, que são fundamentais à representação da rede externa, permitindo a modelagem das várias redes existentes. Estas estruturas garantem que se possa considerar a geometria e topologia das redes que fazem parte da rede externa (por exemplo, rede de cabos e rede de canalização).

No processo de inserção, os dados da rede externa são georeferenciados usando como base tais estruturas e posicionados em relação aos dados do MUB previamente armazenados. Por exemplo, na inserção de um lance de cabo, os postes que fazem parte do percurso do lance devem ser registrados no banco de dados e relacionados ao lance. Este relacionamento se faz através das coordenadas de identificação, que indicam os elementos relacionados através de uma coordenada próxima ao elemento, dentro de uma tolerância.

As caixas subterrâneas são posicionadas sobre o MUB e com lances de duto interligando-as. Para cada caixa subterrânea deve-se associar um sub-mapa, denominado Esquemático de Cabos e Emendas, que descreve sob a forma de diagramas as ligações dos lances de cabos subterrâneos, emendas e demais equipamentos que estão localizados na caixa subterrânea.

Os dados armazenados são extraídos das plantas da rede utilizadas nas empresas operadoras. Estas plantas são visualizadas no SAGRE através de elementos gráficos que estão dispostos em diferentes níveis. Cada elemento representa graficamente um item de rede externa (poste, cabo) e pode possuir dados alfanuméricos associados (tipo, capacidade, comprimento) que estão contidos em relações do banco de dados. As informações estão organizadas por objetos que representam os elementos reais existentes no MUB e na Rede Externa. Além dos dados alfanuméricos, cada elemento possui uma representação gráfica que o caracteriza. Um exemplo de uma tela com objetos de MUB e rede é mostrado na figura 12.2.

## 12.5 Características da Manipulação de Dados

O projeto e desenvolvimento do SAGRE segue uma metodologia orientada a objetos. Assim sendo, cada elemento do cadastro da rede externa está associado a um conjunto de operações que variam segundo suas características particulares, visando a manutenção do cadastro. Cada uma dessas operações é habilitada e desabilitada automaticamente em função da situação contextual do momento. Essas operações podem ser agrupadas em três classes funcionais básicas: Inserção, Modificação e Remoção, que atuam tanto sobre dados espaciais quanto textuais e de

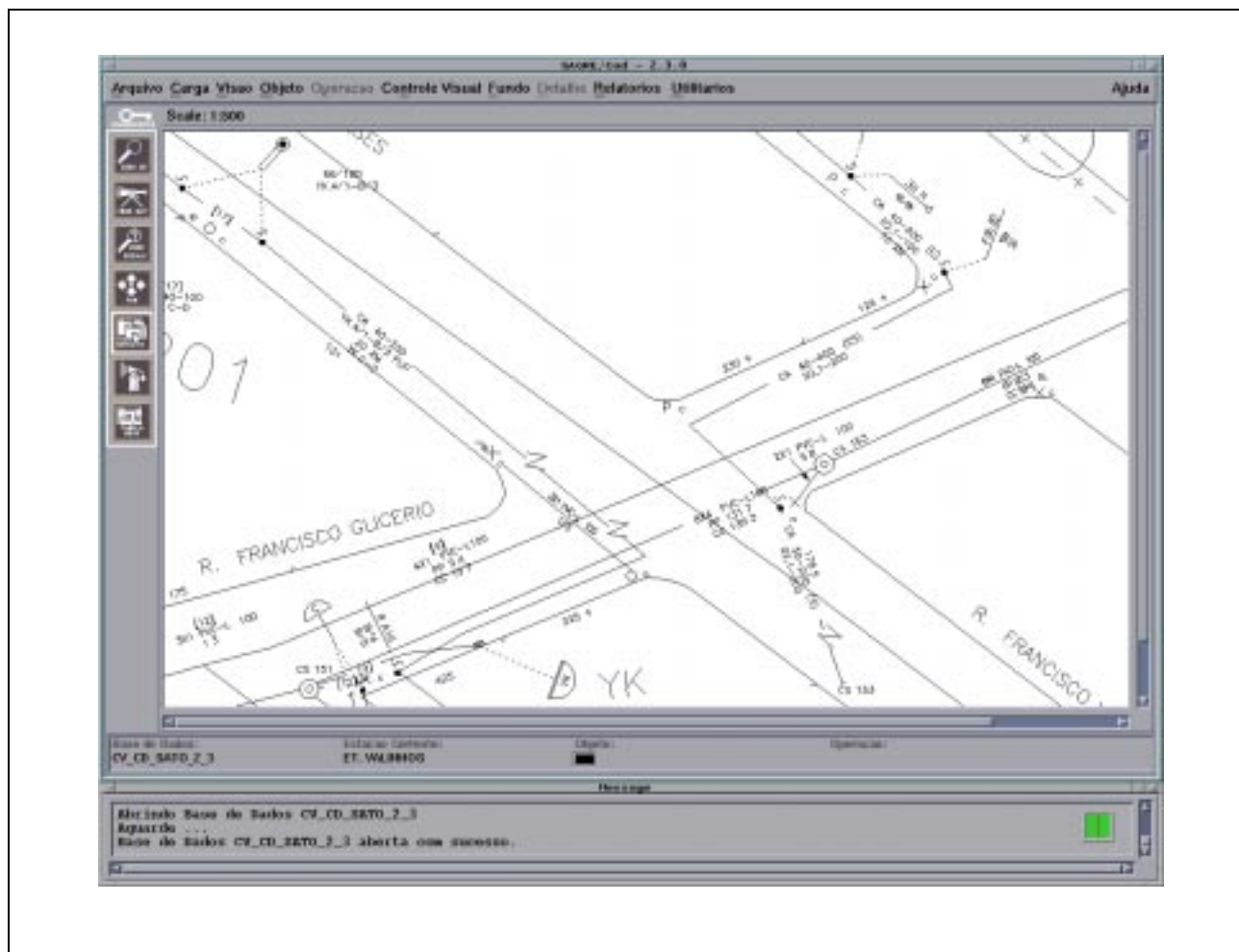


Figura 12.2: Exemplo de tela base do SAGRE

apresentação.

Por exemplo, durante a inserção, o usuário tem a possibilidade de desenhar um objeto gráfico indicando graficamente o ponto inicial, os pontos em que o mesmo muda de direção e o último ponto, ou inserir um símbolo pré-definido que o represente, indicando o ponto onde deve ser inserido e sua orientação. Entretanto, antes de representá-lo graficamente, pode ser necessário indicar os pontos de interesse da rede ou MUB e/ou outros objetos aos quais está associado. Em seguida, o ambiente auxilia o usuário na definição dos atributos do objeto. Um objeto pode ser apresentado associado a diversos fatores de simbologia: cor, legendas e símbolos especiais. A associação de um objeto às legendas pode ser automática ou manual. Neste último caso, o usuário indica a posição (localização e orientação) onde a legenda deve aparecer e digita o texto desejado.

Além do gerenciamento dos elementos da rede externa, o sistema auxilia o gerenciamento automatizado dos componentes de esquemáticos, que podem ser editados e manipulados diretamente através da camada IHM. O usuário pode, inclusive, associar uma imagem *raster* a

qualquer objeto ou a qualquer ponto da região geográfica que está sendo manipulada. A imagem é associada a duas coordenadas da região que está sendo manipulada, sendo dimensionada conforme o tamanho da área determinada pelas coordenadas fornecidas.

Com estas facilidades, o usuário pode cadastrar informações que não são próprias da rede externa, mas que influenciam o seu traçado, como é o caso da rede de esgotos e da rede de energia elétrica.

Em geral, as concessionárias de serviços públicos possuem sistemas de operação desenvolvidos em computadores de grande porte sem o auxílio de um SIG. O uso de SIG nestas aplicações traz uma série de benefícios, em especial na localização espacial dos consumidores e dos equipamentos que podem servi-los.

No ambiente da operação da rede, as transações são curtas mas freqüentes. Normalmente estas transações são originadas no sistema de atendimento ao cliente das operadoras. Por exemplo, uma transação comum é o provimento de um serviço num determinado endereço. Neste caso é fornecido o endereço do cliente e o serviço desejado. De posse do endereço, uma pesquisa convencional é realizada para encontrar a instância. Uma vez encontrado o endereço, sua localização geográfica é dada pelo SIG. Neste ponto é feita uma busca espacial para encontrar os elementos de rede que podem se conectar ao endereço do cliente a uma certa distância deste. Conhecidos os elementos de rede candidatos, percorre-se a rede avaliando-se condições de forma a determinar a solução ótima para atender o cliente. Esta transação também poderia ser disparada por um engenheiro a partir de um micro-computador remoto. Com estes requisitos, o SAGRE teve que utilizar geradores de interfaces capazes de gerar código para várias plataformas, especialmente PC/MS/Windows e Risc/UNIX.

No ambiente de projeto e planejamento a situação é diferente. Neste caso a interação homem-máquina é longa e complexa. Durante um projeto, elementos são inseridos com atributos parcialmente conhecidos. À medida em que o projeto vai progredindo, os atributos deixados incompletos vão sendo preenchidos até que se tenha um projeto concluído. Foi necessário criar mecanismos de verificações de pendências. Também, num ambiente deste tipo é necessário fornecer meios de retroceder no projeto. O uso de versões foi fundamental no provimento desta funcionalidade.

## 12.6 Interface

A interface do SAGRE [OCM95] é dinâmica, ou seja, os componentes de uma tela são definidos em tempo de execução, à medida em que o usuário interage com o banco de dados. Ela deve ser considerada sob dois aspectos: o modelo da interface propriamente dito (arquitetura da interface) e a especificação de uma biblioteca de módulos que permitem criar e combinar os elementos de uma apresentação (funcionalidades disponíveis). Ambos o modelo e a biblioteca de classes seguem o paradigma de orientação a objetos. A especificação do modelo seguiu o modelo dinâmico da metodologia OMT.

A interface foi construída visando a reutilização de esforços, em termos de projeto e de

implementação. O modelo de interface é orientado a objetos, sendo utilizado para definir uma biblioteca de elementos que permitem manipular representações gráficas. O sistema de interface permite reutilização na construção de interfaces dinâmicas, isto é, cujos componentes são determinados em tempo de execução. A capacidade de construção dinâmica dos elementos da interface possibilita a personalização da interface, o que aumenta a flexibilidade do sistema e conseqüentemente o grau de satisfação do usuário.

Pode-se programar, para cada elemento gráfico, os atributos alfanuméricos que devem aparecer no mapa, bem como a forma que os mesmos devem ser exibidos. Além disso, o próprio símbolo do elemento pode ser alterado automaticamente em função da alteração dos seus atributos.

Os componentes de diálogo não precisam ser definidos estaticamente durante o processo de compilação da interface; ao contrário, eles podem ser inseridos e removidos dinamicamente.

O modelo de interface adota a linguagem de definição de interface orientada a objetos CORBA IDL definida na arquitetura de gerenciamento de objetos proposta pelo *Object Management Group* (OMG) [Obj94], garantindo o suporte semântico necessário, que dispõe inclusive de mecanismos para descrição de métodos e passagem de mensagens, e assegurando a padronização entre os diversos projetos de interface, no que se refere ao diálogo com a aplicação.

A principal motivação para a construção do modelo de interface foi promover a reutilização não só de código, mas também de projetos de interface. O modelo é formado por um conjunto de classes inter-relacionadas, onde cada classe descreve um componente típico de um sistema de *interface visual*. Uma interface visual suporta e permite a coexistência de qualquer um dos estilos básicos de interação (linguagens, menus, telas formatadas e manipulação direta). Esta classe de interfaces é, portanto, mais abrangente que a das interfaces gráficas, as quais utilizam primordialmente conceitos de manipulação direta.

O principal elemento do modelo é a classe **Janela**. Qualquer interface visual baseia sua interação com o usuário em algum tipo de janela, que pode ser ou não gráfica, e que contém os elementos de diálogo usados na interação. Uma janela é formada por um conjunto de painéis, onde cada painel agrega, por sua vez, elementos de interface relacionados de acordo com as idéias do projetista da interface. Esta característica de definição recursiva de um painel de controle facilita a reutilização de componentes na interface.

A janela que contém os componentes de interação (painéis de controle), pode estar associada a outras janelas. Existem diferentes tipos de relacionamentos entre janelas, como por exemplo a relação de hierarquia entre janela básica e janelas secundárias, em um ambiente de janelas múltiplas, ou como a relação de ordem de chamada de janelas, em um sistema de telas formatadas.

O classe **Janela** é também relacionada à classe **Interface da Aplicação**, que representa a visão que os componentes de interface possuem da aplicação subjacente. A classe **Interface da Aplicação** pode ser considerada como um elemento de ligação único que associa os componentes da interface com o usuário aos componentes da aplicação propriamente dita. É este conceito de visão que possibilita a utilização do modelo para construção de interfaces dinâmicas.

O objetivo deste relacionamento é prover *independência de diálogo* às interfaces baseadas na biblioteca, garantindo que decisões de projeto que afetam apenas o diálogo homem-computador sejam isoladas daquelas que afetam apenas o software computacional e a estrutura da aplicação. Esta abordagem simplifica as modificações tanto da interface quanto da aplicação, pelo isolamento existente entre os dois componentes do sistema.

Outras classes da interface são **Campo Textual**, **Área de Desenho**, **Botão**, **Lista** e **Menu**.

A metodologia utilizada para o desenvolvimento da biblioteca da interface foi a *Object-Oriented Modeling and Design* (OMT) [Rum91], que mostrou-se bastante eficiente tanto nas fases iniciais quanto na implementação do projeto. Ela possibilitou manter a visão orientada a objetos, mesmo utilizando uma linguagem convencional (no caso, a linguagem C). A biblioteca é extensível, ou seja, novos componentes podem ser acrescentados ou modificados via mecanismo de herança, para reutilização. Este processo de reutilização exige que sejam oferecidas facilidades para pesquisa, recuperação e atualização de componentes da biblioteca.

Um dos componentes da camada de interface com o usuário é o módulo denominado Gerente de Atributos (GAT), que implementa o acesso aos atributos alfanuméricos dos componentes da rede externa. Durante uma sessão de trabalho no SAGRE, o usuário visualiza representações gráficas da rede externa, ou seja, mapas. A seleção de um elemento da rede externa ativa o módulo GAT que permite a manipulação dos atributos do elemento selecionado. A figura 12.3 mostra uma instância do GAT para o objeto Poste.

O SAGRE permite que o usuário visualize de várias formas e com diferentes níveis de detalhes, as informações do banco de dados geográfico subjacente. Esta visualização pode ser textual, gráfica ou uma combinação de ambas. A apresentação abrange diversas escalas, havendo possibilidade de operações de *zoom* e *panning* para navegar em uma região. Há dois tipos de operações básicas: manipulação de informações gráficas e manipulação de atributos.

As informações gráficas são manipuladas através de uma tela denominada **Mapa Base** e uma tela específica para a manipulação de esquemáticos. A edição gráfica possibilita a seleção e a edição de objetos gráficos, georeferenciados através de coordenadas UTM (Universe Transverse Mercator), em escala definida pelo usuário. Durante a manipulação dos objetos no SAGREcad, uma série de mensagens são enviadas ao usuário visando orientá-lo durante a execução das operações, bem como informá-lo de possíveis erros ocorridos.

Os elementos do cadastro de rede externa são, em geral, referenciados ao MUB e, para a visualização, são desenhados sobre a “imagem” do MUB. Além disso, alguns elementos podem ser associados a outros desenhos ou diagramas, que não são desenhados sobre o MUB, e que podem ser armazenados em formato *raster* ou vetorial.

Deve-se observar que em algumas situações, o mesmo elemento do banco de dados pode ser representado graficamente em mais de um esquemático. Um lance de cabo, por exemplo, pode ter uma emenda inicial em uma determinada caixa subterrânea, passar direto por uma ou mais caixas e terminar o seu percurso em uma emenda localizada em outra caixa.

Poste

ArquivoRelacionadosOperacoesDetalheOpcoesAjuda

Area de EdicaoConsultaEXECUTAR

Id1001

Flag Grafico1

Altura5

Propriedade1

Status2>instalado

Tipo80-75-CTPA-APL-G

Area de Selecao: 40

ID	GRAFICO	ALTURA	PROPRIEDADE	
1001	1	5	1	2 > i
1002	2	6	1	1 > p
1003	3	7	1	5 > m
1004	4	8	1	1 > p

RETORNA

Operacao : Insercao - SI = 0 - SA = 0  
Operacao : Renocao - SI = 2 - SA = 2  
Operacao : label op 1 - SI = 1819048224 - SA = 1768122721

Figura 12.3: Interface de atributos do SAGRE



A simbologia pode ser considerada como uma consulta dinâmica que, ao mostrar o lance de cabo (segmento de linha), mostra também (com símbolos e letras) alguns dos atributos e relacionamentos do mesmo. A visualização de elementos, por razões estéticas e regras de apresentação, nem sempre segue os relacionamentos armazenados. Por exemplo, o banco de dados informa quais lances de cabo estão apoiados em que postes. Na apresentação, o segmento de linha que representa o lance de cabo não toca os símbolos que representam os postes.

## 12.7 Mecanismos Internos

### 12.7.1 Conversão dos Dados para Entrada no Sistema

A construção de bancos de dados para o SAGRE envolve um processo de conversão de dados, crítico para o sistema em implantação, pois o desempenho do sistema dependerá da qualidade dos dados [MGS<sup>+</sup>94]. A conversão é realizada por empresas especializadas, tendo em vista sua complexidade.

Normalmente, as empresas de conversão trabalham com ferramental próprio e fornecem os dados convertidos nos formatos (proprietários) dos SIG utilizados. Estes formatos são definidos num nível muito primitivo, tornando o processo de aceitação complicado e oneroso.

Para reduzir os custos, melhorar a qualidade, reduzir o tempo de conversão, e facilitar todo o processo de aceitação dos dados, a TELEBRÁS optou por definir um formato próprio para a conversão.

Este formato, orientado aos elementos de mapeamento urbano e rede externa, é definido numa linguagem de alto nível nos termos utilizados pelos técnicos e engenheiros das empresas operadoras. Com isto, as empresas de conversão não precisam se especializar nos formatos proprietários dos SIG. Outra vantagem é que este formato abre o mercado para empresas nacionais.

Existe uma especificação técnica detalhada em 600 páginas onde estão descritas todas as regras da conversão. Para cada elemento a ser convertido, são descritas as formas de como devem ser representados após a conversão (Descrição de Elemento Gráfico ou DEG), como se apresentam nas plantas (Descrição de Elemento de Planta ou DEP) e a forma como os dados, produto da conversão, devem ser armazenados (Descrição de Formato de Conversão ou DFC). O formato de conversão precisa considerar elementos gráficos e textuais que dependem de cada tipo de elemento a ser convertido. Além disso, a conversão precisa levar em conta dados como data de instalação, afastamento de elementos em relação a pontos de referência, coordenadas de posicionamento (que pode chegar a constituir uma lista de coordenadas, dependendo do tipo de elemento).

## 12.7.2 Controle de Versões

Nos casos de aplicações ambientais, usuários tipicamente trabalham com transações longas, em que inúmeros cenários são gerados a partir de operações de análise, testando diferentes situações, levando a um diagnóstico final. Em ambientes de projeto como o SAGRE, a situação é outra: ao mesmo tempo, projetistas trabalham com transações longas e aninhadas enquanto que operadores manipulam os dados em regime *on-line* com transações curtas. A solução encontrada foi a implementação de um sistema de versões [DGM95].

A solução permite a convivência controlada de atualizações de curta duração (garantidas em princípio pelo mecanismo de controle de concorrência do SGBD relacional do SAGRE) e atualizações de longa duração. Estas últimas envolvem inclusive mudanças condicionais, isto é, que podem ser efetivadas ou revogadas. Além disso, a representação gráfica de um elemento condicional (projetado) pode ser diferente da sua representação na rede existente.

Geralmente, versões são utilizadas para resolver problemas num ambiente de projeto onde vários usuários trabalham cooperativamente para produzir um projeto, gerando alternativas. No SAGRE, o ambiente alvo é bem similar porém com um agravante. Projetos de redes de telecomunicações geralmente atuam sobre uma área geográfica onde já existe alguma rede implantada (rede já em operação). Assim, os projetistas trabalham sobre uma situação existente que não pode ser congelada, uma vez que as redes implantadas são atualizadas *on-line*. Estas alterações podem ser provocadas por acidentes (por exemplo, um temporal pode danificar a rede aérea) ou por pressão da demanda por serviços.

Os projetistas, embora num ambiente de transações longas, gostariam de trabalhar com os dados na versão mais recente mesmo que isto venha a invalidar algumas tarefas já realizadas. Por exemplo, ao lançar um novo lance de cabo aéreo os postes sobre os quais o cabo se apoiará podem ter sido substituídos ou até mesmo retirados.

Outra característica importante é que a maioria das alterações projetadas para a rede são inerentemente condicionais, isto é, nada garante que as mesmas serão efetivadas (implantadas). As atualizações são condicionais por dependerem de dados existentes que foram alterados durante o ciclo do projeto. Quanto maior o grau de concorrência entre os projetistas numa determinada área geográfica, maior a probabilidade de haver conflitos de dependências entre dados projetados e existentes.

A gerência de versões é baseada no esquema de *check-in/check-out* proposta inicialmente em [CK86]. O projetista antes de qualquer ação extrai para um banco de dados de trabalho os dados existentes na área geográfica de interesse – operação de *check-out*. Todas as operações de projeto são executadas neste banco de dados de trabalho, sendo os dados criados todos condicionais. A seleção de elementos para as operações de transferência de dados é feita através de coordenadas geográficas.

Quando as alterações estiverem concluídas e prontas para a instalação, os dados alterados são retornados ao banco de dados central através da operação de *check-in*.

Se houver projetos que se sobreponham em uma mesma área geográfica, isto acarreta a

existência de múltiplos bancos de dados para a área: o central (permanente) e os de trabalho. Somente os dados do primeiro *check-in* serão aceitos, invalidando as modificações ocorridas nestes dados nos outros bancos de dados de trabalho. Esta invalidação ocorre apenas para os elementos atualizados e não para toda a área de interseção de projetos. Havendo invalidação, cada banco de dados que teve seus dados invalidados deve reextraí-los, atualizando-os conforme o banco de dados central.

A existência de dados condicionais e permanentes requer um tratamento diferenciado no armazenamento destes dados no banco de dados central. A maneira encontrada foi garantir uma ordenação parcial das versões de um dado conjunto de elementos. A versão zero armazena os dados permanentes enquanto que as outras versões de projeto representam os dados condicionais. No banco de dados, todos os registros que compõem uma mesma versão de elemento estão associados a um mesmo número de versão. A atualização do banco de dados para o estado permanente é baseada no conceito de dependências de versões, que é função da ordem relativa de criação de versões no tempo. Maiores detalhes estão em [DGM95].

A implementação dos mecanismos de versão requereu uma série de serviços não disponíveis em SGBD relacionais comerciais. Primeiro, a atividade de *check-in/check-out* exigiu a implementação de protocolos de bloqueio especiais. O SIG utilizado no SAGRE oferece uma camada de serviços sobre o SGBD relacional que inclui alguns recursos adicionais que simplificaram a implementação do mecanismo de versões. Por exemplo, o servidor de identificadores deste SIG foi utilizado para garantir a unicidade de identificação dos dados e a ordenação parcial dos identificadores de versão, facilitando a identificação dos registros com dependências de versões. O esquema de exportação/importação de dados foi realizado através de um formato de transferência provido pelo SIG.

# Capítulo 13

## Centros de Dados Geográficos

### 13.1 Introdução

Este capítulo discute soluções emergentes que tratam do gerenciamento de dados geográficos em grande volume, de forma a permitir seu uso cooperativo por agências governamentais ou privadas. Soluções para este problema envolvem o desenvolvimento de *cooperativas* e *centros de dados geográficos*. A seção 13.2 introduz conceitos e definições. A seção 13.3 discute as funções de um centro de dados hipotético e a seção 13.4 descreve sucintamente alguns projetos correlatos.

### 13.2 Cooperativa de Dados e Centro de Dados

Várias agências, governamentais ou não, atuam na aquisição, armazenamento, processamento e distribuição de diversos tipos de dados geográficos para os mais variados fins, tais como previsões agrícolas, análises ambientais e planejamento urbano. Para realizar suas tarefas, uma agência, pode utilizar dados de outras agências, além dos dados coletados e produzidos internamente. Assim, o trabalho entre agências pode ser visto como um trabalho cooperativo. Contudo, esta cooperação é, por muitas vezes, prejudicada pela grande dispersão geográfica existente entre as agências, e agravada pela ausência de uma padronização para os dados.

O acesso a dados ambientais sofre com o fato de cada agência arquivar seus dados em forma particular, de acordo com as suas necessidades (por exemplo, de acordo com o SIG utilizado), complicando a coordenação e combinação destes múltiplos conjuntos de dados. Esses dados gerados são subutilizados ou até mesmo recriados a cada nova aplicação. Por exemplo, nos EUA, mais de 20 agências federais gastam anualmente por volta de 5 bilhões de dólares em pesquisa científica sobre o meio ambiente. Estes problemas de cooperação prejudicam as tomadas de decisões e impedem a utilização efetiva de dados já disponíveis.

Nesse contexto, uma *cooperativa de dados* é um conjunto de agências interligadas através de um sistema de informação global de dados (vide figura 13.1), que, por sua vez, é formado por

centros de dados interligados através de um sistema de comunicação de dados. O sistema de comunicação pode ser uma rede privada, interligando as agências de uma cooperativa, ou uma rede pública, permitindo a comunicação com outras agências externas, sendo possivelmente composto pelas duas redes.

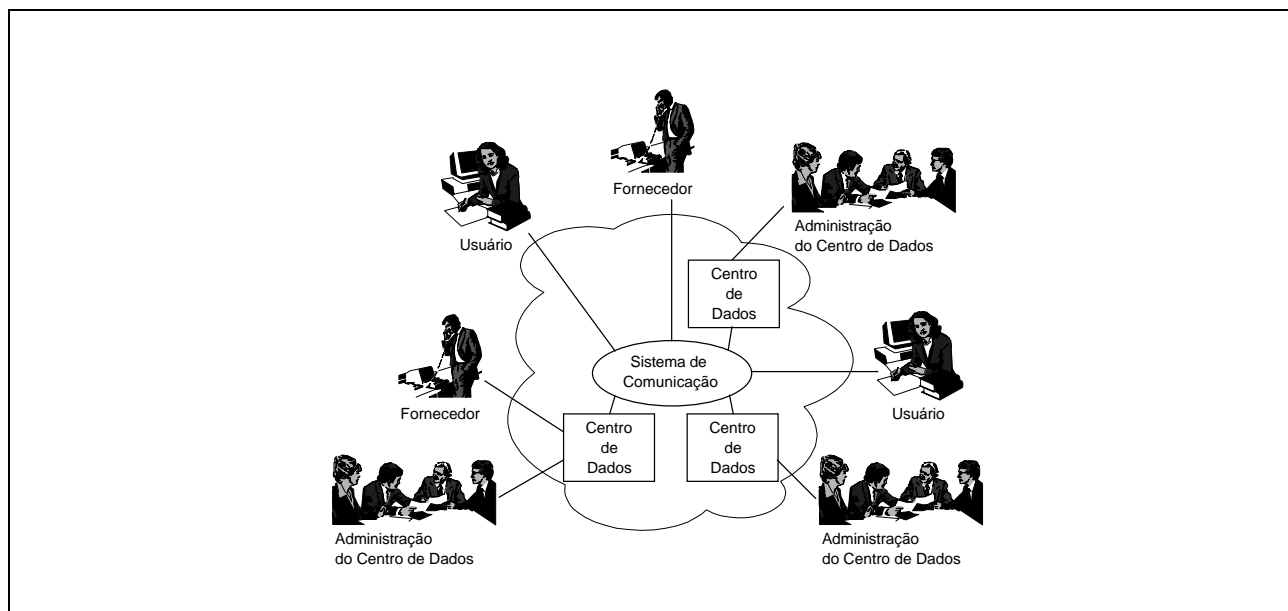


Figura 13.1: Cooperativa de dados

Um *centro de dados* é uma facilidade que permite coleta, armazenamento, processamento, acesso e distribuição de dados (ou programas) através do sistema de comunicação. Um *Sistema de Gerência de Centro de Dados* (SGCD) é responsável pela gerência dos dados em centro de dados, mas não obrigatoriamente por sua propriedade.

Uma agência pode exercer diversos papéis dentro da cooperativa, quer administrando um ou mais centros de dados, quer fornecendo dados e produtos para tais centros, ou mesmo atuando como simples usuário do sistema de informação global. O resultado deste tipo de cooperação é útil não apenas para as agências envolvidas, como também para usuários externos, aos quais os dados podem ser tornados disponíveis, segundo diferentes políticas de acesso.

Um SGCD deverá, em particular, fornecer mecanismos para garantir a segurança dos dados, proteção do patrimônio individual de cada agência, acesso controlado de acordo com a tarifação ou comprometimento da agência (tanto participante como não participante da cooperativa), sendo de responsabilidade dos administradores de cada um dos centros de dados componentes zelar pela manutenção das regras estabelecidas de comum acordo entre os participantes da cooperativa.

## 13.3 Funções de um Centro de Dados

Nesta seção, discutimos características gerais de centro de dados, sintetizadas a partir da descrição de diversos centros de dados. Na seção 13.4 apresentamos alguns projetos de centros de dados.

Em alto nível, um sistema gerenciador de centro de dados deve ser capaz de lidar com quatro entidades: *dados*, *produtos*, *usuários* e *fornecedores*. Dados, usuários e metadados já são comumente tratados por SGBDs, sendo o tratamento feito de forma análoga por SGCDs. Um *produto* é a informação gerada a partir do processamento de dados armazenados, podendo consistir em dados ou programas. Este processamento pode ser tão simples quanto o encapsulamento de um dado para um formato pré-determinado ou tão sofisticado quanto a geração de um mapa a partir de temas selecionados pelo usuário e incorporando procedimentos não automáticos. É possível especificar restrições ao acesso de usuários a dados, produtos e a funções do SGCD, de forma semelhante ao acesso a dados em SGBDs. A manutenção do cadastro de fornecedores é importante para a gerência de atualização e o controle de qualidade do conteúdo do banco de dados no centro de dados.

Usualmente, um SGCD é uma aplicação sobre um SGBD (ou SIG, em função da variedade de funções de processamento que se deseja disponibilizar). Assim, a discussão sobre arquiteturas para SIGs, apresentada no capítulo 8, é pertinente para centros de dados. Centros de dados caracterizam-se por armazenarem dados pouco voláteis, existentes a longo termo e, portanto, sua arquitetura deve ser capaz de acomodar evoluções tecnológicas.

As principais funções de um centro de dados, descritas a seguir, envolvem aquisição, gerenciamento e disponibilização de dados, de forma a permitir integração entre diferentes centros.

### Aquisição de Dados

O processo de aquisição inclui a coleta e pré-processamento dos dados, a sua conversão para um formato de troca, controle de qualidade e inserção no banco de dados de um centro de dados. A execução das etapas de aquisição depende do tipo dos dados e da aplicação – por exemplo, imagens de satélite e dados pluviométricos são adquiridos de forma distinta. Neste processo, a integração ou migração de sistemas de bancos de dados de produção legados deve ser prevista [SSB95].

Um formato de troca visa garantir a interoperabilidade entre sistemas. Ele deve ser autocontido, ou seja, deve encapsular todas as informações relevantes para a compreensão do dado. O dado de troca pode ser um objeto, incluindo inclusive métodos para sua manipulação. Elkington [EMM94] argumenta que uma cooperativa não deve ditar um formato de troca único e rígido, dado que agências podem ter preferências particulares sobre formatos distintos. Um SGCD deve disponibilizar algumas rotinas de conversão de formatos entre os padrões internacionais mais utilizados e os formatos de troca adotados na cooperativa.

O pré-processamento envolve conversão de formato e correção de dados para garantir sua

qualidade. No controle de qualidade, os dados devem ser verificados de acordo com os metadados associados. Os resultados desta verificação devem ser enviados para o fornecedor. Por exemplo, uma imagem de satélite coletada pode ser inicialmente corrigida radio e geometricamente, através de rotinas de processamento de imagens. Ela também deve passar por um processo de georeferenciamento e ser convertida para um padrão de troca. Informações relevantes são então extraídas para futura indexação (como discutido no capítulo 9).

É interessante porém observar que, em muitos casos, há interesse em se armazenar historicamente dados brutos, já que pode haver uma variedade de algoritmos de pré-processamento, e nem sempre estes permitem a recuperação do dado original a partir do pré-processado.

Na inserção de dados, o sistema deve prever a geração automática de amostras para campos longos, a fim de facilitar o folheamento dos dados por parte do usuário. Por exemplo, no caso das imagens de satélite, amostras podem ser obtidas através de subamostragem dos pixels.

## **Armazenamento e Arquivamento de Dados**

A função de armazenamento e arquivamento de dados oferece serviços para as demais funções. O objetivo desta função é gerenciar o armazenamento de dados, de acordo com políticas de arquivamento estabelecidas.

Há basicamente três níveis de dados armazenados: os metadados, o catálogo de dados e os dados propriamente ditos. Os metadados descrevem os dados armazenados. O catálogo mantém informações condensadas sobre instâncias de dados, incluindo amostras de valores longos. Os dados propriamente ditos são freqüentemente volumosos, numerosos e históricos. Atualmente, grande parte dos sistemas em operação armazena o catálogo e os metadados em um SGBD. Já os dados ficam em um sistema de arquivos, com gerência de hierarquia de memória, com uma arquitetura semelhante à dual no que diz respeito a armazenamento. Como o catálogo por sua vez também tende a ficar com muitas entradas (muitas entidades, dados históricos), alguns SGBDs têm conexão direta com tais sistemas de gerência de memória, segundo a política de arquivamento estabelecida.

Os dados devem ser armazenados utilizando um formato neutro, padrão, a fim de minimizar os esforços de conversão de formato. Este formato de armazenamento visa a eficiência na recuperação da informação e no uso de memória. Eventualmente, o formato de troca e o formato de armazenamento podem ser os mesmos.

## **Disponibilização de Dados e Programas**

O objetivo da função de disponibilização de dados e programas é tornar os bancos de dados disponíveis para consulta tanto para os usuários de uma cooperativa quanto para eventuais usuários externos. Consultas envolvendo mais de um centro de dados geram subconsultas a serem formuladas aos centros adequados, sendo suas respostas a seguir integradas em um resultado final.

As consultas podem envolver tanto dados quanto metadados [SSKS95], inclusive tornando disponíveis programas de uso genérico, tais como programas para visualização e animação.

A diversidade potencial de dados e usuários requer atenção especial quanto a estratégias de busca e acesso. A consulta, feita via rede, pode ser navegacional ou associativa, através de uma interface hipermídia [NS91]. Consultas espaciais devem ser permitidas, como por exemplo as descritas no capítulo 6. O aspecto de segurança de acesso é importante. Assim, a disponibilização de dados e funções deve ser customizada de acordo com o usuário.

Após uma sessão de consultas, um usuário gera uma lista de dados e programas selecionados, a partir da qual é possível a solicitação de produtos, conforme discutido a seguir.

### **Distribuição de Produtos**

Esta função abrange o gerenciamento da solicitação e distribuição de produtos para usuários, sendo a distribuição escalonada conforme prioridades de atendimento. Os usuários podem emitir ordens de pedidos de produtos a partir da lista de dados e programas selecionados, acrescentando possíveis informações necessárias para a geração do produto (por exemplo, especificação da mídia de distribuição e do formato de troca).

Os produtos podem ser classificados para facilitar este gerenciamento, considerando por exemplo informações sobre responsabilidades, tempo de médio de produção e outros. A contabilização de dados sobre sessões de usuários ajuda a prever solicitações de produtos, os quais podem ser gerados e armazenados em antecipação à demanda. Esta contabilização também é útil para outros fins, como por exemplo tarifação, otimização de índices de acesso, revisão de políticas de arquivamento, ou definição de perfis de usuários.

Um problema na função de distribuição é a existência de cópias ilícitas. Alguns sistemas [CDD<sup>+</sup>95] sugerem o uso de “marcas d’água” (“watermarking”) identificando a procedência ou os usuários requisitantes nos produtos sendo distribuídos, seja eletrônica ou fisicamente, a fim de inibir a difusão de tais cópias.

### **Integração com Outros Centros de Dados**

Esta função é responsável pela integração de dados e processamentos entre os centros de dados de uma cooperativa. Ela visa permitir combinar diferentes centros de dados apesar de diferenças em seu vocabulário, completude e representações. A adoção de um único padrão de metadados facilita esta tarefa, tendo sido esta uma estratégia adotada pelo FGDC (“Federal Geospatial Data Committee - Content Standard for Digital Geospatial Metadata”) (veja seção 4.3). Caso isto não seja possível, o sistema deve ajudar o usuário a lidar com as diferenças e eventuais incompatibilidades do sistema [EMM94].

A integração de processamentos tem por objetivo a comunicação entre processamentos pela rede de comunicação, possivelmente se baseando em uma especificação tipo OGIS [Car94] ou CORBA [Obj94] conforme discutido na seção 8.4.



A integração de dados e o processamento de transações são áreas de pesquisa ativas em bancos de dados múltiplos (“multidatabases”) [SL90, LMR90]. Em um ambiente de informações globais, no entanto, estas questões assumem novas dimensões, conforme discutimos na próxima seção.

## Outras Funções

Outras funções incluem a gerência de usuários, produtos e fornecedores, a gerência de aquisição e distribuição de dados, o processamento de dados propriamente dito e funções administrativas em geral.

A *gerência de usuários, produtos e fornecedores* controla os cadastros respectivos, agrupando-os em função de seu relacionamento com a cooperativa, de acordo com níveis diferenciados de acesso.

A função de *processamento de dados* é basicamente responsável pela geração de produtos, realizando, por exemplo, encapsulamento de dados, conversões de formato, classificação de imagens, geração de mapas.

A *gerência de aquisição e distribuição de dados* é responsável por gerenciar a frequência com que o banco de dados deve ser atualizado. Por exemplo, um centro de dados que disponibilize imagens do satélite Landsat deve garantir a atualização periódica das imagens. Além disso, usuários poderão fazer *assinaturas* de produtos dos quais necessitem com certa periodicidade.

Finalmente, a função de *administração* é responsável pelas tarefas usuais de administração de dados tais como “back-up” e recuperação de dados. Alguns outros serviços, tais como administrar um fórum aberto de discussões via rede, também são interessantes para a cooperativa.

## 13.4 Exemplos

Nesta seção apresentamos brevemente algumas das iniciativas que envolvem cooperativas e centros de dados geográficos [STMS94].

A iniciativa DLT (“Digital Library Technology”), nos EUA, tem por objetivo o desenvolvimento de novas tecnologias para facilitar o acesso público aos dados da NASA via redes de computadores, em especial ferramentas, aplicações e hardware que enderecem a questão de escalabilidade. A ADL (“Alexandria Digital Library”) [FAB<sup>+</sup>95] é um dos projetos patrocinados por esta iniciativa, sendo desenvolvida na Universidade da Califórnia em Santa Bárbara. A ADL é um projeto de um consórcio de diversos grupos, incluindo setores privados, tendo o objetivo de desenvolver uma biblioteca digital composta de vários nós distribuídos sobre a rede pública Internet. Seu foco inicial é em dados geográficos. Já existe um primeiro protótipo, que armazena mapas digitalizados, fotografias aéreas e imagens de satélite. Nas próximas fases, o projeto pretende tratar também de classes espaciais genéricas e textos.

O EOSDIS (“Earth Observation System / Data and Information System”) [EMM94, EW94]

é um dos componentes da Missão do Planeta Terra, projeto de pesquisa multi- e interdisciplinar da NASA. Seu objetivo é permitir acesso fácil e confiável aos dados do EOS (“Earth Observation System”), em sua maioria obtidos a partir de satélites em órbita baixa da Terra. O EOSDIS baseia-se em uma cooperativa de dados com inicialmente oito centros de dados espalhados pelo território americano, responsáveis pela produção, arquivamento e distribuição de produtos, e diversos centros de processamento, para desenvolver e manter programas, validar dados e programas, gerar produtos e outras atividades. Os dados serão coletados durante 15 anos a partir de 20 instrumentos diferentes colocados a bordo de seis satélites especiais que serão postos em órbita entre os anos de 1998 e 2002. O volume de dados previsto é de 1Tbyte por dia, que deverá ser formatado, calibrado e armazenado para acesso pelos diferentes grupos de pesquisa. Estima-se, atualmente, que cerca de 10000 grupos de pesquisa diferentes deverão utilizar tais dados, além das eventuais consultas que poderão ser feitas através de redes como a Internet. Duas redes de comunicação são previstas, uma dedicada, corporativa, e outra externa, pública.

O ERC (“Environment Resource Center”) tem o objetivo de oferecer facilidades para identificar, acessar e integrar dados ambientais existentes. É um projeto entre a indústria privada e diversos níveis do governo americano, como a NASA e o Estado do Texas. Mais especificamente, o ERC deverá oferecer um SIG integrado, com acesso fácil a dados brutos ou interpretados via interfaces gráficas, capacidade de compartilhar dados entre usuários e aplicações de forma transparente, permitindo integração simples e barata de novos dados e novas aplicações. O ERC prevê ferramentas para visualização, animação e manipulação interativa de dados, interface de comunicação padrão sobre WAN e fácil acesso de usuários a dados através de interfaces via “e-mail” e redes como a Internet. A primeira fase deste projeto foi iniciada em abril de 1994 com a construção de um protótipo, atualmente operacional.

No Brasil, o projeto ProTeM-CC GEOTEC tem interação com o Projeto CODEX. O objetivo do CODEX é prover um conjunto de serviços com metodologias bem definidas e componentes para especificação, desenvolvimento, instalação e operação de um centro de dados para aplicações ambientais [CGFH95].



# Referências Bibliográficas

- [AA93] G. Alonso and A. El Abbadi. GOOSE: Geographic Object Oriented Support Environment. In *Proc. ACM/ISCA Workshop on Advances in Geographic Information Systems*, pages 38–43, 1993.
- [ABC<sup>+</sup>91] J. Antenucci, K. Brown, P. Croswell, M. Kevany, and H. Archer. *Geographic Information Systems - a Guide to the Technology*. Van Nostrand Reinhold, 1991.
- [AHW89] H.-W. Six A. Henrich and P. Widmayer. The LSD-Tree: Spatial Access to Multi-dimensional Point- and Non-Point-Objects. In *Proc 15th VLDB Conference*, pages 45–53, 1989.
- [AKK94] M. Arikawa, H. Kawakita, and Y. Kambayasshi. Dynamic Maps as Composite Views of Varied Geographic Database Servers. In *Proc. International Conf. on Applications of Databases*, 1994.
- [ANS75] ANSI. American National Standards Institute Study Group on Database Management Systems. Technical report, ANSI, 1975.
- [ANS94] ANSI. *SQL - Multimedia Spatial*, 1994. Draft.
- [Arn93] E. Arnberger. *Thematische Kartographie*. Westermann Schulbuchverlag, Braunschweig, 1993.
- [Aro89] S. Aronoff. *Geographic Information Systems*. WDL Publications, Canada, 1989.
- [AS91] W. Aref and H. Samet. Optimization Strategies for Spatial Query Processing. In *Proc. 17th VLDB Conference*, pages 81–90, 1991.
- [AS93] E. Assad and E. Sano. *Sistema de Informações Geográficas: Aplicações na Agricultura*. Edição Embrapa/CPAC, 1993.
- [AYA<sup>+</sup>92] D. Abel, S. Yap, R. Ackland, M. Cameron, D. Smith, and G. Walker. Environmental Decision Support System Project: an Exploration of Alternative Architectures for Geographical Information Systems. *International Journal of Geographical Information Systems*, pages 193–204, 1992.
- [Bar92] N. Barry. How OLE works. *BYTE*, 17(11), 1992.

- [Bat92] P. Batty. Exploiting relational database technology in a GIS. *Computers and Geosciences: An international journal*, 18(4):453–462, 1992.
- [BDK92] F. Bancillon, C. Delobel, and P. Kanellakis, editors. *Building an Object-Oriented System – the Story of O2*. Morgan Kaufmann, 1992.
- [BDN93] C. Broiato, C. Domingues, and M. Néia. A Experiência da Implantação de um Sistema de Informações Geográficas na Prefeitura Municipal de Santo André. In *IV Conferência Latinoamericana sobre Sistemas de Informação Geográfica*, pages 365–382, 1993.
- [Bee89] C. Beeri. Formal Models for Object-oriented Databases. In *Proc. 1st International Conference on Deductive and Object-oriented Databases*, pages 370–395, 1989.
- [BEF93] L. Bins, G. Erthal, and L. Fonseca. Um Método para Classificação Não-Supervisionada de Regiões. In *Proc. VI Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens*, 1993.
- [BER85] D. Beckley, M. Evens, and V. Raman. Multikey Retrieval from K-d trees and Quad-trees. In *Proc. of the ACM SIGMOD Conference*, pages 291–301, maio 1985.
- [Ber93] J. Berry. Cartographic Modeling: the Analytical Capabilities of GIS. In M. Goodchild, B. Parks, and L. Steyaert, editors, *Environmental Modelling with GIS*, pages 58–74. Oxford University Press, 1993.
- [BFG93] J. Boyle, J. Fothergill, and P. Gray. Design of a 3D User Interface to a Database. In J. Lee and G. Grinstein, editors, *Proc. IEEE Visualization Workshop*, pages 273–185, 1993. Springer Verlag Lecture Notes in Computer Science 871.
- [BHKS93] T. Brinkhoff, H. Horn, H. Kriegel, and R. Schneider. A Storage and Access Architecture for Efficient Query Processing in Spatial Database Systems. In *Proc. 3rd International Symposium on Large Spatial Databases*, pages 357–376, 1993.
- [BK85] D. Batory and W. Kim. Modeling Concepts for VLSI CAD Objects. *ACM Transactions on Database Systems*, 10(3):322–346, 1985.
- [BK91] N. Barghouti and G. Kaiser. Concurrency Control in Advanced Database Applications. *ACM Computing Surveys*, 23(3):319–344, 1991.
- [BK95] M. Bieber and C. Kacmar. Designing Hypertext Support for Computational Applications. *Communications of the ACM*, 38(8):99–107, 1995.
- [BKS93] T. Brinkhoff, H. Kriegel, and B. Seeger. Efficient Processing of Spatial Joins Using R-Trees. In *Proc. of the ACM SIGMOD Conference*, junho 1993.
- [BKSS90] N. Beckmann, H. Kriegel, R. Schneider, and B. Seeger. The  $R^*$ -tree: An Efficient and Robust Access Method for Points and Rectangles. In *Proc. of the ACM SIGMOD Conference*, pages 322–331, 1990.

- [BKSS94] T. Brinkhoff, H-P. Kriegel, R. Schneider, and B. Seeger. Multi-Step Processing of Spatial Joins. In *Proc. of the ACM SIGMOD Conf. on Management of Data*, pages 197–208, 1994.
- [BM92] P Boursier and M Mainguenaud. Spatial Query Languages: Extended SQL vs Visual Languages vs Hypermaps. In *Proc 5th International Symposium on Spatial Data Handling*, pages 249–259, 1992. Volume 1.
- [BML94] V. Barbosa, R. Machado, and F. Liporace. A Neural System for Deforestation Monitoring on Landsat Images of the Amazon Region. *International Journal of Approximate Reasoning*, pages 321–359, 1994.
- [Bot95] M. A. Botelho. Incorporação de Facilidades Espaço Temporais em um Banco de Dados Orientado a Objetos. Master’s thesis, UNICAMP, dezembro 1995.
- [BP94] M. Beguin and D. Pumain. *La Representation des Données Geographiques. Statistique et Cartographie*. Armand Colin, 1994.
- [Bro94] K. Brockschmidt. OLE Integration Technologies. *Dr. Dobb’s Special Report on Interoperable Objects Revolution*, pages 42–49, 1994.
- [Bul94] G. Bull. Ecosystem Modelling with GIS. *Environmental Management*, 18(3):345–349, 1994.
- [Bur86] P. Burrough. *Principles of Geographical Information Systems for Land Resources Assessment*. Clarendon Press, 1986.
- [Bur92] P. Burrough. Development of Intelligent Geographical Information Systems. *International Journal of Geographical Information Systems*, 6(1):1–12, 1992.
- [Bur93] P. Burrough. Soil Information Systems. In D. J. Maguire, M. F. Goodchild, and D. W. Rhind, editors, *Geographical Information Systems - volume II*, pages 153–169. John Wiley and Sons, 1993.
- [BX94a] M. Batty and Y. Xie. Modelling inside GIS: Part 2. Selecting and Calibrating Urban Models Using Arc-Info. *International Journal of Geographical Information Systems*, 8(4):451–470, 1994.
- [BX94b] M. Batty and Y. Xie. Modelling inside GIS: Part I. Model Structures, Exploratory Spatial Data Analysis and Aggregation. *International Journal of Geographical Information Systems*, 8(3):291–308, 1994.
- [Cam93] G. Camara. Anatomia de Sistemas de Geoprocessamento: Visão Atual e Perspectivas de Evolução. In *Anais do II Simpósio Brasileiro de Geoprocessamento*, pages 157–183, 1993.
- [Cam94a] G. Camara. Análise de Arquiteturas para Bancos de Dados Geográficos Orientados a Objetos . Technical report, INPE, 1994.

- [Cam94b] F. Campagnoni. IBM's System Object Model. *Dr. Dobb's Special Report on Interoperable Objects Revolution*, pages 24–28, 1994.
- [Cam95] G. Camara. *Modelos, Linguagens e Arquiteturas para Bancos de Dados Geográficos*. PhD thesis, INPE, dezembro 1995.
- [Car94] C. Cargill. Standard View - ACM Perspectives on Standardization. *Communications of the ACM*, 2(3), 1994.
- [Cas93] S. Casettari. *Introduction to Integrated Geo-Information Management*. Chapman & Hall, 1993.
- [Cat94] R.G. Catell. *The Object Database Standard*. Morgan Kauffmann, San Francisco, 1994.
- [CDD<sup>+</sup>95] D. Choy, R. Dievendorff, C. Dwork, J. Lotspiech, R. Morris, L. Anderson, A. Bell, S. Boyer, T. Griffin, B. Hoenig, J. McCrossin, A. Miller, N. Pass, F. Pestoni, and D. Picciano. The Almaden Distributed Digital Library System. In *ADL'95 - A Forum on Reserach and Technology Advances in Digital Libraries*, 1995.
- [CdFvO93] E. Clementini, P. di Felice, and P. van Oosterom. A Small Set of Formal Topological Relationships Suitable for End-User Interaction. In *Proc. 3rd Symposium Spatial Database Systems*, pages 277–295, 1993.
- [CFC94] G. Camara, U. Freitas, and J. Cordeiro. Towards an Algebra of Geographical Fields. In *VII Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens*, 1994.
- [CFR87] T. Sellis C. Faloutsos and N. Rossopoulos. Analysis of Object-Oriented Spatial Access Methods. In *Proc. of the ACM SIGMOD Conference*, pages 426–439, 1987.
- [CFS<sup>+</sup>94] G. Camara, U. Freitas, R. Souza, J. Paiva, M. Casanova, and A. Hemerly. A General Data Model for Integrating Remote Sensing and GIS Data. In *Proc. ISPRS Symposium on Mapping and Geographical Information Systems*, 1994.
- [CGFH95] S. Cruz, L. Gonçalves, S. Fuks, and A. Hemerly. Proposta de uma Cooperativa de Banco de Dados sobre Recursos Naturais na EMBRAPA. In *AgroSoft'95 - Seminário Internacional de Informatização da Agropecuária*, 1995.
- [CHF93a] M. Casanova, A. Hemerly, and A. Furtado. Cooperative Environments for Geographic Databases: a Prescriptive Analysis. In *Anais, 8 Simposio Brasileiro de Bancos de Dados*, pages 266–279, 1993.
- [CHF93b] M. Casanova, A. Hemerly, and A. Furtado. Um Gerenciador para Atributos Espaciais de Objetos Geográficos. In *Anais da IV Conferencia Latino-americana sobre SIG*, pages 223–235, 1993.

- [CHK93] D. Clark, D. Hastings, and J. Kineman. Global databases and their implications for GIS. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume II*, pages 217–231. John Wiley and Sons, 1993.
- [Chr90] N. Chrisman. Deficiencies of Sheets and Tiles: Building Sheetless Databases. *International Journal of Geographical Information Systems*, 4(2):157–168, 1990.
- [Chr93] N. Chrisman. The Error Component in Spatial Data. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume I*, pages 165–174. John Wiley and Sons, 1993.
- [Cif95] R. Ciferri. Um Benchmark Voltado a Análise de Desempenho de Sistemas de Informações Geográficas. Master’s thesis, UNICAMP, junho 1995.
- [CK86] H. Chou and W. Kim. A Unifying Framework for Version Control in a CAD Environment. In *Proc. 12th VLDB Conference*, pages 336–344, 1986.
- [Cou92] H. Couclelis. People Manipulate Objects (but Cultivate Fields): Beyond the Raster-Vector Debate in GIS. In *Proc International Conference on GIS - From Space to Territory: Theories and Methods of Spatial Reasoning*, pages 65–77. Springer Verlag Lecture Notes in Computer Science 639, 1992.
- [Cow90] D. Cowen. GIS versus CAD versus DBMS: what are the Differences? In *Introductory Readings in Geographical Information Systems*, pages 52–61. Taylor and Francis, 1990.
- [Cox91] F. Cox. Análise de Métodos de Acesso a Dados Espaciais Aplicados a Sistemas Gerenciadores de Banco de Dados. Master’s thesis, UNICAMP, dezembro 1991.
- [CSE94] E. Clementini, J. Sharma, and M. Egenhofer. Modelling Topological Spatial Relations: Strategies for Query Processing. *Computer and Graphics*, 18(6):815–822, 1994.
- [CSF92] G. Camara, R. C. M. Souza, and U. Freitas. SPRING: Object-oriented Geoprocessing. In *XV International Congress of Photogrammetry and Remote Sensing*, 1992.
- [CSFP93] G. Camara, R.C.M. Souza, U. Freitas, and J. Paiva. SPRING: Concepção, Evolução, Perspectivas. In *Proc. VII Simposio Brasileiro de Sensoriamento Remoto*, 1993.
- [Dal93] P.F. Dale. Land Information Systems. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume II*, pages 85–99. John Wiley and Sons, 1993.
- [Dan90] K. Dangermond. A Classification of Software Components Commonly Used in Geographic Information Systems. In *Introductory Readings in Geographical Information Systems*, pages 31–51. Taylor and Francis, 1990.



- [DC91] M. Maingenaud D. Calcinelli. The Management of the Ambiguities in a Graphical Query Language for GIS. In *Proc. 2nd Symposium Spatial Database Systems*, pages 99–118. Springer Verlag Lecture Notes in Computer Science 525, 1991.
- [DDY93] J. Patel D. DeWitt and J. Yu. Paradise - A Parallel Geographic Information System. In *Proc. ACM Workshop on Advances in Geographic Information Systems*, pages X–Y, 1993.
- [DF92] J. Doerschler and H. Freeman. A Rule-based System for Dense Map Name Placement. *Communications of the ACM*, 35(1):68–79, 1992.
- [DGM95] E. Dias, S. Granado, and G. Magalhães. Uso de Versões na Garantia de Consistência em Ambientes Mistos de Projeto e Operação. In *Anais, X Simpósio Brasileiro de Bancos de Dados*, pages 321–334, 1995.
- [DRS93a] B. David, L. Raynal, and G. Schorter. Geo2: Object Oriented Contribution for a Geographic DBMS? In *Proc. 4th International Conference Database and Expert Systems Applications*, pages 377–383, 1993.
- [DRS93b] B. David, L. Raynal, and G. Schorter. Geo2: Why Objects in a Geographical DBMS? In *Proc 3rd International Symposium Large Spatial Databases*, 1993.
- [DS93] F.W. Davis and D.S. Simonett. GIS and Remote Sensing. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume I*, pages 191–213. John Wiley and Sons, 1993.
- [ea92] M. Goodchild et al. Integrating GIS and Spatial Data Analysis: Problems and Possibilities. *International Journal of Geographical Information Systems*, 6(5):407–424, 1992.
- [ea93] A. Mendelzon et al. Declarative Database Visualization: Recent Papers from the Hy+/GraphLog Project . Technical Report CSRI-285, University of Toronto, junho 1993.
- [ea94] D. DeWitt et alii. Client-Server Paradise. In *Proc. 20th VLDB Conference*, pages 558–569, 1994.
- [EF88] M. Egenhofer and A. Frank. Designing Object-oriented Query Languages for GIS: Human Interface Aspects. In *Proc 3rd International Symposium on Spatial Data Handling*, pages 79–98, 1988.
- [EF91] M. Egenhofer and R. Franzosa. Point-Set Topological Spatial Relations. *International Journal of Geographical Information Systems*, 5(2):161–174, 1991.
- [Ege89] M. Egenhofer. *Spatial Query Languages*. PhD thesis, University of Maine, maio 1989.

- [Ege92] M. Egenhofer. Why not SQL! *International Journal of Geographical Information Systems*, 6(2):71–86, 1992.
- [Ege94] M. Egenhofer. Spatial SQL: A Query and Presentation Language. *IEEE Transactions on Knowledge and Data Engineering*, 6(1):86–95, 1994.
- [Ege95] M. Egenhofer. Object-oriented GISs: The Principles. Technical report, NCGIA, fevereiro 1995.
- [EH90] M. Egenhofer and J. Herring. A Mathematical Framework for the Definition of Topological Relationships. In *Proc. 4th International Symposium on Spatial Data Handling*, pages 803–813, 1990.
- [EMM94] M. Elkington, R. Meyer, and G. McConaughy. Defining the Architectural Development of EOSDIS to Facilitate Extension to a Wider Data Information System. Technical Report 194-00131, Hughes Applied Information Systems, 1994.
- [EW94] A. Endal and S. Wharton. ECS Science Requirements Summary - White Paper. Technical Report FB9402V2, Hughes Applied Information Systems, Inc., março 1994.
- [FAB<sup>+</sup>95] J. Frew, M. Aurand, B. Battenfield, L. Carver, P. Chang, R. Ellis, C. Fisher, M. Gardner, M. Goodchild, G. Hajic, M. Larsgaard, K. Park, M. Probert, T. Smith, and Q. Zheng. The Alexandria Rapid Prototype: Building a Digital Library for Spatial Information. In *ADL'95 - A Forum on Research and Technology Advances in Digital Libraries*, 1995.
- [FCF92] A. Frank, I. Campari, and U. Formentini, editors. *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*. Lecture Notes in Computer Science 639. Springer Verlag, 1992.
- [Fed93] K. Fedra. GIS and Environmental Modelling. In M. Goodchild, B. Parks, and L. Steyaert, editors, *Environmental Modelling with GIS*, pages 35–47. Oxford University Press, 1993.
- [Feu93] M. Feutchwanger. *Towards a Geographic Semantic Database Model*. PhD thesis, Simon Fraser University, 1993.
- [FG90] A. Frank and M. Goodchild. Two Perspectives on Geographical Data Modelling. Technical Report 90-11, National Center for Geographic Information and Analysis, 1990.
- [FG94] C. Felgueiras and M. Goodchild. An Incremental Constrained Delaunay Triangulation. Technical report, NCGIA, 1994.
- [Fil93] A. Queiroz Filho. Ortofoto Digital para Atualização Cartográfica em SIG. In *IV Conferência Latinoamericana sobre Sistemas de Informação Geográfica*, pages 47–56, 1993.

- [FK93] C. Faloutsos and I. Kamel. High Performance R-trees. *IEEE Data Engineering Bulletin*, pages 28–33, 1993.
- [FM91] A. Frank and D. Mark. Language Issues for GIS. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume I*, pages 147–161. John Wiley and Sons, 1991.
- [FR93] R. Fernandez and M. Rusinkiewicz. A Conceptual Design of a Soil Database for a GIS. *International Journal of Geographical Information Systems*, 7(6):525–540, 1993.
- [FR94] S. Fotherington and P. Rogerson, editors. *Spatial Analysis and GIS*. Taylor and Francis, 1994.
- [Fre75] J. Freeman. The Modelling of Spatial Relations. *Computer Graphics and Image Processing*, 4:156–171, 1975.
- [Fre87] M. Freeston. The BANG File: A New Kind of Grid File. In *Proc. of the ACM SIGMOD Conference*, pages 260–269, 1987.
- [FT93] P. Franciosa and M. Talamo. ESPRIT Project EP6881 - AMUSING. *IEEE Data Engineering Bulletin*, pages 46–50, 1993.
- [Gar93] K. Gardels. The Sequoia Model of Geographic Information. In *Proc. of the GIS '93 Symposium*, 1993.
- [GB90] O. Gunther and A. Buchman. Research Issues in Spatial Databases. *ACM Sigmod Record*, 19(4):61–68, 1990.
- [GEOG94] J. Gonçalves, M. Erhardt, F. Obata, and S. Granado. Automação de Cadastros de Rede Externa. In *Proc. I Encontro de Qualidade de Redes de Telecomunicações e de Equipamento Terminais*, 1994.
- [Goh89] P-C. Goh. A Graphic Query Language for Cartographic and Land Information Systems. *International Journal of Geographical Information Systems*, 3(3):245–256, 1989.
- [Goo87] M. Goodchild. A Spatial Analytical Perspective on Geographical Information Systems. *International Journal of Geographical Information Systems*, 1(4):327–334, 1987.
- [Goo91] M. Goodchild. Spatial Analysis with GIS: Problems and Prospects. In *Proc GIS/LIS 91*, volume 1, pages 40–48, 1991.
- [Goo92] M. Goodchild. Geographical Data Modelling. *Computers and Geosciences*, 18(4):401–408, 1992.

- [Goo93] M. Goodchild. A Perspective on the State of Environmental Simulation. In M. Goodchild, B. Parks, and L. Steyaert, editors, *Environmental Modelling with GIS*, pages 8–15. Oxford University Press, 1993.
- [Gou93] M. Gould. *Human Factors in Geographic Information Systems*, chapter Two Views of the User Interface, pages 101–110. Belhaven Press, 1993.
- [GPS93] M. Goodchild, B. Parks, and L. Steyaert, editors. *Environmental Modelling with GIS*. Oxford University Press, 1993.
- [GR93] O. Gunther and W-F Riekert. The Design of GODOT: an Object-oriented Geographic Information System. *IEEE Data Engineering Bulletin*, pages 4–9, 1993.
- [GR95] V. Gudivada and V. Raghavan. Content-based Image Retrieval Systems. *IEEE Computer*, 28(9):18–22, 1995.
- [Gra93] G. Graefe. Query Evaluation Techniques for Large Databases. *ACM Computing Surveys*, 25(2):73–170, 1993.
- [Gut84] A. Guttman. R-Trees: A Dynamic Index Structure for Spatial Searching. In *Proc. of the ACM SIGMOD Conference*, pages 47–56, junho 1984.
- [Gut88] R. Guting. Geo-relational Algebra: a Model and Query Language for Geometric Database Systems. In *Proc. 1st EDBT Conference*, pages 506–527, 1988.
- [Gut94] R. Guting. An Introduction to Spatial Database Systems. *VLDB Journal*, 3(4), 1994.
- [GV93] J. Gomes and L. Velho. Modelagem em Computação Gráfica. In *VI Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens*, 1993.
- [HC91] L. Haas and W. Cody. Exploiting Extensible DBMS in Integrated Geographic Information Systems. In *Proc. 2nd Symposium Spatial Database Systems*, pages 423–449. Springer Verlag Lecture Notes in Computer Science 525, 1991.
- [HCS93] S. Hirtle, T. Crippa, and M. Spring. The Cognitive Structure of Space: an Analysis of Temporal Sequences. In *Proc. European Conference Spatial Information Theory - COSIT*, pages 177–189. Springer Verlag Lecture Notes in Computer Science 716, 1993.
- [Her92] J. Herring. TIGRIS: A Data Model for an Object-oriented Geographic Information System. *Computers and Geosciences: An international journal*, 18(4):443–451, 1992.
- [Her94] D. Hernandez. *Qualitative Representation of Spatial Knowledge*, volume 804 of *Lecture Notes in Artificial Intelligence*. Springer Verlag, 1994.
- [HFC93] A. Hemerly, A. Furtado, and M. Casanova. Towards Cooperativeness in Geographic Databases. In *Proc 4th International DEXA Conf.*, pages 373–376, 1993.

- [HS92a] W. Hall and D. Simmons. An Open Model for Hypermedia and Its Application to Geographical Information Systems. *Computer Graphics Forum*, 2(3), 1992.
- [HS92b] E. Hoel and H. Samet. A Qualitative Comparison Study of Data Structures for Large Line Segment Databases. In *Proc. of the ACM SIGMOD Conference*, pages 205–214, 1992.
- [HS94] E. Hoel and H. Samet. Performance of Data-Parallel Spatial Operations. In *Proc 20th VLDB Conference*, 1994.
- [Int90] Intergraph. *MGE - The Modular GIS Environment*, 1990.
- [Jai93] R. Jain. NSF Workshop on Visual Information Management Systems. *ACM Sigmod Record*, 22(3):57–76, 1993.
- [JBG93] W. McKenna J. Blakeley and G. Graefe. Experiences Building the Open OODB Optimizer. In *Proc. of the ACM SIGMOD Conference*, pages 287–296, maio 1993.
- [JEM93] JEM. Environmental Assessment and the Application of Expert Systems: an Overview. *Journal of Environmental Management*, 39:27–38, 1993.
- [KBS93] H. Kriegel, T. Brinkhoff, and R. Schneider. Efficient Query Processing in Geographic Database Systems. *IEEE Data Engineering Bulletin*, pages 10–15, 1993.
- [KF92] I. Kamel and C. Faloutsos. Parallel R-Trees. *ACM Sigmod Record*, 21(2), 1992.
- [KHFM93] H. Kriegel, S. Heep, A. Fahldiek, and N. Mysliwitz. Query Processing of Geometric Objects with Free Form Boundaries in Spatial Databases. In *Proc 4th International DEXA Conf.*, pages 349–360, 1993.
- [Kli94] A. Kliskey. A Comparative Analysis of Approaches to Wilderness Perception Mapping. *Journal of Environmental Management*, 41:199–236, 1994.
- [Koi95] H. Koike. Fractal Views – a Fractal-based Method for Controlling Information Display. *ACM Trans. on Information Systems*, 13(3):305–323, 1995.
- [Kri91] H. Kriegel. Query Processing in Spatial Database Systems. In *Proc. 2nd Symposium Spatial Database Systems*, pages 172–191. Springer Verlag Lecture Notes in Computer Science 525, 1991.
- [KT92] Z. Kemp and R. Thearle. Modelling Relationships in Spatial Databases. In *Proc 5th International Symposium on Spatial Data Handling*, pages 313–322, 1992. Volume 1.
- [Kuh91] W. Kuhn. Are Displays Maps or Views? In *Auto-Carto*, volume 10, pages 588–598, 1991.

- [Kuh93] W. Kuhn. Metaphors Create Theories for Users. In *Proc. European Conference in Spatial Information Theory*, pages 366–376. Springer Verlag Lecture Notes in Computer Science 716, 1993.
- [Lan93a] G. Langran. Issues of Implementing a Spatiotemporal System. *International Journal of Geographical Information Systems*, 7(4):305–314, 1993.
- [Lan93b] G. Langran. *Time in Geographic Information Systems*. Taylor & Francis, 1993.
- [LG93] J. Lee and G. Grinstein, editors. *Database Issues for Data Visualization - Proc. IEEE Visualization Workshop*. Springer Verlag, 1993. Lecture Notes in Computer Science 871.
- [LHM94] P. Longley, G. Higgs, and D. Martin. The Predictive use of GIS to Model Property Valuations. *International Journal of Geographical Information Systems*, 8(2):217–236, 1994.
- [LL93] Y. Leung and K. Leung. An Intelligent Expert System Shell for Knowledge-based Geographical Information Systems: 1 - the tools. *International Journal of Geographical Information Systems*, 7(1):189–200, 1993.
- [LMR90] W. Litwin, L. Mark, and N. Roussopoulos. Interoperability of Multiple Autonomous Databases. *ACM Computing Surveys*, 22(3):297–293, 1990.
- [LS89] D. Lomet and B. Salzberg. A Robust Multi-Attribute Search Structure. In *Proc. 5th International Conf. on Data Engineering*, pages 296–304, 1989.
- [LS92] M. Lindholm and T. Sarjakoski. User Models and Information Theory in the Design of a Query Interface for GIS. In *Proc International Conference on GIS - From Space to Territory: Theories and Methods of Spatial Reasoning*, pages 328–347. Springer Verlag Lecture Notes in Computer Science 639, 1992.
- [M. 94] M. Carey and D. Dewitt and M. Franklin and N. Hall and M. MacAuliffe and J. Naughton and D. Schuh and M. Solomon and C. Tan and O. Tsatalos and S. White and M. Zwillig. Shoring up Persistent Objects. In *Proc. of the ACM SIGMOD Conference*, pages 383–394, 1994.
- [Mag94] G. Magalhães. The Development of Open Systems for Engineering Applications. In *Proc International Automated Mapping/Facility Management Conference – AM/FM International*, 1994.
- [Mah93] R. Mahoney. GIS and Utilities. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume II*, pages 101–114. John Wiley and Sons, 1993.
- [MCD94] M. Mediano, M. Casanova, and M. Dreux. V-Trees: A Storage Method for Spatial Data. In *Proc 20th VLDB Conference*, pages 321–330, 1994.

- [MCS<sup>+</sup>94] C. Medeiros, G. Camara, R. Souza, U. Freitas, M. Casanova, and A. Hemerly. A Model to Manipulate Objects and Cultivate Fields. In *Proc. ACM Workshop on Advances in Geographic Information Systems*, pages 30–37, 1994.
- [MD91] D. Maguire and E. Dangermond. The Functionality of GIS. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume I*, pages 319–335. John Wiley and Sons, 1991.
- [Med95] M. Mediano. Métodos de Acesso a Mapas com Agrupamento. Technical report, Depto. de Informatica, Puc-Rio, 1995.
- [MEdF94] E. Clementini M. Egenhofer and P. di Felice. Topological Relationships between Regions with Holes. *International Journal of Geographical Information Systems*, 8(2):129–142, 1994.
- [MGR93a] D. Maguire, M. Goodchild, and D. Rhind, editors. *Geographical Information Systems - volume I*. John Wiley and Sons, 2 edition, 1993.
- [MGR93b] D. Maguire, M. Goodchild, and D. Rhind, editors. *Geographical Information Systems - volume II - Applications*. John Wiley and Sons, 2 edition, 1993.
- [MGS<sup>+</sup>94] G. Magalhaes, A. Giglioni, C. Santos, D. Teijero, and E. Argondizio. Especificação Técnica de Conversão de Dados Proposta da Telebras – Projeto SAGRE. In *Anais GIS-Brasil*, pages 43–52, 1994.
- [MJ93] C. Medeiros and G. Jomier. Managing Alternatives and Data Evolution in GIS. In *Proc. ACM/ISCA Workshop on Advances in Geographic Information Systems*, pages 34–37, 1993.
- [MMS93] P. Milne, S. Milton, and J. Smith. Geographical Object-oriented Databases: a Case Study. *International Journal of Geographical Information Systems*, 7:39–56, 1993.
- [Mon91] M. Monmonier. *How to Lie with Maps*. University of Chicago Press, 1991.
- [Mor66] G. Morton. A Computer Oriented Geodetic Data Base and a New Technique in File Sequencing. Technical report, IBM Canada, 1966.
- [Mor92] S. Morehouse. The ARC/INFO geographic information system. *Computers and Geosciences: An international journal*, 18(4):435–443, 1992.
- [Mou93] H. Mounsey. Multisource, Multinational Environmental GIS: lessons learnt from CORINE. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume II*, pages 185–200. John Wiley and Sons, 1993.
- [Mue86] P. Muehercke. *Map Use*. JP Publications, 1986.
- [MV89] N. Mascarenhas and F. Velasco. *Processamento Digital de Imagens*. Editorial Kapelusz S.A., 1989.

- [MW92] J. Morrison and K. Wortman. Cartography and Geographic Information Systems - Special Issue: Implementing the Spatial Data Transfer Standard. *American Congress on Surveying and Mapping*, 19(5), 1992.
- [NCG89] NCGIA. The Research Plan of the National Center for Geographic Information and Analysis. *International Journal of Geographical Information Systems*, 3(2):117–136, 1989.
- [NHS84] J. Nievergelt, H. Hinterberger, and K. Sevcik. The Grid File: An Adaptable, Symmetric Multikey File Structure. *ACM Transactions on Database Systems*, 9:38–71, 1984.
- [NS91] J. Noll and W. Scacchi. Integrating Diverse Information Repositories: A Distributed Hypertext Approach. *IEEE Computer*, 15(12):38–45, 1991.
- [NTE92] R. Newell, D. Theriault, and M. Easterfieldy. Temporal GIS - Modeling the Evolution of Spatial Data in Time. *Computers and Geosciences: An international journal*, 18(4):427–434, 1992.
- [Nye93] T. Nyerges. Understanding the Scope of GIS: its Relationship to Environmental Modeling. In M. Goodchild, B. Parks, and L. Steyaert, editors, *Environmental Modelling with GIS*, pages 8–15. Oxford University Press, 1993.
- [Obj94] Object Management Group, Framingham, Massachussets. *The Common Object Request Broker: Architecture and Specification*, 1.2 edition, 1994.
- [OCM95] J. Oliveira, C. Cunha, and G. Magalhaes. Modelo de Objetos para Construção de Interfaces Visuais Dinâmicas. In *Anais, IX Simpósio Brasileiro de Engenharia de Software*, pages 143–158, 1995.
- [Oli95a] C. Oliveira. Cartas Náuticas Eletrônicas: Operações e Estruturas de Dados. Master's thesis, DCC-UNICAMP, dezembro 1995.
- [Oli95b] J. Oliveira. User Interface Issues in Geographic Information Systems. Technical Report 18-95, DCC - UNICAMP, dezembro 1995.
- [OM95] J. Oliveira and C. Medeiros. A Direct Manipulation User Interface for Querying Geographic Databases. In *Proc. 2nd International ADB Conference*, 1995.
- [Ooi90] B. Ooi. *Efficient Query Processing in Geographic Information Systems*. Springer Verlag Lecture Notes in Computer Science, 1990.
- [Ore86] J. Orenstein. Spatial Query Processing in an Object-Oriented Database System. In *Proc. of the ACM SIGMOD Conference*, pages 326–336, 1986.
- [Ore89] J. Orenstein. Redundancy in Spatial Databases. In *Proc. of the ACM SIGMOD Conference*, pages 294–305, 1989.



- [Par94] E. Paredes. *Sistema de Informação Geográfica – Princípios e Aplicações (Geoprocessamento)*. Editora Érica, 1994.
- [Pet95] M. Petre. Why Looking Isn't Always Seeing: Readership Skills and Graphical Programming. *Communications of the ACM*, 38(6):33–44, 1995.
- [Peu93] D. Peuquet. What, Where and When - a Conceptual Basis for Design of Spatiotemporal GIS Databases. In *Proc. ACM/ISCA Workshop on Advances in Geographic Information Systems*, pages 117–122, 1993.
- [Peu94] D. Peuquet. It's about time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems. In *Annals of the Association of American Geographers*, 1994.
- [PM90] D. Peuquet and D. Marble, editors. *Introductory Readings in Geographic Information Systems*. Taylor and Francis, 1990.
- [PMP93] N. Pissinou, K. Makki, and E. Park. Towards the Design and Development of a New Architecture for Geographic Information Systems. In *Proc. 2nd International Conference on Information and Knowledge Management- CIKM*, pages 565–573, 1993.
- [PS90] C. Pietrobon and A. Staa. Controle de Consistência em um Ambiente de Desenvolvimento de Software Distribuído com Objetos Imutáveis. In *Anais do IV Simpósio Brasileiro de Engenharia de Software*, 1990.
- [PST93] B. Pagel, H. Six, and H. Toben. The Transformation Technique for Spatial Objects Revisited. In *Proc. 3rd International Symposium on Large Spatial Databases*, pages 73–88, 1993.
- [PTO95] A. Prezzoto, D. Teijero, and F. Obata. Telecommunications Outside Plant Design Automation. In *Proc. II International Congress on Information Engineering (ICIE'95)*, pages 14–15, 1995.
- [Ram94] M. Ramirez. Sistemas Gerenciadores de Banco de Dados para Geoprocessamento. Master's thesis, COPPE-UFRJ, abril 1994.
- [RFS88] N. Roussopoulos, C. Faloutsos, and T. Sellis. An Efficient Pictorial Database System for PSQL. *IEEE Transactions on Software Engineering*, SE-14(5):639–650, 1988.
- [Rhi93] D. Rhind. Counting the People: the Role of GIS. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume II*, pages 127–137. John Wiley and Sons, 1993.
- [Rig95] P. Rigaux. *Interfaces Graphiques pour Bases de Données Spatiales: Application à la Représentation Multiple*. PhD thesis, CEDRIC - Conservatoire National des Arts et Métiers, 1995.

- [RK91] J. Raper and B. Kelk. Three Dimensional GIS. In D. J. Maguire, M. F. Goodchild, and D. W. Rhind, editors, *Geographical Information Systems - volume I*, pages 299–317. John Wiley and Sons, 1991.
- [RM92] J. Raper and D. Maguire. Design Models and Functionality in GIS. *Computers and Geosciences*, 18(4):387–400, 1992.
- [RND<sup>+</sup>95] B. Raoult, B. Norris, J. Daabeck, R. Souza, and G. Camara. Distributed Architectures for Environmental Visualization Systems. In *Anais, VII Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens*, pages 249–256, 1995.
- [Rob81] J. Robinson. The KDB-Tree: A Search Structure for Large Multidimensional Dynamic Indexes. In *Proc. of the ACM SIGMOD Conference*, pages 10–18, 1981.
- [Roc95] J. Rocha. Gerenciamento de Operações Agrícolas em Sistema de Informações Geo-referenciadas. *Cadernos de Informática Georeferenciada*, 1(1), 1995.
- [Rot91] D. Rotem. Spatial Join Indices. In *Proc. IEEE Data Engineering Conference*, pages 500–509, 1991.
- [RSM78] A. Robinson, R. Sale, and J. Morrison. *Elements of Cartography*. John Wiley & Sons, 1978.
- [Rum91] J. Rumbaugh. *Object-Oriented Modeling and Design*. Prentice-Hall, 1991.
- [SA93] R. Subramanian and N. Adam. The Design and Implementation of an Expert Object-Oriented Geographic Information System. In *Proc. 2nd International Conference on Information and Knowledge Management- CIKM*, pages 537–546, 1993.
- [SAI] SAIF. <http://www.wimsey.com/infosafe/saif/saifHome.html>.
- [Sam89] H. Samet. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley, 1989.
- [SCR93] L. Soares, M. Casanova, and N. Rodriguez. Um Modelo Conceitual Hipermissão com Nós de Composição e Controle de Versões. In *Anais VII Simpósio Brasileiro de Engenharia de Software*, pages 365–381, 1993.
- [SDT] SDTS. <ftp://sdts.er.usgs.gov>.
- [SFGM93] M. Stonebraker, J. Frew, K. Gardels, and J. Meredith. The Sequoia 2000 Benchmark. In *Proc. of the ACM SIGMOD Conference*, pages 2–11, 1993.
- [SH91] P. Svensson and Z. Huang. Geo-Sal: A Query Language for Spatial Data Analysis. In *Proc. 2nd Symposium Spatial Database Systems*, pages 119–140. Springer Verlag Lecture Notes in Computer Science 525, 1991.

- [SK90] B. Seeger and H-P. Kriegel. The Buddy-Tree: an Efficient and Robust Access Method for Spatial Database Systems. In *Proc. 16th VLDB Conference*, pages 590–601, 1990.
- [SL90] A. Sheth and J. Larson. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 22(3):183–236, 1990.
- [SMJ94] E. Strand, R. Mehta, and R. Jairam. Applications Thrive on Open Systems Standards. *Standard View - ACM Perspectives on Standardization*, 2(3):148–154, 1994.
- [SMSE87] T. Smith, S. Menon, J. Star, and J. Estes. Requirements and Principles for the Implementation and Construction of Large-scale Geographic Information Systems. *International Journal of Geographical Information Systems*, 1(1):13–31, 1987.
- [Sno92] R. Snodgrass. Temporal Databases. In *Proc International Conference on GIS - From Space to Territory: Theories and Methods of Spatial Reasoning*, pages 22–63. Springer Verlag Lecture Notes in Computer Science 639, 1992.
- [SPR93] SPRING - Manual do Usuário. Versão Preliminar, fevereiro 1993.
- [SQ93] R. Silva and J. Quintanilha. SIG na Gestão Municipal. In *IV Conferência Latino-americana sobre Sistemas de Informação Geográfica*, pages 353–364, 1993.
- [SRD<sup>+</sup>91] A. Skidmore, P. Ryan, W. Dawes, D. Short, and E. O’Loughlin. Use of an Expert System to Map Forest Soils from a Geographical Information System. *International Journal of Geographical Information Systems*, 5(4):431–446, 1991.
- [SRF87] T. Sellis, N. Roussopoulos, and C. Faloutsos. The  $R^+$ -Tree: A Dynamic Index for Multi-Dimensional Objects. In *Proc. 13th VLDB Conference*, pages 507–518, 1987.
- [SS94] R. Snodgrass and M. Soo. Bibliography on Spatiotemporal Databases. *International Journal of Geographical Information Systems*, 8(1):95–103, 1994.
- [SSB95] L. Shklar, K. Shah, and C. Basu. Putting Legacy Data on the Web: A Repository Definition Language. In *Third International WWW Conference*, 1995.
- [SSKS95] L. Shklar, A. Sheth, V. Kashyap, and K. Shah. InfoHarness: Use of Automatically Generated Metadata for Search and Retrieval of Heterogeneous Information. In *Proc. CAiSE’95*. Springer Verlag Lecture Notes in Computer Science 932, 1995.
- [STMS94] L. Shklar, S. Thatte, H. Marcus, and A. Sheth. The InfoHarness Information Integration Platform. In *Second International WWW Conference*, 1994.
- [Sto93] M. Stonebraker. The SEQUOIA 2000 Project. *IEEE Data Engineering Bulletin*, pages 24–28, 1993.

- [Sur94] Surveys and Resource Mapping Branch - Ministry of Environment, Lands and Parks, British Columbia - Canadá. *Spatial Archive and Interchange Format: Formal Definition*, 3.1 edition, 1994. Reference Series.
- [SV92] M. Scholl and A. Voisard. Geographic Applications – an Experience with O2. In F. Bancilhon, C. Delobel, and P. Kanellakis, editors, *Building an Object-oriented System – the Story of O2*. Morgan Kaufmann, 1992.
- [Tam82] M. Tamminen. The Extendible Cell Method for Closest Point Problems. *BIT*, 22:27–41, 1982.
- [Thi93] J. Thieman. The International Directory Network and Connected Data Information System for Research in the Earth and Space Sciences. In *Proc. of the ACM SIGMOD Conference*, pages 475–478, 1993.
- [TMC92] A. Teixeira, E. Moretti, and A. Cristofolletti. *Introdução aos Sistemas de Informação Geográfica*. Edição de Autor, 1992.
- [Toc95] C. Tocantins. Armazenamento e Manipulação de Mapas Vetoriais em uma Arquitetura Paralela. Master's thesis, PUC-RJ, agosto 1995.
- [Tom90] D. Tomlin. *Geographic Information Systems and Cartographic Modeling*. Prentice Hall, New York, 1990.
- [Tom94] H. Tom. The Geographic Systems (GIS) Standards Infrastructure. *StandardView*, 2(3), 1994.
- [Tos94] N. Tosta. Standards to Support the National Spatial Data Infrastructure. *StandardView - ACM Perspectives on Standardization*, 2(3):143–147, 1994.
- [Tsa93] V. Tsai. Delaunay Triangulations in TIN Creation: an Overview and a Linear time Algorithm. *International Journal of Geographical Information Systems*, 7(6):501–524, 1993.
- [Tve93] B. Tversky. Cognitive Maps, Cognitive Collages, and Spatial Mental Models. In *European Conference Spatial Information Theory - COSIT*, pages 14–24. Springer Verlag Lecture Notes in Computer Science 716, 1993.
- [Uni93] University of California. *Postgres 4.1 Reference Manual*, 1993.
- [U.S92] U.S. Army Construction Engineering Research Laboratory. *GRASS 4.0 Programmer's Manual*, 1992.
- [vEU89] J. van Eck and M. Uffer. A Presentation of System 9. *Photogrammetry and Land Information Systems*, pages 139–178, março 1989.
- [VM90] E. Victorelli and G. Magalhães. MVC: Um Modelo para Controle de Versões e Configurações. In *Anais V Simpósio Brasileiro de Engenharia de Software*, 1990.

- [VMS<sup>+</sup>93] K. Vadaparty, R. Molmen, H. Salem, P. Whiting, and S. Naqvi. Towards a Fully Sheetless GIS with Incremental Querying. In *Proc. ACM/ISCA Workshop on Advances in Geographic Information Systems*, pages 94–99, 1993.
- [Voi91] A. Voisard. Towards a Toolbox for Geographic User Interfaces. In *Proc. Second Symposium on Spatial Database Systems*, pages 75–97. Springer Verlag Lecture Notes in Computer Science 525, 1991.
- [Voi94] A. Voisard. Designing and Integrating User Interfaces of Geographic Database Applications. In *Proc. ACM Workshop on Advanced Visual Interfaces*, pages 133–142, 1994.
- [vOV91] P. von Oosterom and T. Vijlbrief. Building a GIS on Top of the Open DBMS POSTGRES. In *Proc European GIS Conference*, 1991.
- [VvO91] T. Vijlbrief and P. von Oosterom. The GEO++ System: an Extensible GIS. In *Proc European GIS Conference*, 1991.
- [VvO92] T. Vijlbrief and P. van Oosterom. The GEO++ System: an Extensible GIS. In *Proc. 5th International Symposium on Spatial Data Handling*, pages 40–50, 1992.
- [Wan94] F. Wang. Towards a Natural Language User Interface: an Approach of fuzzy Query. *International Journal of Geographical Information Systems*, 8(2):143–162, 1994.
- [Web90] C. Webster. Rule-based Spatial Search. *International Journal of Geographical Information Systems*, 4(3):241–260, 1990.
- [WH92] T. Waugh and S. Hopkins. An Algorithm for Polygon Overlay Using Cooperative Processing. *International Journal of Geographical Information Systems*, 6(6), 1992.
- [Whi93] M. White. Car Navigation Systems. In D. Maguire, M. Goodchild, and D. Rhind, editors, *Geographical Information Systems - volume II*, pages 115–125. John Wiley and Sons, 1993.
- [WHM90] M. Worboys, H. Hearnshaw, and D. Maguire. Object-oriented Data Modelling for Spatial Databases. *International Journal of Geographical Information Systems*, 4(4):369–384, 1990.
- [Wol83] P. Wolf. *Elements of Photogrammetry*. McGraw Hill Book Co., 1983.
- [Woo93] M. Wood. *Human Factors in Geographic Information Systems*, chapter Interfacing with Maps, pages 111–123. Belhaven Press, 1993.
- [Wor94a] M. Worboys. Object Oriented Approaches to Georeferenced Information. *International Journal of Geographical Information Systems*, 8(4):385–400, 1994.
- [Wor94b] L. Worral. Justifying Investment in GIS: a Local Government Perspective. *International Journal of Geographical Information Systems*, 6(8):545–566, 1994.

- [WSS90] C. Woodcock, C. Shaw, and B. Shaw. Comments on Selecting a Geographic Information System for Environmental Management. *Environmental Management*, 14(3):307–315, 1990.
- [ZFS89] E. Zube, S. Freedman, and D. Simcox. Landscape Change: Perceptions and Physical Measures. *Environmental Management*, 13(5):639–644, 1989.
- [ZM92] F. Zhan and D. Mark. Object-Oriented Spatial Knowledge Representation and Processing: Formalization of Core Classes and their Relationships. In *Proc 5th International Symposium on Spatial Data Handling*, pages 662–671, 1992. Volume 2.