Caracterização de sistemas de banco de dados espaciais para análise de desempenho

Walter Paulo Costenaro

Dissertação de Mestrado

Instituto de Computação
Universidade Estadual de Campinas

Caracterização de sistemas de banco de dados espaciais para análise de desempenho

Walter Paulo Costenaro[1]

18 de junho 1997

Banca Examinadora:

- Geovane Cayres Magalhães
  Instituto de Computação - UNICAMP (Orientador)
- Claudia Maria Bauzer Medeiros
  Instituto de Computação - UNICAMP
- Jansle Vieira Rocha
  Faculdade de Engenharia Agrícola - UNICAMP
- Cecília Mary Fischer Rubira (Suplente)
  Instituto de Computação - UNICAMP

[1]Bacharel em Ciência da Computação pela Universidade Estadual de Maringá - Paraná.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DO IMECC DA UNICAMP

Costenaro, Walter Paulo

C824c Caracterização de sistemas de banco de dados espaciais para análise de desempenho / Walter Paulo Costenaro -- Campinas, [S.P. :s.n.], 1997.

Orientador : Geovane Cayres Magalhães

Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Computação.

1. Sistemas de informações geográficas. 2. Banco de dados relacionais. 3. Planejamento urbano. I. Magalhães, Geovane Cayres. II. Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Caracterização de sistemas de banco de dados espaciais para análise de desempenho

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Walter Paulo Costenaro e aprovada pela Banca Examinadora.

Campinas, 18 de junho de 1997.

Prof. Dr. Geovane C. Magalhães
Instituto de Computação - Unicamp
(Orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Tese de Mestrado defendida e aprovada em 18 de junho de 1997 pela Banca Examinadora composta pelos Professores Doutores

Prof. Dr. Jansle Vieira Rocha

Profª. Drª. Claudia Maria Bauzer Medeiros

Prof. Dr. Geovane Cayres Magalhães

Prefácio

Nesta dissertação, a área de sistemas de banco de dados espaciais é enfocada utilizando a técnica de *benchmark* para análise de desempenho. Esta técnica requer o monitoramento de sistemas de banco de dados utilizando banco de dados e carga de trabalho (transações) reais ou sintéticos. Vários fatores apontam como ideal o uso de dados sintéticos que melhor representem situações normalmente encontradas em aplicações práticas (reais). Esta dissertação utiliza-se de dados reais (convencionais e não convencionais) de uma aplicação de gerência de planta externa de telecomunicações para validar e melhorar técnicas de caracterização de sistemas de banco de dados espaciais e tirar conclusões úteis para estudos de desempenho de banco de dados em geral.

Abstract

In this thesis, the area of spatial database systems is approached using the benchmark technique for performance analysis. This technique requires the monitoring of a database system using a real or synthetic database and workload (transactions). The ideal situation is the use of synthetic data that better resembles the situations found in real applications. This thesis uses real data (spatial and non-spatial) of a telecomunications outside plant management system to validate and enhance techniques to provide more realistic synthetic data and workload and to derive conclusions useful for performance studies of database system in general.

Em memória de meus pais, Walter e Mercedes.
Aos meus irmãos, Vanderlei e Érika.
Aos familiares, amigos e aqueles que se fizeram amigos para que eu chegasse até aqui.
A Ele ...

Agradecimentos

Aos meus irmãos Vanderlei e Érika pelo incentivo e carinho existente entre nós.
Ao meu orientador Prof. Geovane, pela confiança, incentivo e ensinamentos proporcionados durante esta caminhada.
Ao grupo de discussão de banco de dados, em especial à professora Claudia.
Aos amigos do Grupo Fogança, companheiros de um período de grandes sonhos e realizações.
Aos amigos de velhos tempos: Ângela, Carla, Carlos, Edmar, Edson, Lâmia, Lucila, Marta, Milian, Mônica e Silvia que me ensinaram o valor de uma grande amizade.
Aos amigos do vôlei, das turmas de mestrado de 93, 96 e em especial à turma de 95.
Aos amigos Ana, Carrilho, Cláudio, Clauber, Claudir, Denilson, Edmundo, Elba, Elvira, Hirano, Jr, Jaqueline, Lúcia, Luciano, Luciclélia, Marcos, Natal, Osvaldo, Orestes, Patrícia, Pedro, Sávio e Silvia, sempre presentes e incentivando durante todo este período de convivência.
Aos funcionários do IC pelo carinho com que sempre me atenderam.
Aos companheiros do CPqD-Telebrás que mantive contato durante a realização deste trabalho. Vocês fortificaram em mim o espírito de companheirismo no desenvolvimento de um projeto.
Ao CNPq, FAPESP e ao CPqD Telebrás - Telecomunicações Brasileiras S/A, pelo suporte a este projeto.
Este trabalho foi desenvolvido parcialmente dentro do projeto CNPq PROTEM/CC-GEOTEC.

Conteúdo

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

Sistemas de informações geográficas (SIG) têm sido usados cada vez mais na construção de complexos sistemas de informações. A análise de desempenho destes sistemas pode trazer benefícios para o processo de escolha do SIG, para o ajuste de parâmetros que afetam o desempenho e, especialmente, para melhorar os algoritmos utilizados por estes sistemas. Uma técnica bastante utilizada para este tipo de análise de desempenho é a comumente conhecida como *benchmark*.

Esta dissertação faz uso dos dados de uma aplicação específica de sistemas de informações geográficas para validar e melhorar técnicas de caracterização de dados tendo em vista o uso de *benchmarks* na análise de desempenho.

1.1 Motivação

Com o advento da indústria de informática, as empresas passaram a oferecer produtos de *hardware* e *software* com as mesmas ou semelhantes funcionalidades. Diante desse contexto, mecanismos que pudessem medir e comparar estes produtos tornaram-se necessários. A análise de desempenho é um dos mecanismos mais utilizados para tal propósito.

A análise de desempenho consiste em medir a eficiência com a qual produtos efetuam uma determinada funcionalidade a partir de alguma medida escalar (tempo, por exempo).

Em relação às muitas áreas da ciência da computação na qual a análise de desempenho pode ser aplicada, merece destaque a análise de desempenho de SGBD.

Atualmente, embora as pesquisas em banco de dados estejam bem evoluídas com a utilização dos SGBDs em quase todos os ambientes computacionais, a variedade de áreas de aplicação onde esta tecnologia tem sido requisitada vem dirigindo as pesquisas para prover o melhor suporte às aplicações.

Por exemplo, nas empresas [1] que prestam serviço público de telecomunicações é necessário uma enorme quantidade de equipamentos (cabos, centrais telefônicas, armários, etc) distribuídos geograficamente em uma determinada área para suportar serviços de telecomunicações tais como telefonia, telex e comunicação de dados, entre outros. Além disso, os funcionários realizam inúmeras atividades que visam proporcionar serviços para os usuários bem como administrar as operações da companhia. Uma vez que o volume de informações gerado durante as atividades dessas empresas é considerável, sua administração deve dispor de um sistema de informações (SI) adequado para manipular, armazenar e distribuir estas informações entre as diferentes áreas da empresa de modo a garantir confiabilidade, integridade e disponibilidade das informações, assim como a redução de custos de geração e manutenção.

No contexto de aplicações geográficas, um sistema de informação geográfica - SIG é caracterizado como um *software* composto por vários subsistemas integrados, os quais são voltados à geração de mapas, com o auxílio de um sistema gerenciador de banco de dados (SGBD) não convencional, responsável por permitir o uso conjunto de uma enorme quantidade de dados espaciais e convencionais, através de estruturas de armazenamento de dados, linguagens e otimizadores de consultas específicos [Cif95].

O crescente desenvolvimento destes sistemas de informações geográficas tem fomentando a existência de SGBDs cada vez mais adequados a estas aplicações. O desempenho destes SGBDs exerce papel relevante para o sucesso comercial destas aplicações.

Entre os modelos de análise de desempenho encontrados, a técnica conhecida popularmente por *benchmark* de banco de dados é largamente empregada para se analisar o desempenho de sistemas de banco de dados. Genericamente, a técnica de *benchmark* quando aplicada a SGBD consiste na execução de um conjunto conhecido de transações, ou carga de trabalho como é comumente denominado, sobre um banco de dados também conhecido [Cif95, Per90, Gra91]. Um dos objetivos desta dissertação é caracterizar estes sistemas de bancos de dados o mais próximo possível de aplicações reais, possibilitando sua geração de maneira sintética.

Ainda, esta dissertação enfoca sistemas de banco de dados espaciais. Aplicações SIGs manipulam tanto dados convencionais como dados não convencionais. Em relação a dados convencionais manipulados pelos SGBDs, Rafles Pereira ([Per90]) propõe ampliar a abrangência do uso da técnica de *benchmark* para análise de sistemas de gerenciamento de bancos de dados através de uma Metodologia de *Benchmark* Especializada (MBE). Já Ricardo Ciferri ([Cif95]) propõe a carga de trabalho e a caracterização dos dados de um *benchmark* voltado à análise de desempenho de sistemas de informações geográficas, levando em consideração características especiais de aplicações que se utilizam de SIG.

Sendo assim, o objetivo desta dissertação é integrar a metodologia desenvolvida por

[1] "empresa" é um termo genérico podendo designar uma organização comercial, científica, técnica, etc.

[Per90] para a geração de dados convencionais e o modelo proposto por [Cif95] para geração de dados espaciais de aplicações geográficas, possibilitando a caracterização de banco de dados espaciais os mais próximos possíveis de aplicações reais. Esta caracterização possibilitará a análise de desempenho de sistemas de banco de dados espaciais. Ainda, um conjunto expressivo de transações comumente encontradas em aplicações SIGs são identificadas e poderão compor a carga de trabalho de um *benchmark* voltado a estas aplicações.

De modo geral, a análise de desempenho de sistemas de banco de dados procura responder questões tais como:

- **Avaliação da relação** *custo x benefício*. A partir desta relação, tem-se condições de se optar por um sistema computacional que se ajuste o mais próximo possível às necessidades de desempenho de uma aplicação e às restrições de gastos impostas por uma determinada empresa. Em geral, esta relação é utilizada em situações tais como: escolha das diversas alternativas de projeto, avaliação de um SGBD ou ainda, de um *hardware* específico.

- **Avaliação da capacidade**. A análise de desempenho pode ser necessária para avaliar se uma determinada alternativa possui um nível de desempenho aceitável para uma aplicação específica. Uma questão que sempre surge para um gerente de sistema é saber se a aplicação desejada pode ser executada convenientemente em um determinado sistema computacional.

- **Comparação entre diferentes tecnologias**. Neste caso, os resultados da análise de desempenho facilita a identificação de restrições de desempenho em determinados produtos. Estas restrições (tanto a nível de pesquisa como tecnologias empregadas) tendem a direcionar os trabalhos futuros para que estes produtos se tornem comercializáveis.

1.2 Organização da tese

Esta dissertação está dividida em 6 capítulos. Este capítulo inicial introduziu o trabalho mostrando o contexto, principais motivações e objetivos.

O capítulo 2 trata dos principais conceitos de banco de dados bem como novas necessidades da área para se trabalhar com Sistemas de Informações Geográficas - SIGs. Define o conceito de SIG a ser utilizado ao longo desta dissertação e apresenta algumas das áreas de aplicações de SIGs.

Já o capítulo 3 trabalha os principais conceitos envolvendo análise de desempenho, mais precisamente, voltada à área de banco de dados. Ainda, em relação à análise de

desempenho de banco de dados, enfoca-se aplicações da técnica de *benchmark* de banco de dados onde são discutidas características do banco de dados e da carga de trabalho. Também, são apresentadas algumas propostas que vêm se tornando verdadeiros padrões de avaliação de SGBDs, além dos trabalhos dos autores Rafles Pereira ([Per90]) e Ricardo Ciferri ([Cif95]), por irem de encontro ao trabalho proposto por esta dissertação.

O capítulo 4 propõe o esquema de caracterização de sistemas de banco de dados espaciais para análise de desempenho utilizando dados sintéticos que melhor representam aplicações reais, com base nos trabalhos dos autores Rafles Pereira e Ricardo Ciferri e nos conceitos discutidos nos capítulos anteriores. Esta caracterização enfoca tanto dados convencionais como dados não convencionais. Também, um conjunto representativo de transações encontradas em SIGs que poderão compor a carga de trabalho de um *benchmark* são apresentadas.

O capitulo 5 apresenta um estudo de caso junto a uma aplicação real - o projeto SAGRE da Telebrás - Telecomunicações Brasileiras S/A.

O capítulo 6 apresenta a conclusão do trabalho e extensões propostas.

Finalizando, tem-se a bibliografia utilizada.

Capítulo 2

Sistema de banco de dados

2.1 Introdução

As pesquisas na área de banco de dados iniciaram-se na década de 60 [Cox91], motivadas pela dificuldade de gerenciar grande volume de dados. Estas pesquisas foram, inicialmente, frutos dos centros industriais que buscavam atender o grande desenvolvimento das aplicações comerciais.

Como resultado dessas pesquisas, surgiram os Sistemas Gerenciadores de Banco de Dados - SGBDs, que tinham como objetivo tornar mais eficaz o gerenciamento de grandes quantidades de dados.

Atualmente, embora as pesquisas em banco de dados estejam bem evoluídas com a utilização dos SGBDs em quase todos os ambientes computacionais, a variedade de áreas de aplicação onde esta tecnologia tem sido requisitada vem dirigindo as pesquisas para prover o melhor suporte às aplicações.

Por exemplo, nas empresas que prestam serviço público de telecomunicações é necessário uma enorme quantidade de equipamentos (cabos, centrais telefônicas, armários, etc) distribuídos geograficamente em uma determinada área para suportar serviços de telecomunicações tais como telefonia, telex e comunicação de dados, entre outros. Além disso, os funcionários realizam inúmeras atividades que visam proporcionar serviços para os usuários bem como administrar as operações da companhia. Uma vez que o volume de informações gerado durante as atividades dessas empresas é considerável, sua administração deve dispor de Sistema de Informações (SI) adequados para manipular, armazenar e distribuir estas informações entre as diferentes áreas da empresa de modo a garantir confiabilidade, integridade e disponibilidade das informações, assim como a redução de custos de geração e manutenção.

Na parte central de um SI, geralmente, reside um sistema de bancos de dados. Este sistema pode ser visto como a combinação de:

- *Software* específico de gerenciamento de dados (SGBD);
- Um conjunto de programas de aplicação;
- Banco de dados e
- Sistema computacional - *hardware* e sistema operacional.

Um banco de dados é uma coleção de dados operacionais pertencentes a um sistema de informação e que podem ser utilizados por vários indivíduos dentro de uma ou mais empresas.

Um SGBD, por sua vez, é uma ferramenta de *software* genérica destinada a definição, manutenção, análise e manipulação de um banco de dados. O sistema deve proporcionar segurança das informações armazenadas no banco de dados contra eventuais quedas do sistema ou mesmo tentativas de utilização não apropriadas. Deve permitir o compartilhamento dos dados, evitando possíveis erros de integridade.

Em particular, tem-se procurado incorporar a tecnologia dos SGBDs a aplicações geográficas e cartográficas, aplicações de gerenciamento urbano (rede de telefonia, tráfego viário e de energia), aplicações CAD/CAM (projeto e fabricação auxiliado por computador) e aplicações com imagem (robótica e reconhecimento de padrões), dentre outras. Nestes casos, busca-se obter os mesmos benefícios encontrados na integração da referida tecnologia às aplicações comerciais tradicionais: eficiência no acesso e modificação dos dados, persistência, concorrência e segurança.

A incorporação da tecnologia dos SGBDs às aplicações citadas porém, não se resume à simples adequação de um SGBD existente a cada uma destas aplicações. As aplicações comerciais, as quais a maioria dos SGBDs visam atender, possuem características bem distintas dessas novas aplicações.

Vale ressaltar a distinção de aplicações convencionais e aplicações espaciais. Em seu trabalho, [Cox91] diferencia aplicações comerciais (convencionais) e aplicações espaciais (não convencionais) através dos seus tipos de dados básicos. Enquanto as aplicações comerciais, em geral, possuem como tipos de dados inteiros e cadeias de caracteres (*strings*), as aplicações espaciais, embora com características distintas, possuem em comum tipos de dados em n-dimensões e algumas vezes imagens.

Outro ponto importante refere-se à linguagem de consulta a ser utilizada. Uma consulta envolvendo predicados espaciais é apresentado abaixo, utilizando a linguagem de consulta GeoQL [Ooi90]. Esta consulta pretende determinar todas as cidades que são cortadas pelo rio Tietê e que possuem população superior a 500.000 habitantes.

```
SELECT      CIDADE.nome
FROM        CIDADE, RIO
WHERE       RIO.nome = ''TIETE'' and
            CIDADE.populacao > 500000 and
            RIO intersects CIDADE ;
```

Nesta consulta percebe-se que as aplicações espaciais também possuem atributos não espaciais (RIO.nome, CIDADE.população) e que, portanto, também devem ser suportados.

Em decorrência dos requisitos de dados das aplicações espaciais serem bem distintos em relação às aplicações convencionais, os SGBDs destinados àquelas aplicações devem possuir algumas características próprias para essa classe de aplicações.

2.2 Sistemas de banco de dados espaciais

Quando os primeiros SIGs surgiram, pouco êxito foi obtido devido às grandes dificuldades inerentes aos recursos existentes na época. No início dos anos 80, com os avanços significativos na área, o dilema dos construtores de SIG recaia no fato de que os SGDBs comerciais existentes lidavam tipicamente com dados simples - registros alfanuméricos unidimensionais. Esses SGBDs não se adequavam às novas aplicações espaciais emergentes. Dentre as principais razões, pode-se destacar que os SGBDs:

- Não foram projetados para lidar com *arrays* de tamanho variável. A localização de um objeto em SIG é descrita através de coordenadas geográficas. Estas coordenadas podem ser representadas por um *array* de tamanho *um* (representando um único ponto) ou assumir grandes dimensões (quando representando um segmento de linha ou polígono).

- Ofereciam apenas índices unidimensionais. Aplicações SIGs requerem índices bidimensionais (às vezes tridimensionais) sobre milhões de registros.

- Permitiam modelar relacionamentos 1:N (um para muitos) e N:1 (muitos para um) porém com baixo desempenho no caso de junção de relacionamentos (estabelecidos em tempo de execução). Aplicações SIGs requerem a composição (ou combinação) de objetos primitivos permitindo a construção de objetos complexos. Um cabo telefônico, por exemplo, pode conter centenas de condutores.

- Não ofereciam análise de proximidade. Um dos requisitos de aplicações SIGs é localizar e associar objetos regionais a algum interesse específico: as propriedades vizinhas numa aplicação de rezoneamento, por exemplo.

- Bibliotecas com rotinas de apresentação (que pode constituir um banco de dados por si mesmo), não eram devidamente suportadas pelos SGBDs no passado. As regras para apresentação visual de um banco de dados espacial devem ser simples, levando em consideração o usuário, a escala e a aplicação sendo modelada.

- Não eram adequados ao armazenamento e indexação de objetos binários grandes tais como fotos, arquivos gráficos, etc.

Atualmente, graças ao crescente desenvolvimento tecnológico e o surgimento de novas necessidades por parte das aplicações dos usuários, os SGBDs necessitam manipular adequadamente dados convencionais e não convencionais para atender os novos requisitos de projetos. [Sea95] destaca as seguintes características a serem suportadas para o sucesso comercial desses SGBDs:

- Grande volume de dados. Bancos de dados geográficos contêm milhões de entidades geográficas, cada qual associada a uma ou mais localização dentro de um território.

- Os dados geográficos devem estar disponíveis a várias aplicações e não serem mantidos isoladamente.

- Deve-se satisfazer diferentes grupos de usuários. Cada grupo tem diferentes necessidades em termos de método de acesso a dados geográficos, visualização, alteração e visões dos mesmos.

- Os dados não convencionais devem existir em harmonia com dados convencionais em ambiente de banco de dados.

- Adequados requisitos de segurança. O acesso aos dados e arquivos deve ser controlado para que apenas usuários autorizados possam acessá-los e/ou modificá-los.

- A construção do banco de dados geográfico é dispendiosa, tornando-se um ponto estratégico dentro da empresa ou organização. Integridade do banco de dados e considerações de segurança tornam-se significativas.

- Dados geográficos devem ser explorados por visões e , ocasionalmente, alterados por aplicações comerciais que utilizem *interface* de programas de aplicação adequado.

- Alterações do banco de dados podem ser realizadas por vários usuários simultaneamente, devendo ser adequadamente gerenciadas e controladas. Em algum momento, as alterações dos dados não convencionais podem ser on-line (em um ambiente de processamento de transações), com vários usuários trabalhando sobre o mesmo conjunto de objetos do banco de dados. No caso de transações de alterações longas, o novo estado do banco de dados deve ser refletido somente se a transação for executada corretamente.

- O banco de dados deve existir em várias versões concorrentemente, cada qual representando um cenário operacional diferente ou possibilidades comerciais.

Já [Cam95] aborda os requisitos abaixo como sendo necessários ao sucesso dos SGBD não convencionais:

- Avanços na modelagem conceitual em geoprocessamento para quebrar a dicotomia matricial-vetorial e para gerar interfaces com maior conteúdo semântico.

- Integração Sensoriamento Remoto-Geoprocessamento. Um dos requisitos mais importantes para análise espacial é a integração entre mapas temáticos, modelos de terreno e imagens de satélites necessário a aplicações ambientais e processamento de imagens de satélites.

- Representações topológicas em múltiplas escalas e projetos. O gerenciamento de um banco de dados geográfico de grandes dimensões requer que se mantenham múltiplas representações geométricas associadas ao mesmo dado geográfico.

- Linguagem de consulta, manipulação e apresentação de objetos não convencionais de grande poder expressivo.

- Novas técnicas de análise geográfica para satisfazer de forma plena os requisitos de análise e modelagem de grandes bases de dados espaciais.

- Arquitetura de banco de dados de grande porte. Mudanças nos esquemas tradicionais de arquitetura de sistemas de gerência de banco de dados serão requeridos.

A linguagem de consulta é parte integrante de um SGBD. Ela deve ser simples de apreender e usar, possibilitando um poder de expressão onde os usuários possam interagir com o SGBD de maneira amigável e consistente.

Em SIG utiliza-se, geralmente, a linguagem de consulta do SGBD sobre o qual foi implementado, com algumas extensões. Salienta-se que esta linguagem deve fornecer operadores não convencionais, geométricos e topológicos. Os operadores geométricos retornam um valor escalar (operadores de distância entre dois objetos e área de um objeto

geográfico, por exemplo). Operadores topológicos retornam um valor *booleano*, explorando conceitos de disjunção, intersecção, inclusão, etc entre objetos geográficos. Operadores espaciais retornam um objeto geográfico a partir de dois ou mais objetos geográficos.

Conceitualmente, todo SGBD possui um modelo de dados que direciona toda a sua filosofia de utilização. Entre os modelos de dados mais conhecidos destacam-se: modelo em rede, hierárquico, relacional e o modelo orientado a objetos ([Dat86], [KS89] e [EN94]).

Apesar de serem utilizados de forma eficiente em muitas aplicações, estes modelos apresentam limitações do ponto de vista semântico. Para tentar superar estas limitações foram propostos modelos que possuem construtores para modelar abstrações mais poderosas dos relacionamentos entre entidades (objetos), permitindo uma visão mais natural e realista. Entre os modelos semânticos mais conhecidos estão os modelos entidade-relacionamento - MER, o modelo funcional e o modelo de objetos ([Dat86], [KS89] e [EN94]). Normalmente, cada modelo possui uma nomenclatura própria para descrever os vários objetos que compõem um banco de dados.

Para facilitar o entendimento ao longo desta dissertação, adota-se um modelo de representação dos dados conforme [Per90]. Este modelo é estruturado em dois níveis: nível lógico e nível físico.

Em nível lógico, a preocupação consiste em obter apenas a parte de interesse da aplicação a ser analisada. Para auxiliar na obtenção dessas características, adotou-se o modelo de entidade-relacionamento.

Em relação ao nível físico, as informações obtidas no nível lógico são utilizadas para caracterizar o banco de dados. Isso pode ser realizado obtendo-se valores para as relações (seus atributos), contidas no esquema obtido anteriormente, através da caracterização mais detalhada dos dados, em termos dos domínios relacionados e de descrições suplementares dos atributos. Além das informações sobre os dados, as estruturas de índices e dos agrupamentos devem também ser informadas .

Maiores detalhes sobre o modelo de dados adotado são apresentados no capítulo 4.

Percebe-se, como mencionado anteriormente, que SGBDs voltados a SIGs necessitam oferecer conjuntamente mecanismos e técnicas adequadas para tratamento de dados convencionais e não convencionais (espaciais e gráficos),bem como fornecer a integração desses dados que podem estar distribuídos por diversas aplicações dentro de uma empresa ou organização. O sucesso desses SGBDs colaborará no desenvolvimento de aplicações geográficas cada vez mais adaptáveis à realidade dos usuários e, principalmente, das empresas e organizações.

2.3 Caracterização de SIG

A caracterização de sistemas de informações geográficas torna-se uma tarefa difícil devido à rápida e recente evolução desses sistemas e também pelo fato de existirem várias classificações desses, cada qual voltada basicamente à aplicação a que se destinam.

Classificações baseadas em custo, plataforma, funcionalidade, área de aplicação e modelo de dados são as mais discutidas. Em [Cif95],[Agu95] e [Cam95] várias abordagens de classificação são apresentadas sendo que, nesta dissertação, adota-se a caracterização final de [Cif95] uma vez que visa-se implementar muitos dos conceitos e contribuições que foram apresentados por este autor.

Assim, segundo [Cif95] um SIG é caracterizado por três componentes básicos. O primeiro deles enfoca a produção de mapas, ou seja, é baseado em aspectos cartográficos [Tom91]. O segundo componente enfatiza o uso de um SGBD, constituindo uma visão voltada para sistemas de banco de dados. O último componente, enfatiza a importância da presença de análise espacial, ou seja, de um conjunto de funções analíticas que manipula objetos espaciais (geográficos).

Consolidando estes três componentes com o conceito de sistema de informação, um SIG pode ser definido como um sistema com:

- **Capacidade cartográfica.** Refere-se à captura (e entrada) de dados e a geração de informações. A captura dos dados envolve técnicas tais como fotografia, digitalização de mapas existentes, documentos arquivos, etc. A saída das informações se refere ao processo de vizualização das operações realizadas por um SIG (impressão de mapas e *zoom*, por exemplo).

- **Capacidade de gerenciamento de dados.** Um SIG é formado por um SGBD não convencional, o qual é usado para armazenar dados relativos a objetos geográficos [1], e por um conjunto de processos especializados no tratamento de dados espaciais.

  Destaca-se que SGBDs não convencionais devem suportar de forma transparente problemas relacionados a dados espaciais, geométricos e topológicos. Entre estes problemas temos: linguagem de consulta, métodos de acesso, otimizador de consultas, representação física dos dados, modelo de dados que represente tanto os dados espaciais quanto os dados convencionais, além de relacionamentos ([Agu95], [Cif95]).

- **Capacidade analítica.** O sistema deve ter a habilidade de interpretar os dados espaciais armazenados no SGBD. Algumas funções analíticas são:

  – Computação de operações escalares: por exemplo, distância entre dois pontos.

[1] Um objeto geográfico representa uma entidade estática do mundo real, possuindo uma localização fixa em relação à superfície terrestre, descrita por uma geometria.

- Superposição de polígonos: polígonos representando um tema (por exemplo: tipo de solo) são superpostos por polígonos representando um outro tema (por exemplo: limites geográficos regionais) para gerar novos polígonos.
- Análise de proximidade: conhecida também como zona de *buffer*, esta função consiste em gerar uma área (objeto geográfico bidimensional), na forma de um "corredor", ao redor do objeto geográfico fonte.
- Busca espacial: retorna um conjunto de objetos geográficos que satisfaz um certo relacionamento topológico em relação a um objeto geográfico fonte.

Desta forma, um SIG é caracterizado como um *software* composto por vários subsistemas integrados, os quais são voltados para a geração de mapas e para a extração de informações sobre os objetos geográficos representados nestes mapas, com o auxílio de um SGBD não convencional e de um conjunto de funções analíticas (figura 2.1 adaptada de [Cif95]).

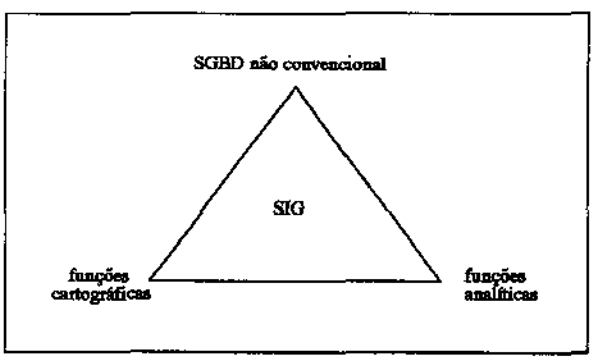

Figura 2.1: Componentes de um SIG.

A figura 2.2 indica os relacionamentos dos principais componentes de um SIG segundo [Cam95] e [CCH+96]. Percebe-se que, embora seja apresentado de maneira mais detalhado, os componentes básicos de um SIG são geralmente os mesmos: a entrada e integração dos dados e o processo de visualização e plotagem correspondem a funções cartográficas apresentados por [Cif95]; a consulta e análise espacial correspondem a funções analíticas, e assim por diante. Salienta-se ainda que cada sistema, em função de seus objetivos e necessidades, implementa estes componentes com algumas particularidades, mas todos os subsistemas citados estão presentes num SIG.

2.4 Tipos de dados em SIG

Dados geo-referenciados descrevem fatos, objetos e fenômenos do globo terrestre associados à sua localização sobre a superfície terrestre, num certo instante ou período de

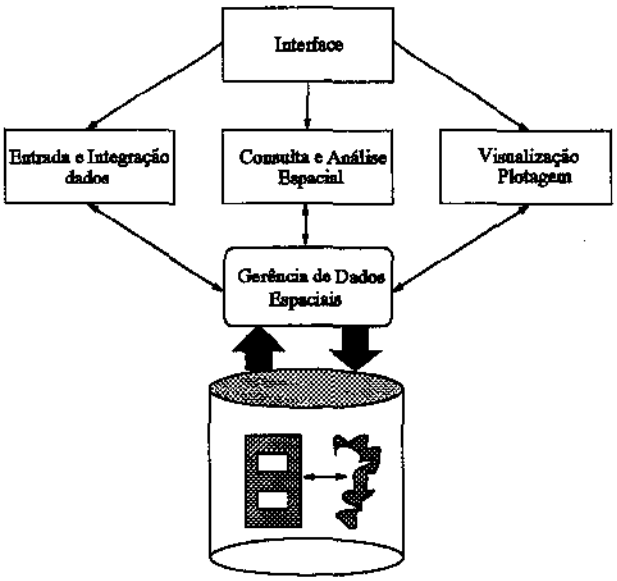

Figura 2.2: Arquitetura de Sistemas de Informação Geográfica.

tempo. Estes dados, tratados em SIG, têm como principal característica a diversidade de fontes geradoras e de formatos apresentados. Para cada objeto geográfico, o SIG necessita armazenar seus atributos e as várias representações gráficas associadas.

De acordo com [MM93] e [Tim94], os dados geo-referenciados associados a objetos geográficos são classificados em: espaciais, convencionais ou gráficos.

Dados espaciais referem-se a um conjunto de coordenadas que descrevem a geometria de um objeto geográfico. Para representar esta geometria, são usados basicamente os conceitos de ponto, linha e polígono. Um ponto é a menor unidade possível para representar um objeto geográfico, como por exemplo, uma cidade em um mapa. Uma linha é uma seqüência de pontos conectados, onde cada par de pontos conectados corresponde a um segmento de linha, por exemplo, rios e estradas. Um polígono é formado por uma linha fechada, por exemplo, área ocupada por um bairro ou cidade. Pontos, linhas e polígonos são estruturas 0-dimensionais, 1-dimensionais e 2-dimensionais, respectivamente.

Os dados convencionais [2] representam os atributos de um objeto geográfico - nome de uma cidade, por exemplo.

Dados gráficos são formados por *pixels*. Um *pixel* corresponde à menor unidade de exibição de uma imagem. Estes dados são utilizados para representar dados espaciais e convencionais na tela do computador, de acordo com estilo, cor, forma e tamanho; e são manuseados por funções de processamento de imagens.

[Bou88], por sua vez, classifica os dados encontrados em SIG em dados cadastrais, dados topológicos, redes e dados temáticos.

[2] descritivos

Dados cadastrais preocupam-se com a localização geográfica, identificação e caracterização do objeto geográfico. Cada objeto geográfico possui atributos e pode estar associados a várias representações gráficas. Por exemplo, os lotes de uma cidade possuem atributos (proprietário, valor venal, etc) e podem ter representações gráficas diferentes em escalas distintas.

Dados topológicos refere-se à descrição de rodovias, ruas e avenidas, entre outras.

O termo redes denota basicamente as informações associadas a redes de utilidade pública (água, luz, esgoto e telefone), redes de drenagem e rodovias. Cada objeto geográfico (cabo telefônico, caixa subterrânea, cano de água) possui uma localização exata e está sempre associado a atributos descritivos, presentes no banco de dados.

As redes formam um capítulo à parte na tipologia de SIGs; são o resultado direto da intervenção humana sobre o meio ambiente. Cada aplicação de rede tem características próprias e com alta dependência cultural (por exemplo, a largura das rodovias nos EUA é distinta das usadas em São Paulo) [Cam95].

Dados temáticos são dados relativos a um assunto ou tema específico, representados por polígonos. Alguns exemplos são o uso do solo, área de poluição, divisão política e aptidão agrícola de uma região.

Já [Cam95] classifica os dados de um sistema de geoprocessamento de acordo com uma fonte e aplicação. São eles: mapas temáticos, mapas cadastrais, redes, imagens e modelos numéricos de terreno (MNT).

Mapas temáticos, mapas cadastrais e redes manipulam os mesmos tipos de dados comentados anteriormente, ou seja, dados temáticos, cadastrais e redes, respectivamente. Já imagens referem-se a dados obtidos por sensoriamento remoto. O termo modelo numérico de terreno é utilizado para denotar a representação de uma grandeza que varia continuamente no espaço, sendo comumente associados a altimetria; podem também ser utilizados para modelar unidades geológicas (como teor de minerais ou propriedades do solo ou subsolo).Os modelos numéricos de terreno podem ser convertidos para mapas temáticos e para imagens.

Nesta dissertação adota-se a classificação dos dados geográficos em convencional, não convencional (espacial) e gráficos.

2.5 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. Por outro lado, os usuários SIG também têm uma grande variedade de perfis, como cientistas

especialistas em determinado domínio do conhecimento [3], técnicos [4] ou especialistas em administração e planejamento urbano. Ainda, cada aplicação requer a manipulação de fenômenos geográficos distintos, associados a diferentes características e propriedades que variam no espaço e no tempo, promovendo assim a necessidade de um conjunto adequado de funções de análise e manipulação de dados geográficos. A partir de tais fatos, surgem diferentes especializações do termo SIG.

[MGR93] classifica as aplicações em:

- sócio-econômicas: envolve o uso de terra, seres humanos e infraestrutura existente. Exemplos típicos são o acompanhamento e inventário de cadastros imobiliários rurais e urbanos; definição de uma política para uso do solo; aplicações envolvendo serviços de utilidade pública (redes de telefonia, eletricidade, esgotos, transporte); sistema de auxílio à navegação; estudos de *marketing*; e alocação de recursos em geral para manutenção ou expansão de infraestrutura de uma região.

- ambientais: enfoca o meio ambiente e o uso de recursos naturais. Como exemplos têm-se aplicações ligadas à modelagem climática e ambiental, uso do uso, previsão numérica do tempo, monitoração do desflorestamento, monitoração da emissão e ação de poluentes, identificação e mapeamento mineral e petrolífero, planejamento e supervisão de redes hidroelétricas, e gerenciamento costeiro e marítimo.

- gerenciamento: envolve a realização de estudos e projeções que determinam onde e como alocar recursos para remediar problemas ou garantir a preservação de determinadas características. Como exemplos desta classe de aplicações há planejamento de tráfego urbano, planejamento e controle de obras públicas, planejamento da defesa civil.

Maiores detalhes sobre aplicações de SIG podem ser encontrados em [CCH+96]. Neste trabalho, pode-se encontrar descrições de várias aplicações bem como bibliografia relacionadas. Também em [Cif95] pode-se obter informações sobre aplicações porém, este autor classifica as aplicações em urbanas e ambientais.

2.6 Conclusões

Neste capítulo apresentou-se um breve histórico dos avanços e necessidades dos SGBDs, mais especificamente aqueles voltados a aplicações geográficas. Também, caracterizou-se o conceito de SIG e os demais conceitos envolvidos em geoprocessamento relevantes ao

[3] biólogos, geólogos, sociólogos etc.

[4] engenheiros, arquitetos etc.

entendimento do trabalho. Ainda, algumas áreas onde estas aplicações são utilizadas foram citadas.

Sendo assim, vale ressaltar que um SIG é caracterizado por três componentes básicos:

- capacidade cartográfica;
- capacidade de gerenciamento de dados (SGBD não convencionais); e
- capacidade analítica.

Quanto aos dados geográficos adota-se a classificação em: não convencional (espacial), convencional e gráfico.

Para facilitar o entendimento ao longo deste trabalho e ajudar na caracterização do banco de dados das aplicações, adotamos um modelo estruturado em dois níveis: nível lógico e nível físico.

No nível lógico, preocupamos em obter, a partir do mundo real, apenas a parte de interesse da aplicação a ser analisada. Para auxiliar na obtenção dessas características, adotou-se o modelo de entidade-relacionamento.

Em relação ao nível físico, as informações obtidas no nível lógico são utilizadas para caracterizar o banco de dados. Isso pode ser realizado obtendo-se valores para as relações (seus atributos), contidas no esquema obtido anteriormente, através da caracterização mais detalhada dos dados, em termos dos domínios relacionados e de descrições suplementares dos atributos. Além das informações sobre os dados, as estruturas de índices e dos agrupamentos devem também ser informadas.

Maiores detalhes sobre o modelo de dados adotado são apresentados no capítulo 4.

Capítulo 3

Análise de desempenho de SGBD espacial

3.1 Introdução

Com o advento da indústria de informática, as empresas passaram a oferecer produtos de *hardware* e *software* com as mesmas ou semelhantes funcionalidades. Diante desse contexto, mecanismos que pudessem medir e comparar estes produtos tornaram-se necessários. A análise de desempenho é um dos mecanismos mais utilizado para tal propósito.

A análise de desempenho consiste em medir a eficiência com a qual produtos efetuam uma determinada funcionalidade a partir de alguma medida escalar (tempo, por exempo). A comparação entre vários produtos similares pode ser obtida através da ordenação dos resultados de desempenho produzidos.

Observa-se que a análise de desempenho é realizada em várias áreas da ciência da computação, dentre elas: sistemas operacionais, arquitetura de computadores, teoria da computação e banco de dados. Para esta dissertação, a área de banco de dados será abordada pois um dos componentes principais de SIGs é um SGBD espacial (não convencional).

De modo geral, a importância da análise de desempenho de sistemas de banco de dados se resume em resolver questões como:

- **Avaliação da relação** *custo x benefício.* A partir desta relação, tem-se condições de se optar por um sistema computacional que se ajuste o mais próximo possível às necessidades de desempenho de uma aplicação e às restrições de gastos impostas por uma determinada empresa. Em geral, esta relação é utilizada em situações tais como: escolha das diversas alternativas de projeto, avaliação de um SGBD ou ainda, de um *hardware* específico.

- **Avaliação da capacidade.** A análise de desempenho pode ser necessária para avaliar se uma determinada alternativa possui um nível de desempenho aceitável para uma aplicação específica. Uma questão que sempre surge para um gerente de sistema é saber se a aplicação desejada pode ser executada convenientemente em um determinado sistema computacional.

- **Comparação entre diferentes tecnologias.** Neste caso, os resultados da análise de desempenho facilitam a identificação de restrições de desempenho em determinados produtos. Estas restrições (tanto a nível de pesquisa como tecnologias empregadas) tendem a direcionar os trabalhos futuros para que estes produtos se tornem comercializáveis.

3.2 Análise de desempenho de SGBD

Os SGBDs possuem inúmeras facilidades para o desenvolvimento de sistemas. Conceitualmente, um SGBD é um *software* genérico concebido para gerenciar grandes quantidades de dados. Este gerenciamento envolve tanto a definição das estruturas para o armazenamento das informações assim como a provisão de mecanismos para manipulá-las.

O gerenciamento e manutenção desses SGBD ainda é considerada uma tarefa bastante complexa. Entre as primeiras decisões a serem tomadas com relação à utilização de um SGBD referem-se a escolha de quais aspectos das atividades da empresa devem ser representados no banco de dados, escolha do modelo de dados para representar as informações, a linguagem de desenvolvimento e, até mesmo, a escolha das estruturas físicas dos arquivos. A um nível mais baixo, muitas outras decisões ainda são necessárias, como a escolha de índices para otimizar o custo de processamento das transações, a determinação dos tamanhos dos blocos etc. Todas essas decisões guardam uma forte interação entre si e, se por um lado algumas delas podem parecer bastante intuitivas, em geral é muito difícil determinar qual a influência de tantas variáveis sobre o desempenho do SGBD [Per90].

Ainda, em relação ao desempenho de um SGBD, dois aspectos devem ser considerados: o desempenho do usuário e o desempenho do próprio sistema. O desempenho do usuário se refere a aspectos relativos à facilidade de utilização, nível da interface do usuário, aprendizagem da linguagem de programação etc. O desempenho do sistema se preocupa basicamente com a avaliação de fatores quantitativos que procuram expressar a eficiência do sistema no atendimento de requisições dos usuários, tais como: o tempo de resposta, o "*throughput* [1], número de referências lógicas e físicas à memória secundária.

[1] Número de transações executadas por unidade de tempo (tps).

Nesta dissertação, o interesse está no desempenho do sistema. Os aspectos referentes ao desempenho do usuário, apesar de sua relevância, não serão tratados aqui.

O desempenho do sistema é resultado de uma complexa interação entre fatores internos e externos ao SGBD. Em seu trabalho, [Per90] destaca os seguintes fatores internos:

- O grau de concorrência e compartilhamento do sistema.
- A carga de trabalho (conjunto de transações e perfil de freqüência de utilização).
- A disponibilidade de caminhos de acesso.
- A configuração adequada de parâmetros do sistema (tamanho de blocos, de *buffers* etc).
- O conteúdo, distribuição e disposição dos valores no banco de dados.
- A política de gerenciamento de memória.
- O *software* que implementa o próprio SGBD, com seus módulos que são responsáveis pelas atividades de organização e manutenção do banco de dados, execução de transações etc.

Já os fatores externos causam uma carga de trabalho adicional ao ambiente computacional no qual o SGBD está instalado. Esta carga de trabalho é composta de programas que são executados concorrentemente ao SGBD e que disputam os mesmos recursos computacionais (memória, processadores etc).

O esquema de caracterização proposto nesta dissertação destaca os fatores conteúdo, distribuição, disposições dos valores no banco de dados e a carga de trabalho da aplicação. Os resultados obtidos visam a melhor representação de um sistemas de banco de dados sintético, e que mais se aproxima de aplicações reais, ou conjunto típico de aplicações.

Entre os fatores que influenciam o desempenho, existem aqueles em que os usuários não possuem nenhum controle (o código que implementa o próprio SGBD, por exemplo). Porém, algumas modificações como a criação de índices e agrupamentos, a definição de parâmetros físicos (tamanho de bloco, número de *buffers* etc) do SGBD devem ser realizadas pelo próprio usuário ou pelo gerente do sistema e podem otimizar a utilização dos recursos disponíveis no sistema.

Geralmente, na análise de desempenho se utilizam os seguintes modelos:

- **Modelo analítico**: baseia-se na obtenção de um conjunto de equações matemáticas juntamente com os algoritmos para resolvê-las, que relacionam medidas de desempenho a parâmetros do sistema. Usualmente, empregam-se várias hipóteses teóricas

a respeito do conteúdo do banco de dados, colocação dos registros nos arquivos e comportamento da carga de trabalho de modo a simplificar o modelo obtido. O custo econômico de sua aplicação é relativamente baixo, dado a facilidade de ser utilizado e a rapidez com que são obtidos os resultados.

- **Modelo de simulação**: procura reproduzir as atividades do sistema de banco de dados de acordo com um conjunto de hipóteses e condições, eliminando a necessidade de experimentação no próprio sistema. Modelos de simulação consideram a entrada (*input*) e saída (*output*) de dados como parte do mundo real e portanto, devem ser modelados. Para reproduzir resultados precisos o modelo de simulação deve crescer em complexidade e isso requer um conhecimento maior e mais completo dos valores dos parâmetros que descrevem o sistema. Em alguns casos, no entanto, não é possível ou então é muito difícil obter esses valores; por exemplo, durante os estágios primários de desenvolvimento de um sistema, em que os dados não estão disponíveis, ou mesmo durante sua operação, em que muitas vezes são difíceis de serem obtidos sem afetar o sistema em produção.

- **Modelo experimental**: também chamado de modelo empírico, procura utilizar o próprio sistema de banco de dados para se obter os resultados. Neste modelo, duas técnicas são popularmente conhecidas: *benchmark* e monitoração. Na análise de desempenho de um sistema computacional, a técnica de *benchmark* consiste na execução de um conjunto fixo de programas sobre um determinado sistema (pode se utilizar cargas de trabalho sintéticas sobre um banco de dados também sintético). Já a técnica de monitoração consiste em utilizar ferramentas próprias de avaliação estatística presentes no sistema sendo analisado. Devido à ausência de padronização das ferramentas utilizadas na técnica de monitoração, seu uso torna-se específico e portanto limitado. Por outro lado, a técnica de *benchmark* pode ser aplicada na comparação de diferentes sistemas pois é padronizada, sendo os testes invariáveis e bem definidos. Ao contrário dos modelos analítico e simulação, o modelo experimental depende da maturidade da tecnologia envolvida, ou seja, todos os componentes do sistema já devem ter sido implementados.

Em alguns casos é possível utilizar modelos híbridos, envolvendo mais de um modelo. Estes modelos procuram associar as vantagens de cada modelo distinto de forma a obter uma representação do sistema mais realista [Cif95, Per90]. Nesta dissertação enfoca-se o modelo experimental, utilizando a técnica de *benchmark* utilizando dados sintéticos. A técnica de *benchmark* será apresentada na próxima seção.

Observa-se que a proliferação de SGBDs no mercado, bem como ofertas de versões mais atualizadas dos sistemas já existentes, têm fomentado cada vez mais a preocupação com o desempenho desses sistemas. Também, se por um lado esta grande disponibilidade

de SGBDs favorece o usuário no processo de aquisição/compra tendo em vista os muitos fornecedores, por outro lado, aumenta a expectativa em relação a qual melhor se aplicaria às suas reais necessidades.

3.3 A técnica de *benchmark*

O objetivo principal de um *benchmark* é a avaliação de desempenho de sistemas computacionais. Em sua grande maioria, quer-se medir quão rápido um dado sistema computacional efetua um determinado conjunto de tarefas.

Um *benchmark* deve possuir as seguintes características [Cif95, Gra91]:

- ser relevante (representativo) para a aplicação em questão;
- ser portável entre diferentes arquiteturas e configurações de sistemas computacionais;
- ser escalável, podendo ser executado tanto por pequenos computadores como *mainframes*; e
- ser simples de entender.

Estas características devem ser alcançadas conjuntamente sempre que possível, tomando o devido cuidado em relação à simplicidade. [Kim95] destaca ainda a importância de se projetar um *benchmark* que reflita um ambiente multi-usuário e, também, o fato de se ter sistemas ricos ou pobres em características. Por exemplo, um sistema de banco de dados relacional que suporta *joins* é relativamente rico se comparado a sistemas que não o fazem. Nestes casos, quando o sistema sendo avaliado não suportar uma particular característica, o sistema deve reportar "característica não suportada" ou "infinito" para o tempo de execução de uma determinada operação, melhor do que forçar uma simulação e levar a resultados menos confiáveis.

Através da técnica de *benchmark*, implementa-se e executa-se um conjunto de programas (sobrecarga de execução) em um dado sistema computacional onde se pretende avaliar o desempenho. Esta sobrecarga de execução específica é caracterizada como carga de trabalho do *benchmark*. Sabe-se que cargas de trabalho distintas influenciam no desempenho do sistema sendo analisado. A indefinição e composições distintas dos dados que comporão o *benchmark* podem degradar o desempenho de maneira diferenciada também. Deste modo, para possibilitar a comparação de resultados de desempenho, deve-se inserir uma constante de degradação única no desempenho de todos os sistemas a serem analisados e aspectos relacionados aos dados devem ser fixados para que os processos sempre atuem sobre dados com as mesmas características.

A técnica de *benchmark*, por se tratar de uma técnica experimental, requer que o sistema a ser medido esteja disponível para uso. O fato desta técnica ser aplicada sobre o próprio sistema sendo analisado torna os resultados gerados altamente confiáveis.

3.4 *Benchmark* de banco de dados

A técnica de *benchmark*, conforme descrito anteriormente, consiste em um modelo de análise experimental onde é executado um conjunto fixo de testes sobre um sistema para avaliar seu desempenho. Para sistemas que possuam como um de seus principais componentes um SGBD, utiliza-se a técnica de *benchmark* de banco de dados, uma especialização genérica de *benchmark*, onde transações são definidas e posteriormente são executadas sobre um banco de dados conhecido, com o objetivo de medir o desempenho do sistema.

Vale ressaltar que esta dissertação tratará o conceito de processo e transação como equivalentes. Uma transação logicamente consiste em um conjunto de operações de escrita e leitura em um banco de dados, além de operações de processamento. A implementação de transações é efetuada através de linguagem de alto nível, geralmente com o uso de uma linguagem de quarta geração, fornecida pelo SGBD, em adição a uma linguagem convencional, tal como a linguagem C. Como transações são implementadas através de programas e estes ao executarem tornam-se processos, pode-se dizer que uma transação é um tipo especial de processo, o qual tem por objetivo a leitura e escrita de dados em vários arquivos pertencentes a um determinado banco de dados.

A carga de trabalho de um *benchmark* de banco de dados é composta por um conjunto de transações e de suas respectivas freqüências de execução. Geralmente, denomina-se de carga de trabalho do banco de dados (*database workload*) aquela proveniente apenas das transações que são submetidas pelo usuários, e carga de trabalho externa (*NonDBMS workload*) aquela provocada pelos processos que são executados concorrentemente ao SGBD e que disputam os mesmos recursos computacionais: processadores, memória e unidades de disco.

Os dados, em *benchmarks*, podem ser caracterizados em dados sintéticos ou reais. Dados sintéticos consistem em dados gerados artificialmente. Dados reais consistem em um conjunto de dados existentes na própria aplicação em estudo. Esta dissertação busca obter junto a uma aplicação real características e mecanismos que possam gerar dados sintéticos tão próximos quanto possíveis destas aplicações para a análise de desempenho de sistemas de banco de dados espaciais.

Após definidas as transações e os dados que comporão o *benchmark*, deve-se definir a abrangência de cada transação em relação aos dados. Por exemplo: uma dada transação pode acessar exclusivamente um arquivo X, selecionando dados de acordo com o valor do

campo C_1 de cada registro deste arquivo; outra transação pode acessar exclusivamente o mesmo arquivo X, mas selecionando dados de acordo com o valor do campo C_2. Desta maneira, pode-se determinar a seletividade de cada transação (em relação aos dados).

Outro aspecto muito importante relaciona-se ao sistema de análise. O sistema de análise pode englobar o sistema computacional inteiro (ambiente de aplicação), ou apenas parte deste, tais como: um nível do sistema computacional, alguns processos pertencentes a um nível específico ou alguns processos pertencentes a vários níveis, entre outros.

A identificação do sistema de análise é essencial pois cada nível de *software* de um sistema computacional exerce específica influência no desempenho, através de sobrecargas de execução distintas [Cif95]. Por exemplo, suponha que um determinado *benchmark* considere o sistema de análise como sendo o sistema computacional completo, ou seja, quer-se medir o desempenho como um todo, não importando quais partes deste geram mais ou menos sobrecarga de execução. Estes *benchmarks* modelam o ambiente de aplicação onde procuram reproduzir as atividades tipicamente encontradas em um determinado tipo de aplicação. A figura 3.1 mostra a caracterização destes *benchmarks* que têm como principal objetivo medir o desempenho de uma determinada aplicação, rodando em uma determinada plataforma de *hardware* com um determinado sistema operacional, sendo que esta aplicação utiliza um *software* específico [2] para auxiliar a execução de suas atividades. Observa-se que neste caso não há carga de trabalho externa ao *benchmark*.

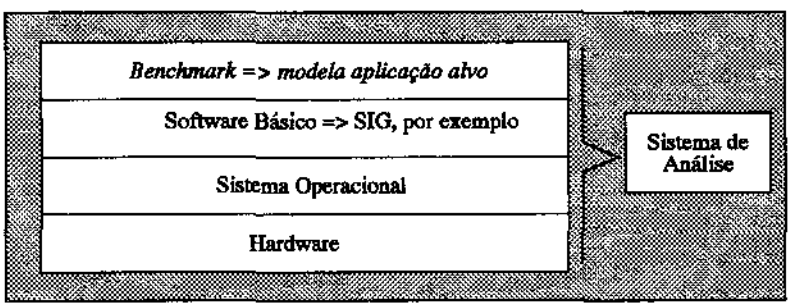

Figura 3.1: Sistema de análise - total sem carga de trabalho externa.

Um *benchmark* que leva em consideração um sistema de análise completo com carga de trabalho externa ao *benchmark* é apresentado na figura 3.2. Estes *benchmarks* têm por objetivo medir a degradação de desempenho provocada pela carga de trabalho externa ao *benchmark*, a qual influencia diretamente o ambiente de aplicação que o *benchmark* representa.

Algumas vezes, porém, o sistema de análise é parte do sistema computacional total. Nestes casos, deseja-se medir a influência proporcionada por partes específicas no desempenho do sistema computacional total e, em geral, utiliza-se a abordagem de análise

[2] *Software também conhecido como software básico.*

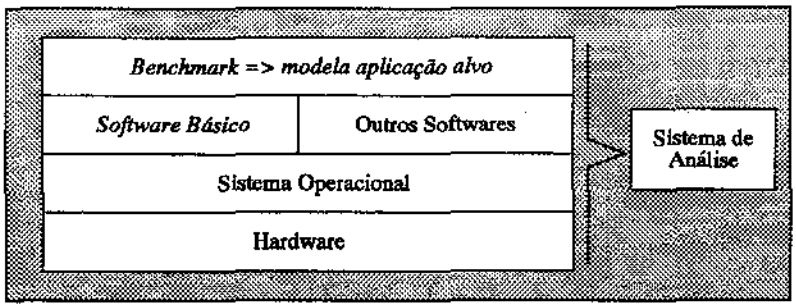

Figura 3.2: Sistema de análise - total com carga de trabalho externa.

comparativa entre diferentes configurações de sistemas computacionais.

3.5 Tipos de medidas de desempenho

Benchmarks são criados e usados freqüentemente para responder questões tais como: "Qual sistema computacional eu devo comprar?" ou "Qual SIG eu devo escolher para minha empresa?". A resposta mais adequada deveria ser "O sistema que faz o trabalho que você quer da melhor maneira possível e ao menor custo". As medidas de desempenho existentes, não respondem completamente a estas questões, conseguem apenas determinar se um dado sistema possui a capacidade mínima, em termos de desempenho, para suportar a aplicação alvo requerida. Sendo assim, sistemas que não atingirem esta capacidade mínima não estarão incluídos na resposta a estas questões.

Os tipos de medidas de desempenho comumente usadas são medidas que analisam **produtividade, utilização** e **tempo de resposta de transações individuais.**

A medida de **produtividade** [3] consiste no número de operações que são executadas em um determinado período de tempo. Esta medida é caracterizada como macroscópica (avalia o desempenho a nível global). Entre as medidas mais populares tem-se *mips* [4] e *mflops* [5] que são ditas serem de baixo nível, mais especificamente para análise de desempenho de processadores, e não podem ser utilizadas em sistemas de análise mais abrangentes visto que a influência gerado pelo processador no desempenho global é parcial, e muitas vezes reduzida. Esta influência dependende de vários outros recursos computacionais, principalmente de dispositivos de entrada/saída. Já as medidas de produtividade *tps* [6] e *tpm* [7] necessitam de uma caracterização precisa do conjunto de transações uma vez que tipos diferentes de transações influenciam de modo diferente o desempenho. Antes de

[3] *throughput.*
[4] Milhões de instruções por segundo.
[5] Milhões de operações de ponto flutuante por segundo.
[6] Número de transações efetuadas por segundo.
[7] Número de transações efetuadas por minuto.

comparar uma medida como *tps*, divulgada entre *benchmarks* distintos, deve-se comparar a caracterização da carga de trabalho total de cada um dos *benchmarks*.

A medida de **utilização** de recursos computacionais serve para detectar possíveis fontes de gargalos decorrentes de recursos computacionais específicos. Esta medida de utilização pode ser reportada através de uma quantificação absoluta (caracterizada por medidas de tempo e de capacidade) ou por uma quantificação relativa [8] (expressada em porcentagem (%)) que indica a freqüência relativa de utilização de um dado recurso computacional.

As medidas de **tempo de resposta de transações individuais** são ditas microscópicas, tais como hora, minuto e segundo. Estas medidas podem ser obtidas externamente através de algum mecanismo de contagem de tempo (cronômetro, por exemplo) ou internamente, através de rotinas que acessam os dados do relógio presente no sistema computacional. Em geral, efetua-se várias vezes a medição do tempo de resposta de uma dada transação e em seguida calcula-se a média aritmética desses tempos, evitando-se a generalização de erros decorrentes de uma única medição. Uma importante caracterização é a definição do **espaço de medição de medidas microscópicas**, ou seja, a definição do início (momento em que o primeiro *byte* é recebido do dispositivo de entrada) e do fim (quando o último *byte* dos resultados gerados é mostrado no dispositivo de saída) do momento de medição. O espaço de medição pode ser caracterizado por externo e interno. O espaço de medição externo leva em consideração, para o resultado de desempenho, tanto o tempo necessário para processamento quanto o tempo necessário para a entrada e saída de dados. Em alguns casos, quando o usuário precisa fornecer dados para a transação, o tempo gasto para este fim deve ser incluído, sendo medido empiricamente ou adicionado ao tempo total de execução da transação de acordo com estimativas estatísticas. Espaços de medição externo são utilizados com transações *on-line*. Por outro lado, o espaço de medição interno envolve apenas o tempo de entrada e saída de dados, sendo usado com transações *batch*.

Vale ressaltar que estas medidas, mesmo sendo complementares (cada uma reporta um tipo particular de desempenho, possibilitando uma visão genérica do desempenho do sistema de análise em questão), não capturam um importante fator presente no processo de escolha de sistema: o **custo**. Geralmente, o custo é disponibilizado na forma de uma razão entre grandezas escalares do tipo custo/desempenho ou desempenho/custo, chamadas de medidas de *custo* x *benefício* ou medidas de *custo* x *desempenho*.

Devido à impossibilidade de se medir alguns tipos de custo [9], uma comparação simplificada torna-se importante. Na prática, os custos reportados são os custos associados

[8] Também chamada de taxa de utilização.

[9] Custos de análise e projeto, de programação, de operação, de manutenção, entre outros.

ao *hardware* , ao *software* e à manutenção para um período de 5 a 6 anos [10]. Esta simplificação não invalida totalmente o processo de medição do custo, pois os fatores medidos correspondem a uma boa parcela do custo final. Ainda, os custos reportados por um *benchmark* devem ser vistos apenas como uma estimativa parcial do custo total, medida exclusivamente para proporcionar comparação entre sistemas de análise distintos.

No processo de escolha de sistemas, além da análise de desempenho e custos associados já comentados acima, outros fatores devem ser levados em consideração, tais como: funcionalidade oferecidas, disponibilidade de *software* e *hardware* compatíveis, escalabilidade do sistema, adequação a padrões consolidados, facilidade de uso e aprendizado etc. *Benchmarks* correspondem a uma ferramenta de auxílio no processo de escolha de sistemas sendo mais adequados na determinação de capacidades e levantamento superficial de custos pois se mostram incompletos na análise de todos os fatores pertinentes ao processo de escolha de sistemas.

3.6 Aplicações da técnica de *benchmark*

A principal utilização da técnica de *benchmark* de banco de dados está relacionada à análise de desempenho. A escolha de um SGBD é uma das muitas situações na qual esta técnica é aplicada.

Uma comparação entre SGBDs utilizando-se *benchmarks* está correta desde que o padrão aplicado em cada um dos sistemas analisados seja o mesmo.

Um fator que deve ser levado em consideração refere-se à execução do *benchmark*, que pode estar tanto a cargo do usuário como dos próprios fornecedores. Quando o próprio usuário executar o *benchmark*, ele podo adquirir condições de conhecer os resultados de desempenho do sistema mais detalhadamente, conhecer aspectos relativos à qualidade do produto em si, tais como: documentação, suporte, facilidade de uso etc. Também, pode-se evitar que os próprios fornecedores alterem o *benchmark* de modo a melhor se adaptar a seus produtos.

Observa-se que os resultados obtidos através de *benchmarks* estão de algum forma restritos às condições em que foram obtidos. Quando se avalia uma determinada versão de um SGBD, não se pode fazer considerações genéricas e extrapolações de resultados sob pena de se cometer sérios enganos, pois cada vez mais novas versões tornam-se disponíveis e apresentam resultados significativos em comparação com a versão anterior.

Em geral, as críticas às metodologias de *benchmark* recaem sobre aspectos controvertidos tais como: a formação do banco de dados, o conteúdo e disposição dos dados e a carga de trabalho utilizada. Em muitos casos, estas críticas se justificam devido ao fato

[10] Período estimado para que o custo do equipamento se deprecie totalmente.

de as características de alguns *benchmark* não serem suficientemente representativas para se avaliar um sistema.

Muitas metodologias de *benchmark* têm sido apresentadas na literatura. Dentre elas, duas merecem destaque especial e são descritas a seguir: a metodologia de Wisconsin, conhecida como *benchmark* de Wisconsin ou *benchmark* de DeWitt e o *benchmark* de Débito e Crédito. Vários outros *benchmarks* foram propostos baseando-se na metodologia destes *benchmarks*. Maiores detalhes podem ser encontrados em [Per90],[Gra91] e [Cha95].

Ainda, a metodologia de [Per90] - Metodologia de análise de desempenho baseada em *Benchmark* Especializado - MBE e o *benchmark* voltado à análise de desempenho de sistemas de informações geográficas de [Cif95] são apresentadas por serem fonte de interesse desta dissertação.

3.6.1 *Benchmark* de Wisconsin

O *benchmark* de Wisconsin (BW) pode ser considerado como um padrão para análise de desempenho de SGBDs relacionais. Entre os sistemas avaliados com este *benchmark* podemos citar: INGRES, ORACLE, IDM500 e SQL/DS. Devido á sua simplicidade, portabilidade e o pouco tempo necessário para a sua execução, este *benchmark* tornou-se largamente adotado por indústrias e laboratórios de pesquisas.

O *benchmark* de Wisconsin apresenta as seguintes características gerais: utilização de um banco de dados e um conjunto de transações sintéticas (ou fictícias), ambiente de execução mono-usuário [11] e *stand-alone* [12].

O principal objetivo do *benchmark* de Wisconsin é procurar obter dados de desempenho do SGBD que sejam independentes da escolha de uma aplicação (*benchmark* de sistema). Neste sentido, são realizados testes sobre blocos básicos de implementação, tais como: soluções de algoritmos de operações de *joins*, mecanismos de indexação, escolha de plano de acesso etc.

O banco de dados de Wisconsin é constituído por 3 relações: *OneKtup*, *TenKtup1* e *TenKtup2*, com 1000, 10000 e 10000 tuplas respectivamente. Cada relação possui 13 atributos do tipo inteiro e de tamanho igual a 2 *bytes* e 3 atributos do tipo *string* de tamanho 52 *bytes*. Todas as relações possuem os mesmos atributos, apenas modificando-se a cardinalidade em cada relação. Na tabela 3.1 mostra-se as construções de cada atributo (domínios). Os valores dos atributos das relações são distribuídos uniformemente, permitindo a obtenção de vários graus de seletividade. Os intervalos de valores de atributos do tipo inteiro são indicados pelo próprio nome, ou seja, two possui um intervalo de 0 a 1,

[11] No ambiente mono-usuário, as transações do *benchmark* são submetidas seqüencialmente, como se apenas um usuário as estivesse executando, internamente ao SGBD.

[12] A característica *stand-alone* significa que não existem outros processos, externos ao SGBD, sendo executados concorrentemente no mesmo sistema computacional.

Nome	Tipo	Intervalo	Ordem	Comentário
uniquel	int	0 - 9999	randômico	chave primária
unique2	int	0 - 9999	randômico	
two	int	0 - 1	cíclico	0,1,0,1,...
four	int	0 - 3	cíclico	0,1,3,0,1,...
ten	int	0 - 9	cíclico	0,1,...,9,0,1,...
twenty	int	0 - 19	cíclico	0,1,...,19,0,1,...
hundred	int	0 - 99	cíclico	0,1,...,99,0,1,...
thousand	int	0 - 999	cíclico	0,1,...,999,0,1,...
twothous	int	0 - 1999	cíclico	0,1,...,1999,0,1,...
fivethous	int	0 - 4999	cíclico	
tenthous	int	0 - 9999	cíclico	0,1,...,9999,0,1,...
old100	int	1 - 99	cíclico	1,3,5,...,99,1,...
even100	int	2 - 100	cíclico	2,4,...,100,2,...
stringu1	char		randômico	chave primária
stringu2	char		cíclico	chave primária
string4	char		cíclico	

Tabela 3.1: Descrição dos atributos das relações do BW.

four de 0..3, ... , hundred de 0..99, thousand de 0...999, assim por diante.

O atributo uniquel é chave primária das relações e assume valores únicos, aleatóriamente dispostos sobre a relação. Para *OneKtup* os valores de uniquel vão de 0..999. Na tabela 3.2 mostra-se um fragmento da relação *OneKtup*. O atributo unique2 também assume valores únicos, mas sua disposição na relação é seqüencial. Para a realização dos testes, são utilizadas relações sem e com agrupamento, tomando o atributo unique2 como referência. Os atributos tipo *string* (stringu1 e stringu2) possuem comprimento fixo de 52 *bytes* e valores únicos; são formados por caracteres significativos nas posições 1, 27 e 52 da cadeia, sendo o restante composto por caracteres não significativos ("x"); desta forma, é possível a formação de até 26^3 combinações de cadeias, o suficiente para preencher a relação *TenKtup*.

Devido à forma como são construídas as relações, mesmo um usuário que não possua um conhecimento prévio sobre elas pode facilmente entender sua estrutura e aprender a utilizá-las. Pode-se construir, com grande facilidade, várias transações com diferentes graus de seletividade. Por exemplo, uma transação em SQL que recupera 10% das tuplas de uma tabela poderia ser:

Unique1	Unique2	Two	Ten	Hundred	Thousand
378	0	1	3	13	615
616	1	1	4	4	695
910	2	0	6	26	313
180	3	0	2	52	74
879	4	0	0	20	447
557	5	1	9	29	847
916	6	0	7	47	247
73	7	1	4	54	455
279	8	0	5	18	437
457	9	1	9	39	747

Tabela 3.2: Fragmento da relação OneKtup.

SELECT * FROM OneKtup WHERE unique1<100

Este *benchmark* é formado por um conjunto de várias transações que incluem transações simples, tais como: recuperação de uma tupla em uma relação ou a busca em um determinado intervalo, agregações (*min* e *sum*), junções de 2 ou 3 relações, projeções, atualizações e inserções. Ao final do *benchmark*, todas as relações conservam as mesmas características iniciais, a mesma cardinalidade e distribuição de valores, pois as tuplas que são inseridas ou alteradas são, posteriormente, atualizadas e removidas. Maiores detalhes sobre o conjunto de transações do *benchmark* de Wisconsin podem ser encontrados em [Dew91].

Algumas críticas ao *benchmark* de Wisconsin:

- Emprega-se apenas dois tipos de dados: numérico e *string*. Os valores numéricos são de 2 *bytes* e os de tipo *string* são de 52 *bytes* com comprimento fixo. De acordo com especialistas, os tamanhos variando de 20 a 30 *bytes* são mais freqüentes em aplicações reais.

- A ausência de operações envolvendo decimais e ponto flutuante. Os efeitos mais importantes de operações deste tipo é a sobrecarga de processamento e a sobrecarga de armazenamento. Tal fato, não é examinado.

- Os testes são executados com valores numéricos de 2 *bytes* e *strings* com 52 *bytes* de comprimento fixo. Torna-se mais realista supor a utilização de *string* com tamanho

variável, o que poderia introduzir um comportamento não uniforme ao banco de dados (número de tuplas por página variável) realizados em ambiente mono-usuário.

- Os testes são realizados em um ambiente *stand-alone*. Os SGBDs rodam, na maioria dos casos, em computadores não dedicados; torna-se importante verificar o efeito da interação do SGBD com os outros processos que estão sendo executados simultaneamente na mesma máquina.

3.6.2 *Benchmark* de Débito e Crédito

O *benchmark* de Débito e Crédito foi desenvolvido visando criar um padrão de transação para classificar o nível de desempenho de sistemas comerciais tipo OLTP [13].

Basicamente, o *benchmark* de Débito e Crédito consiste em três testes multi-usuários sendo o TP1 o mais popular. Resumidamente, o teste TP1 consiste na execução de transações comumente encontradas em aplicações bancárias do tipo depósito e débito em uma conta corrente. A figura 3.3 apresenta as entidades e relacionamentos modelados de acordo com regras pré-estabelecidas para este tipo de aplicação.

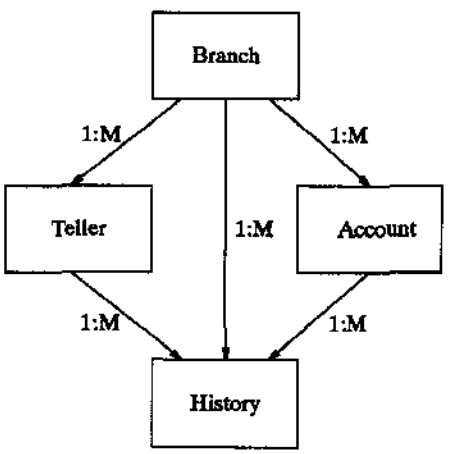

Figura 3.3: Diagrama lógico TP1.

No TP1, proposto originalmente, utiliza-se um terminal conectado ao equipamento principal via X-25. Inicialmente o sistema faz um serviço de apresentação para mapear a entrada para um programa COBOL, que utiliza um sistema de banco de dados para debitar ou creditar um conta bancária. Para realizar um transação deste tipo, é requerido que 95% das transações sejam executadas com tempo de resposta abaixo de 1 segundo. No caso do teste TP1, mede-se o número de *tps* que são executadas pelo SGBD, mantendo-se o requisito de que o tempo de resposta deve estar abaixo de 1 segundo.

[13] On-line transaction processing.

O código abaixo mostra o perfil da transação TP1. Aid(Account_id), Tid(Teller_id) e Bid(Branch_id) são chaves das tabelas correspondentes. Delta é o valor movimentado na conta(Account). O valor de Branch_id, na mensagem de entrada, é o identificador da agência na qual está localizado o caixa automático (Teller).

```
* Transacao TP1

Read 100 bytes including Aid, Tid, Bid, Delta from terminal
BEGIN TRANSACTION
      Update Account where A_ID = Aid:
             Read A_Balance from Account
             Set A_Balance = A_Balance + Delta
             Write A_Balance to Account
      Write to History:
             Aid, Tid, Bid, Delta, Time_stamp
      Update Teller where T_ID = Tid:
             Set T_Balance = T_Balance + Delta
             Write T_Balance to Teller
      Update Branch where B_ID = Bid:
             Set T_Balance = T_Balance + Delta
             Write T_Balance to Branch
COMMIT TRANSACTION
Write 200 bytes including Aid, Tid, Bid, Delta, A_Balance from terminal
```

Além do teste TP1, o *benchmark* de Débito e Crédito possui testes de busca e atualização seqüenciais de registros, e testes de ordenação de um grande número de registros. Mais detalhes com relação a estes testes podem ser encontrados em [Ser91].

Entre as principais críticas ao *benchmark* de Débito e Crédito se colocam:

- Representatividade: o TP1 é um *benchmark* de operação de um cheque bancário (saque ou depósito) onde se assume uma propriedade invariante que é questionável: o sistema mantém constante a quantidade de dinheiro na gaveta de cada caixa, assim como a quantidade em cada agência do banco. Estas hipóteses podem ser facilmente criticadas e a solução adotada pode ser considerada muito simplista.

- Baixa simplicidade: O TP1 não possui transações de consulta, não testa *joins* entre relações, otimização de transações, consultas aninhadas ou funções sobre agregados e nem tampouco inclui controles de lógica ou desvios; ou seja, o TP1 não testa

aplicações reais, é somente um teste de quão rapidamente um SGBD pode processar um grande número de atualizações.

3.6.3 Metodologia de *benchmark* especializado - MBE

A metodologia de análise de desempenho baseada em *benchmark* especializado - MBE foi desenvolvida por [Per90] e seu principal objetivo é tornar-se uma ferramenta de suporte para análise de desempenho de sistemas de banco de dados, podendo modelar convenientemente o ambiente operacional do usuário. Esta metodologia engloba as etapas de concepção, coleta, organização e construção do *benchmark* (banco de dados e carga de trabalho).

Nesta metodologia, o usuário parametriza o *benchmark* a ser obtido de acordo com as características de uma aplicação ou de um conjunto típico de aplicações [14]. Os vários objetos que modelam a aplicação são armazenados em um dicionário de dados centralizado, que representa o modelo do sistema de bancos de dados. Este modelo é sub-dividido em um modelo de representação dos dados e um modelo de carga de trabalho.

No modelo de representação dos dados, o usuário define a tabelas (relações), fornece informações sobre tipo de dados utilizados, número de atributos, distribuição de valores, número de valores distintos etc, complementando as informações obtidas inicialmente na fase de modelagem do banco de dados. A fase de modelagem de dados, por sua vez, utiliza um diagrama de entidades e relacionamentos ([Che83]) onde se caracteriza o esquema do banco de dados.

O modelo de carga de trabalho é representado por um modelo de execução juntamente com um modelo de descrições de transações. No modelo de execução são representadas hierarquicamente os vários níveis de concorrência (externo, interno, de aplicação e de transação) presentes no sistema que se deseja modelar. O nível externo especifica a carga externa ao SGBD representada por processos que são executados concorrentemente no mesmo sistema computacional. O nível interno e o nível de aplicação especificam respectivamente o número de usuários utilizando diferentes aplicações simultaneamente no SGBD e o número de usuários utilizando uma mesma aplicação. No nível de transação podem ser especificadas as freqüências de execução de cada tipo específico de transação, obtendo-se, assim, um perfil de utilização. O modelo de descrição de transações baseia-se em um paradigma de utilização de banco de dados através de formulários ou janelas. Neste paradigma as transações são centradas em uma visão base associada à janela sobre a qual as interações são realizadas e, paralelamente, outras operações são realizadas sobre outras tabelas relacionadas à visão base.

[14] O banco de dados e a carga de trabalho são convenientemente modelados e posteriormente gerados automaticamente por programas que implementam a metodologia.

A metodologia proporciona um ambiente que pode ser facilmente gerado, onde o desempenho do sistema pode ser avaliado convenientemente para cada uma das alternativas sugeridas pelo usuário. Ela pode ser utilizada tanto para selecionar um SGBD através da análise de desempenho, como para auxiliar um projetista de aplicação ou um gerente de sistema a escolher uma boa alternativa de projeto para uma dada configuração de dados e transações. As alternativas podem ser avaliadas em um ambiente de testes, composto de um banco de dados e carga de trabalho, sendo geradas rapidamente pela ferramenta e executadas sobre o SGBD.

Na MBE o usuário tem a liberdade de modificar a estrutura do banco de dados, incluindo novas relações ou modificando as relações já existentes. Além disso, ele tem a possibilidade de modificar os domínios dos dados e a distribuição de valores. Os dados podem ser gerados utilizando uma distribuição uniforme ou uma distribuição não uniforme, tipo Zipf.

A distribuição uniforme única (sem repetição de valores), segue o seguinte algoritmo para a geração de valores:

- Um mapa do tipo *bit-map* é construído com o tamanho igual ao número de valores distintos desejado. A seqüência aleatória de valores é obtida pesquisando-se no *bit-map* através de um valor escolhido randomicamente no intervalo de $1..N$; onde N corresponde ao número de valores distintos.

- Caso o valor escolhido randomicamente já tenha sido escolhido anteriormente, o *bit-map* deve indicá-lo como ocupado. Caso seja a primeira vez, o valor é incluído na seqüência e a posição referente no *bit-map* é marcada como ocupada.

- A pesquisa termina após todos os valores do *bit-map* terem sido escolhidos. Na prática, podemos considerar que após 90% dos valores terem sido escolhidos, os valores restantes da seqüência podem ser obtidos percorrendo-se o *bit-map* em ordem seqüencial, retirando-se os valores não escolhidos e incluindo-os na coluna de valores aleatórios únicos na mesma ordem em que foram encontrados.

Quando deseja-se obter uma distribuição de valores aleatórios únicos para um atributo cujo domínio seja definido em um intervalo entre um valor mínimo e um máximo, percorre-se a coluna de valores aleatórios em ordem seqüencial e seleciona-se os valores da coluna que estão dentro do intervalo desejado.

Ainda, assumindo apenas comportamento uniforme, pode-se obter uma estimativa incorreta da seletividade de um predicado. Desta forma, a utilização de distribuição assimétrica em modelagens de bancos de dados é muito importante. Segundo [Per90], *G. K. Zipf* foi o primeiro a observar e relatar o comportamento de distribuições assimétricas

na utilização de recursos. Observando o comportamento da utilização de palavras [15] em populações de cidades, nomes de pessoas, etc, *G. K. Zipf* formulou a seguinte lei:

" O *ranking*[16] das palavras em um texto é inversamente proporcional à sua freqüência de ocorrência."

Nesta lei, o elemento mais freqüente recebe a posição do topo do *ranking*, ou seja, o valor numérico 1, e o menos freqüente recebe a última posição.

[Per90] apresenta a fórmula abaixo como sendo de grande utilidade na geração de dados assimétricos. Detalhes de como a fórmula é obtida podem ser encontradas no trabalho do autor.

$$rank^z \times w = 1/constante$$

O valor de w corresponde à freqüência do elemento, z é chamado de fator de decaimento. Dependendo do valor de z a distribuição pode se tornar mais ou menos assimétrica. A tabela 3.3 mostra os valores de freqüência para 10 valores distintos em distribuições do tipo *Zipf* com fator de decaimento z iguais a 0, 0.5, 1 e 3. Pode-se observar que a distribuição se torna mais assimétrica na medida em que z aumenta. O valor da constante pode ser calculada pelo e-nésimo harmônico de ordem z:

$H_n^z = \sum_{k=1}^{n} 1/k^z$, onde n é igual ao número de valores distintos.

Ainda, na metodologia de [Per90], foram analisadas várias situações de geração de dados convencionais combinando as características de distribuição dos dados apresentadas acima, disposição na relação, unicidade etc. Estes algoritmos são citados em seguida.

1. Geração/Distribuição de valores numéricos aleatórios únicos e com disposição aleatória na relação.

2. Geração/Distribuição uniforme de valores numéricos aleatórios, com disposição também aleatória.

3. Geração/Distribuição de valores do tipo *string* com valores únicos, gerados aleatoriamente, com tamanho fixo ou variável, e disposição dos dados aleatória. Para atributos do tipo *string*, o seguinte formato é adotado na sua geração: $XY_1Y_2..Y_nZ_1Z_2..Z_m$. A primeira parte é composta por um caracter fixo X. A segunda parte é preenchida com um valor numérico e a terceira parte, que pode possuir tamanho fixo ou variável,

[15] Um exemplo resultante do livro Ulysses de James Joyce mostra que das 260.430 palavras existentes, as 135 palavras mais freqüentes entre as 26.899 palavras distintas, correspondem a 50% da contagem total das palavras.

[16] Ordenação de acordo com a freqüência de ocorrência de cada elemento distinto.

rank	freqüências			
	z = 0	z = 0.5	z = 1	z = 3
1	0.1	0.1992	0.34	0.8351
2	0.1	0.1408	0.17	0.1044
3	0.1	0.1150	0.11	0.0309
4	0.1	0.0996	0.09	0.0130
5	0.1	0.0891	0.07	0.0067
6	0.1	0.0813	0.06	0.0038
7	0.1	0.0753	0.05	0.0024
8	0.1	0.0704	0.04	0.0016
9	0.1	0.0664	0.04	0.0012
10	0.1	0.0629	0.03	0.0009
total	1.0	1.0000	1.00	1.0000

Tabela 3.3: Exemplos de uma distribuição do tipo *Zipf*.

é formada pela concatenação de caracteres não significaticos com o intuito apenas de formar o tamanho final desejado para o atributo. O valor numérico n representa o número de valores distintos que o atributo pode assumir.

4. Semelhante à anterior, exceto que, neste caso, os valores são distribuídos uniformemente, ou seja, pode haver repetições de valores.

5. Geração/Distribuição para gerar uma seqüência de valores numéricos que se inicia no valor mínimo definido no domínio do atributo; o valor máximo corresponde ao número de ocorrências que a relação deve possuir.

6. Geração/Distribuição para gerar uma seqüência de valores com uma ordem seqüencial.

7. Geração/Distribuição para gerar valores do tipo *string* conforme uma distribuição do tipo *Zipf*. Os valores podem ser obtidos automaticamente a partir das descrições dos atributos ou, então, podem ser fornecidos pelos usuários em uma lista de valores. Para os valores obtidos automaticamente, utiliza-se o item quatro (4) comentado acima. Para obter-se a distribuição de *Zipf* as freqüências (W_i) de cada elemento distinto são calculadas de acordo com a fórmula apresentada anteriormente e inseridos na relação em posições aleatórias.

8. Geração/Distribuição numérica aleatória sem qualquer controle de unicidade.

De maneira sintética, esta metodologia permite: avaliar o desempenho de diferentes SGBDs; avaliar diferentes soluções de projeto (escolha de índices e agrupamentos, normalizações etc); analisar o desempenho do sistema para diferentes distribuições de dados; testar o efeito de diferentes níveis de concorrência no sistema e testar diferentes composições de carga de trabalho com diferentes níveis de concorrência.

Algumas críticas a MBE podem ser levantadas:

- Ampliação da caracterização dos dados,ou seja, estabelecer correlações entre atributos e proporcionar novos tipos de dados que ocorram usualmente em aplicações reais (datas, tempo etc).
- Análise dos resultados produzidos pois o trabalho concentrou-se no desenvolvimento da metododologia (MBE) e na construção de uma ferramenta que implementa a metodologia.

3.6.4 *Benchmark* voltado a SIGs

Ciferri ([Cif95]) propõe em seu trabalho um conjunto de transações primitivas que representam a carga de trabalho de um *benchmark* voltado à análise de desempenho de sistemas de informações geográficas, além de caracterizar aspectos relacionados aos dados.

A carga de trabalho do *benchmark* é composta por um conjunto de transações primitivas, especificadas em alto nível, que podem ser utilizadas para a formação de transações mais complexas. Estas transações primitivas propostas são predominantemente orientadas aos dados não convencionais, sendo independentes do formato de dados utilizado (*raster* ou vetorial). Ainda, algumas transações definidas são voltadas especificamente para o formato *raster*, ou utilizam ambos formatos conjuntamente. A carga de trabalho proposta é baseada em transações realizadas durante a fase de execução do sistema, ou seja, não são consideradas, por exemplo, operações realizadas durante a fase de captura de dados.

A definição da carga de trabalho de um *benchmark* voltado à análise de desempenho de SIGs requer um conjunto abrangente de transações. Algumas destas transações são: reclassificação, superposição, análise de ponderação, análise de proximidade simples, múltipla e ao redor de múltiplos objetos geográficos, decomposição de objetos geográficos bidimensionais, transações topológicas booleanas, busca topológica, transações baseadas em conjuntos e transações de conversão de formato de dados.

A caracterização do dados do *benchmark* é efetuada em termos dos tipos de dados necessários para a representação de aplicações georeferenciadas. Esta caracterização é independente da forma de organização dos dados (estrutura de arquivos, quantidade de dados armazenados, entre outros), de modo a permitir a execução da carga de trabalho do *benchmark* segundo aplicações georeferenciadas específicas.

Os principais tipos de dados a serem considerados por um *benchmark* voltado a SIG são:

- **dados convencionais**: dados comumente encontrados em SGBDs convencionais. *Integer, long integer, string, date*, entre outros.
- **célula**: dado armazenado no formato *raster* (quadrático/retangular), correspondente a uma área específica do espaço geográfico. Para este tipo de dado deve-se associar uma categoria e armazenar as suas coordenadas centrais x e y. A derivação de coordenadas geográficas é efetuada de acordo com a posição das células na matriz, as quais possuem implicitamente associada à sua posição uma localização geográfica.
- **ponto**: dado espacial de dimensão zero armazenado no formato vetorial. Para este tipo de dado deve-se armazenar as suas coordenadas x e y.
- **linha**: dados espaciais unidimensionais armazenados no formato vetorial. Uma conexão retilínea entre dois pontos é denominada segmento de linha. A interligação de vários segmentos de linha forma uma linha. Deve-se armazenar para cada linha, ordenadamente, as coordenadas de todos os pontos que a compõem, sendo necessário fixar o número de pontos, orientação e comprimento.
- **polígono**: dado espacial bidimensional armazenado no formato vetorial. Para este tipo de dado deve-se armazenar ordenadamente as coordenadas de todos os pontos que o compõem. Deve-se fixar o número de pontos (vértices), orientação e área.

A descrição destes tipos de dados serve apenas como referência, tendo por objetivo padronizar o conteúdo dos dados presentes no *benchmark*. A troca de qualquer um destes tipos é permitida desde que seja por um tipo mais abrangente (que englobe o intervalo de variação do tipo de dados trocado). Dados gráficos não são considerados pelo *benchmark*.

Para se permitir a comparação de desempenho entre SIGs distintos, foi definida uma aplicação alvo na qual a carga de trabalho do *benchmark* proposto pode ser executada. Esta aplicação é composta de dados sintéticos, sendo voltada para representar uma aplicação de grande porte.

No caso, na geração de pontos, linhas e polígonos, dois conceitos foram utilizados: *extent* e *division*. Um *extent* consiste na área total abrangida pela aplicação (coordenadas máximas). Uma *division* consiste em uma área contida inteiramente no *extent*.

Para a geração de pontos gera-se aleatoriamente as coordeandas x e y de tal forma que seus valores estejam contidos no interior da *division*.

Já a geração de linhas no formato vetorial é efetuada com base nos fatores forma geométrica, tamanho e complexidade. Seguiu-se o seguinte algoritmo:

- Gera-se aleatoriamente um ponto contido no interior da *division* que será o primeiro ponto final da linha.

- Traça-se um círculo de raio igual ao tamanho requerido a partir do primeiro ponto gerado. Escolhe-se um ponto contido nos limites do círculo que esteja dentro da *division* e respeite a forma geométrica escolhida. O ponto escolhido consiste no segundo ponto final da linha. Caso isto não seja possível, descarta-se o primeiro ponto gerado e inicia-se novamente a seqüência.

- Para cada linha gerada cujo número de pontos requerido seja maior que dois, gera-se aleatoriamente n pontos ao longo dos pontos finais (na reta que liga o primeiro e segundo ponto final da reta), de modo que estes não ocupem as mesmas coordenadas.

A figura 3.4 ilustra exemplos gerados a partir do algoritmo acima. No primeiro exemplo (a esquerda), para uma *division* pode-se gerar linhas com forma geométrica aleatória, tamanho aleatório no intervalo [10%,90%] e número fixo de pontos igual a 2, por outro lado, no segundo exemplo, gera-se linhas com forma geométrica com orientação horizontal, tamanho aleatório no intervalo [20%,80%] e número de pontos aleatórios no intervalo [2,5].

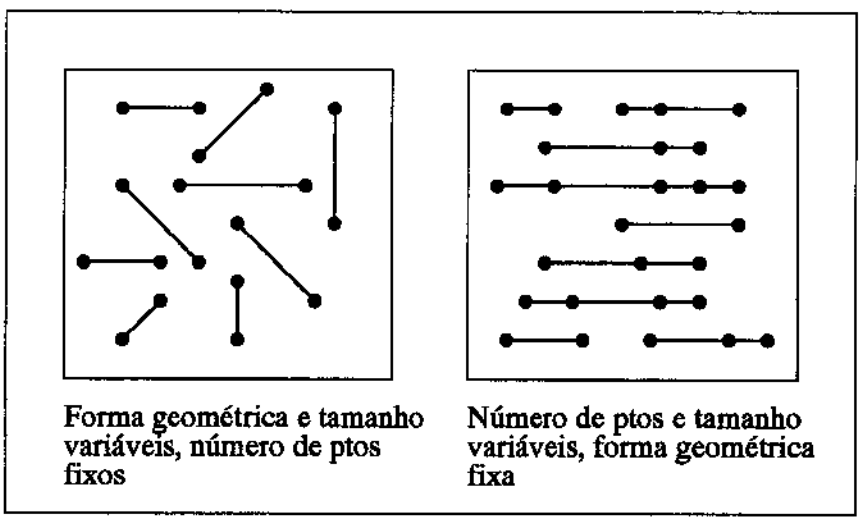

Figura 3.4: Exemplos de variações de forma geométrica, tamanho e complexidade na geração de linhas.

A geração de polígonos é efetuada com base nos fatores forma geométrica, tamanho e complexidade, semelhante a linhas.

Em [Cif95], a forma geométrica visa caracterizar a orientação dos polígonos (figura 3.5). Cada tipo de forma geométrica gera sempre dois polígonos, de mesma área e formato. Definidos a quantidade e a forma geométrica dos polígonos na *division*, o tamanho

destes é automaticamente determinado, isto se deve porque quanto maior a quantidade de polígonos em uma *division*, menor será o tamanho dos polígonos. A complexidade corresponde ao número de pontos que os polígonos gerados terão. Para polígonos cujo a forma geométrica corresponde ao conjunto A, o número mínimo de pontos é 4 por polígono, sendo estes distribuídos nos extremos dos dois retângulos formados pela forma geométrica. Para polígonos cuja forma geométrica corresponde ao conjunto B, o número mínimo de pontos é 3 por polígono, sendo estes distribuídos nos extremos dos dois triângulos formados pela devida forma geométrica. Para polígonos cujo a forma geométrica corresponde ao conjunto C, o número mínimo de pontos é 6 por polígono, sendo estes distribuídos nos extremos de cada reta formada pela devida forma geométrica. A figura 3.5 ilustra o número mínimo e a distribuição dos pontos de acordo com as formas geométricas dos conjuntos A, B e C.

Ainda, segundo [Cif95], a complexidade de um polígono pode ser fixa ou aleatória. No primeiro caso, fixa-se um número inteiro que corresponde ao número de pontos de todos os polígonos da *division*. No segundo caso, fixa-se um intervalo no qual o número de pontos deve ser gerado, tal como [6,12]. Deve-se observar, no entanto, o número mínimo de pontos que cada forma geométrica requer.

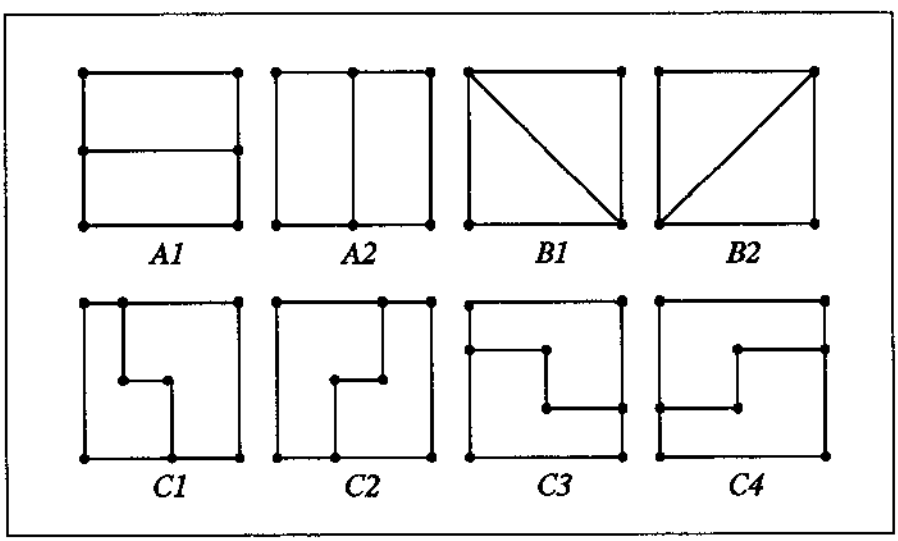

Figura 3.5: Forma geométrica de polígonos e suas respectivas distribuições de pontos.

Os passos para a geração de polígonos são:

- Determina-se a quantidade de polígonos de uma *division*. Esta pode ser nula, ou uma potência - 2^k, sendo que k deve ser um número inteiro, ímpar e maior que zero (*0*). Caso a quantidade escolhida seja nula, não gera-se nenhum polígono para a *division*. Para uma quantidade de polígonos não nula, divide-se a *division* segundo

o modelo *quadtree* em 2^{k-1} partes iguais. Para cada uma destas partes gera-se dois polígonos segundo a forma geométrica escolhida.

- Determina-se a complexidade dos polígonos aos pares, para os polígonos gerados a partir de uma mesma forma geométrica. Desta forma, distribui-se somente uma vez o número de pontos excedente (observando-se o número mínimo de pontos por polígonos) ao longo da divisa destes polígonos (excetuando-se os pontos finais da divisa). A complexidade de um polígono pode ser fixa ou aleatória como comentado anteriormente. Em relação às formas geométricas pode-se escolher qualquer combinação com o cuidado de que, quando várias formas geométricas diferentes são permitidos, deve-se escolher aleatoriamente uma das formas geométricas do conjunto.

A figura 3.6 ilustra a geração de polígonos para duas *divisions* distintas com número de polígonos igual a oito (8). Na primeira (à esquerda) gera-se somente polígonos com forma geométrica $B2$, com número de pontos aleatórios no intervalo [4,9]. A segunda, gera-se aleatoriamente as formas geométricas dos polígonos (usando os conjuntos A, B, C), todos com um um número fixo de 7 pontos.

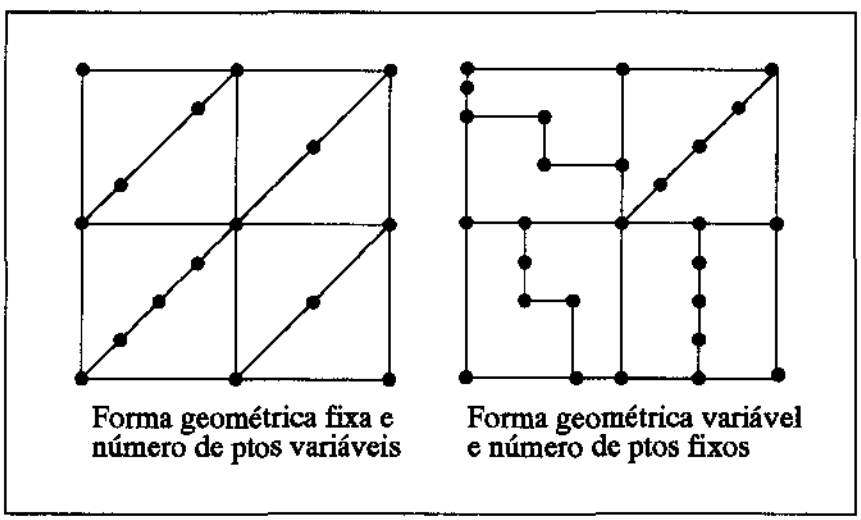

Figura 3.6: Exemplos de variações de forma geométrica e número de pontos na geração de polígonos.

Quanto a dados *raster* [17] a distribuição pode ser baseada exclusivamente na porcentagem que cada categoria ocupa no *extent*. Deste modo, a seletividade de dados *raster*

[17]Dados *raster* são predominantemente usados para representar dados temáticos e dão ênfase no conteúdo de áreas geográficas mais que nos seus limites.

é controlada utilizando-se as porcentagens aplicadas às categorias. Em seguida, deve-se fixar a resolução espacial das células e os limites do *extent*.

Segundo [Cif95], o uso de porcentagens deve ser feito de modo a se produzir quantidades inteiras, evitando-se assim aproximações. Um método simples de se garantir a geração correta de porcentagens de cada categoria segue abaixo:

- Fixa-se a quantidade de células por tipo de categoria, em função do tamanho do *extent* e da porcentagem de ocupação. Cada categoria distinta deve ocupar um nó separado em um fila encadeada. Os nós desta fila possuem além da categoria, a quantidade de células desta categoria.

- Gera-se aleatoriamente um número k no intervalo de $1..n$, onde n representa o número de nós presentes na fila. Inicialmente, n corresponde à quantidade de categorias distintas do tema.

- Procura-se na fila o K-éssimo nó e associa-se para a célula em questão o valor da categoria deste nó. Em seguida, decrementa-se a quantidade de células deste nó de um (1). Quando a quantidade de células de um nó atingir o valor zero (0), este deve ser removido da fila. Repete-se os passos anteriores até que nenhum nó exista na fila.

Outra forma de se analisar dados *raster*, de modo que sejam gerados objetos geográficos (2D) bem definidos em todo *extent*, consiste em executar uma transação de conversão de dados - Vetor/Raster. A seletividade dos dados *raster* é derivada diretamente do tamanho dos polígonos convertidos, sendo no entanto sujeita a aproximações decorrentes do processo de conversão.

Algumas críticas levantadas:

- Representatividade: um grande número de transações foram levantadas junto a diferentes aplicações, obtendo-se resultados genéricos; a carga de trabalho do *benchmark* proposto poderia ser aplicado segundo aplicações georeferenciadas específicas e assim obter resultados mais próximos das aplicações dos usuários. Também, poder-se-ia utilizar as transações primitivas para testar a representatividade de linguagens de consulta.

- Integração de dados: o enfoque é sobre dados geográficos sendo que aplicações georeferenciadas trabalham tanto com dados convencionais como dados não convencionais, poder-se-ia acoplar ao modelo proposto, para a geração de dados não convencionais, modelos voltados à geração de dados convencionais.

3.7 Conclusões

Neste capítulo apresentou-se conceitos relacionados à análise de desempenho.

Dentre os modelos existentes de análise de desempenho, esta dissertação enfoca o modelo experimental e a técnica de *benchmark* de banco de dados.

O modelo experimental procura utilizar o próprio sistema de banco de dados para se obter os resultados de desempenho. A técnica de *benchmark* de banco de dados consiste em um modelo de análise experimental onde é executado um conjunto fixo de testes (transações) sobre um sistema conhecido para se avaliar seu desempenho.

Algumas aplicações desta técnica de *benchmark* de banco de dados foram apresentadas. Dentre os trabalhos citados, destaca-se o trabalho de [Per90] que sugere a caracterização de um *benchmark* de acordo com as características de uma aplicação, ou conjunto típicos de aplicações, segundo às necessidades do usuário. Trata-se, portanto, de um *benchmark* de aplicação onde o *script* gerado não é representativo para qualquer tipo de aplicação, mas reflete as características próprias de uma aplicação, ou conjunto de aplicações. Todo seu trabalho é voltado a dados convencionais. Por outro lado, destaca-se o trabalho de [Cif95] que enumera um grande número de transações e mecanismos de geração de dados voltados a SIGs. Ambos autores trabalham com dados sintéticos.

No próximo capítulo, é apresentado um esquema de caracterização de sistemas de banco de dados espaciais para análise de desempenho utilizando dados sintéticos que melhor representam aplicações reais, de acordo com as propostas de [Per90] e [Cif95]. Procura-se com isto validar os trabalhos/extensões desses autores e derivar mecanismos adequados de proceder à caracterização de uma aplicação real e gerar dados sintéticos. Esta caracterização visa auxiliar a validação de propostas e estudos na área de análise de desempenho de banco de dados espaciais.

Capítulo 4

Esquema de caracterização do banco de dados e da carga de trabalho

4.1 Introdução

Aplicações SIGs são encontradas em muitas áreas de atuação (capítulo 2 - seção 2.5). De forma geral, estas aplicações têm em comum a necessidade de manipularem tanto dados convencionais como dados não convencionais, fomentando a existência de SGBDs cada vez mais adequados a estas aplicações. O desempenho destes SGBDs exerce papel relevante para o sucesso comercial dessas aplicações.

No que se refere a dados convencionais, [Per90] propõe ampliar a abrangência do uso da técnica de *benchmark* para análise de SGBDs através de uma Metodologia de *Benchmark* Especializada - MBE. Já [Cif95] propõe a carga de trabalho e a caracterização dos dados de um *benchmark* voltada à análise de desempenho de SIGs, levando em consideração características especiais de aplicações que utilizam SIG. Estes dois autores baseiam-se na utilização/geração de dados sintéticos.

Em relação à análise de desempenho, a técnica de *benchmark* (capítulo 3 - seção 3.4) quando aplicada a SGBD consiste na execução de um conjunto conhecido de transações, ou carga de trabalho como é comumente chamado, sobre um banco de dados também conhecido ([Cif95], [Per90] e [Gra91]).

Este capítulo apresenta um esquema de caracterização de sistemas de banco de dados espaciais para análise de desempenho utilizando dados sintéticos, tomando-se como base os modelos propostos por [Per90] e [Cif95]. O objetivo é a caracterização do esquema e conteúdo do banco de dados o mais próximo possível de uma aplicação real, ou conjunto de aplicações. Além disso, paralelamente, orienta-se como as mesmas caracterizações podem ser obtidas em relação a aplicações reais SIGs.

Em seguida, algumas características que foram detectadas em um estudo sobre aplica-

ções georeferenciadas reais implantadas em instituições localizadas no território brasileiro e que vão ao encontro desta dissertação são citadas [Cif95]:

- verificou-se o uso de distintos SIGs no Brasil. Entre os SIGs mais utilizados estão Arc/Info, Spring, SGI/Inpe, Idrisi, Vision*, Grass, Microstation, Intergraph, Apic, MapInfo.

- verificou-se a necessidade de armazenamento dos três tipos de dados geográficos (espaciais, convencionais e gráficos) para a maioria das aplicações.

- verificou-se uma quantidade de poucos *Mbytes* de dados geográficos armazenados em aplicações de pequeno porte, até 5 *Mbytes* em aplicações de grande porte. Nestas últimas aplicações, o banco de dados geográfico de cada uma destas ainda está em fase de formação, sendo que projeções de até 30 *Gbytes* de dados geográficos estão previstos na conclusão da aplicação.

- verificou-se um uso predominante de dados no formato *raster* (quadrático) para o modelo baseado em campos, e um uso predominante de dados no formato vetorial segundo os modelos Arc/Info, DIME e de objetos relacionais para o modelo baseado em objetos. Os formatos de dados *raster* e vetorial foram detectados conjuntamente na maioria das aplicações.

- verificou-se que os dados geográficos, principalmente os dados convencionais, estão armazenados predominantemente segundo o modelo relacional. A utilização do paradigma de orientação a objetos, apesar de ser pouco utilizado em sistemas reais, foi constatada inclusive em uma aplicação de médio-grande porte. Além disso, verificou-se que a modelagem conceitual, quando efetuada, segue em maioria o modelo entidade-relacionamento. Em poucas aplicações constatou-se o uso do modelo entidade-categoria-relacionamento, e em somente uma aplicação constatou-se o uso de um modelo de dados geográficos desenvolvido especialmente pelo grupo de desenvolvimento da aplicação.

- verificou-se o uso predominantemente de plataformas baseadas em estações de trabalho (IBM Risc/6000, Sun Sparc, HP-Apolo, entre outras), com sistema operacional UNIX (Aix, SunOS, Solaris e HP-UX, entre outros). Também, constatou-se uma grande parcela de plataformas baseadas em PCs. A quantidade de memória principal presente nas configurações de *hardware* variou de 4 Mbytes em PCs até 128 *Mbytes* em estações de trabalho.

- para a maioria das aplicações, verificou-se uma necessidade real do SIG oferecer algum modo de se formar consultas envolvendo operadores espaciais, geométricos,

topológicos e convencionais conjuntamente. Isto é efetuado basicamente através de uma linguagem de consulta proprietária do SIG subjacente (freqüentemente baseada em SQL), ou por uma estrutura de menus. O resultado de consultas ficou dividido entre a dependência e a independência em relação à sua visualização.

- dentre as funções típicas de um SGBD, somente as funções de consulta e inserção foram tidas como freqüentes por todas as aplicações consultadas, caracterizando um ambiente com poucas alterações nos dados (em parte devido ao alto custo e tempo para se realizar a fase de coleta de dados).

- para a maioria das aplicações, verificou-se a necessidade de funções analíticas (descritas na seção 4.4).

De acordo com as informações acima, a caracterização do esquema do banco de dados apoiou-se no modelo de dados relacional e no modelo entidade-relacionamento - MER para discutir as características a serem obtidas junto à aplicação. Estes modelos, além de serem largamente utilizados em aplicações comerciais, são de fácil entendimento. Acredita-se que outros modelos poderiam ser empregados sem prejuízo das idéias básicas propostas.

4.2 Esquema do banco de dados

No desenvolvimento de uma aplicação, procura-se obter, a partir do mundo real, apenas a parte de interesse da aplicação a ser construída. Este processo se faz através das características estáticas [1] das aplicações que se preocupam em descrever os dados e esquema do banco de dados.

Na caracterização de uma aplicação o mais próximo possível de uma aplicação real, estas características devem ser identificadas sobre uma documentação e um banco de dados já existentes. A documentação pode facilitar a obtenção dessas características, porém é interessante validá-las junto ao banco de dados (poder-se-ia ter resultados não adequados no processo de análise de desempenho se estas informações não estiverem coerentes). Deste modo, para auxiliar na obtenção dessas características, utiliza-se um modelo de representação de dados estruturado em dois níveis de abstração: lógico e físico.

Em nível lógico o interesse está na especificação das estruturas/características mais abstratas que não são representadas diretamente sobre o SGBD, mas que podem ser transformadas e refinadas em informações mais concretas no nível físico. No processo de construção da aplicação tais características tornam-se importantes, podendo influenciar a qualidade da aplicação sendo analisada. Como comentado anteriormente, nesta fase

[1] Propriedades verdadeiras todo o tempo.

adota-se o modelo de entidade-relacionamento para discutir as características que são obtidas na fase de modelagem lógica do banco de dados.

A modelagem utilizando o MER permite a identificação dos objetos ou entidades como são chamadas neste modelo, e os relacionamentos entre estas entidades. Uma entidade é representada por um conjunto de atributos. Cada atributo possui um domínio de valores permitidos. Relacionamentos podem existir entre uma ou mais entidades e também podem possuir atributos.

Nesta dissertação, as entidades podem ser classificadas em convencional e não convencional (ponto, linha ou polígono).

Certas restrições de integridade podem ser definidas sobre o MER e devem ser obedecidas na obtenção dos valores para as entidades no banco de dados.

Uma restrição importante é a cardinalidade de mapeamento, que expressa o número de entidades ao qual outra entidade pode estar associada via um relacionamento. A cardinalidade de mapeamento pode ser: um-para-um (1:1), muitos-para-um (N:1), um-para-muitos (1:N) e muitos-para-muitos (N:M). Outra classe de restrições importante é dependência existencial, ou seja, quando x é existencialmente dependente de y, significa que, quando y for removido, então x também deverá ser removido.

Após obter esta caracterização/representação dos dados a nível mais semântico, devemos verificar como estas informações são mapeadas em uma representação esquemática no SGBD existente, ou seja, deve-se garantir que o esquema de entidades e relacionamentos obtido esteja de acordo com a linguagem de esquemas disponíveis no SGBD que está sendo avaliado.

Neste momento, uma das tarefas importantes é a identificação de chaves: chaves primárias e estrangeiras. Uma chave primária identifica uma ocorrência em uma entidade ou em um relacionamento entre entidades unicamente. As chaves primárias podem ser formadas por apenas um atributo (chave simples) ou podem ser formadas por mais de um atributo (chaves compostas). Uma chave estrangeira é formada por um atributo (ou composição de atributos) de uma entidade cujos valores devem pertencer ao mesmo domínio dos valores de uma chave primária de uma outra entidade à qual a entidade está relacionada.

Também, para cada atributo deve ser definido o seu domínio de valores. Alguns atributos podem possuir domínios distintos ou então compartilhar o mesmo domínio.

Vale ressaltar que no processo de conversão das características encontradas no nível lógico no modelo relacional, as entidades são representadas por relações e os atributos das entidades são representados por atributos das relações, ou seja, o banco de dados é representado por um conjunto de tabelas [2] (ou relações) onde cada tabela é composta de

[2] O conceito de uma tabela envolve noções bastante simples e intuitivas, que facilitam o usuário a definir seu banco de dados. Além disso, existe uma correspondência direta entre o conceito de tabela e o

linhas (ou tuplas) e de colunas (ou atributos). Ainda, os relacionamentos, de acordo com a sua cardinalidade, poderão ser traduzidos em novas relações. No caso de relacionamento N:M, utiliza-se chaves estrangeiras correspondentes às chaves primárias das relações definidas pelo relacionamento e no caso de relacionamentos N:1 e 1:N, utiliza-se uma chave estrangeira incluida em alguma das relações que participam do relacionamento.

Em relação ao nível físico, as informações obtidas no nível lógico são utilizadas para caracterizar o banco de dados. Neste nível, deve-se verificar junto à aplicação real se o banco de dados corresponde às características levantadas no nível lógico. Esta verificação deve ser realizada sobre o banco de dados da aplicação e pode-se utilizar qualquer linguagem de consulta suportada pelo SGBD. Além disso, o objetivo é obter valores para as relações (seus atributos), contidas no esquema obtido anteriormente, através da caracterização mais detalhada dos dados, em termos dos domínios relacionados e de descrições suplementares dos atributos: a distribuição dos dados, a disposição sobre as relações, a ordem de colocação no agrupamento (se ele participa de algum) e se o atributo aceita valores nulos. Além das informações sobre os dados, as estruturas de índices e dos agrupamentos devem também ser informadas.

Problemas com a escolha de atributos na composição dos índices (no caso de existirem índices compostos) não são nada triviais. Uma boa escolha de índices pode acarretar um melhor desempenho do SGBD quando no processamento de uma transação ou simplesmente desperdiçar espaço em memória secundária e, além disso, encarecer a execução das operações de inclusão ou atualização no banco de dados, por exemplo. Neste trabalho, o problema da escolha dos índices (determinação dos atributos que devem possuir índices e quais participam de um mesmo índice e em qual ordem) fica a critério do próprio usuário.

De acordo com o apresentado, ao final desta fase deve-se identificar:

- Entidades (relações) que compõem o banco de dados.
- Tipos das entidades.
- Relacionamentos entre as entidades.
- Atributos das entidades.
- Domínios dos atributos e domínios em comum.
- Identificação das chaves primárias.
- Identificação das chaves estrangeiras.
- Identificação das restrições de mapeamento (propriedades).

conceito matemático de relação.

Ainda, vale ressaltar que a seqüência apresentada aqui não precisa ser rigidamente seguida. Em aplicações reais, nem sempre há a existência de documentação da fase inicial de desenvolvimento (nível conceitual/lógico). Sendo assim, temos que identificar as características mencionadas acima diretamente em nível físico, ou seja, sobre o banco de dados da aplicação.

4.3 Esquema de análise de distribuição espacial dos dados.

Para a caracterização do conteúdo do banco de dados voltado a SIGs, o ambiente da aplicação deve ser considerado. Aplicações SIGs têm em comum o fato de seus bancos de dados estarem distribuídos entre as coordenadas máximas de um sistema de coordenadas, ou seja, coordenadas ($x_{inicial}$,$y_{inicial}$) e (x_{final}, y_{final}), para uma aplicação que considera o sistema de coordenadas xy.

Ainda, estas aplicações manipulam tanto dados convencionais como dados não convencionais. Em relação a dados não convencionais, leva-se em consideração na caracterização do conteúdo do banco de dados, além do seu domínio (valores que podem ser gerados para as coordenadas x e y), aspectos relacionados à geometria do dado espacial, tais como: forma geométrica [3], tamanho [4] e complexidade [5].

Para a análise destes fatores, os conceitos de *extent* e *division* são considerados. Um *extent* consiste na área total abrangida pela aplicação (coordenadas máximas). Uma *division* consiste em uma área contida inteiramente no *extent*. Ambos, *extent* e *division*, podem possuir inúmeras formas de geometria. Entretanto, considera-se nesta dissertação a forma quadrática na geração dos valores de análise.

Desta forma, os dados devem ser analisados de acordo com disposições específicas de *divisions* possibilitando, assim, controlar a localização e a seletividade dos dados. De maneira geral, assumindo uma divisão hierárquica do *extent* segundo o modelo *quad-tree*, mantém-se o formato quadrático para todas as *divisions* do *extent*, o que facilita a formação de *divisions* e simplifica os algoritmos de análise de distribuição de dados. A figura 4.1 ilustra um *extent* contendo um conjunto específico de 40 *divisions*.

Sendo assim, para controlar a seletividade de cada tipo de dado vetorial, basta determinar a sua quantidade para cada *division* de um *extent* obtendo-se o valor $x\%$ do total de dados do *extent*, ou seja:

$x = (total\_de\_dados\_na\_division / total\_de\_dados\_no\_extent) * 100.$

[3] Chamado de *shape* no trabalho de [Cif95].

[4] Comprimento para linhas e área para polígonos.

[5] Número de pontos que compõem a geometria.

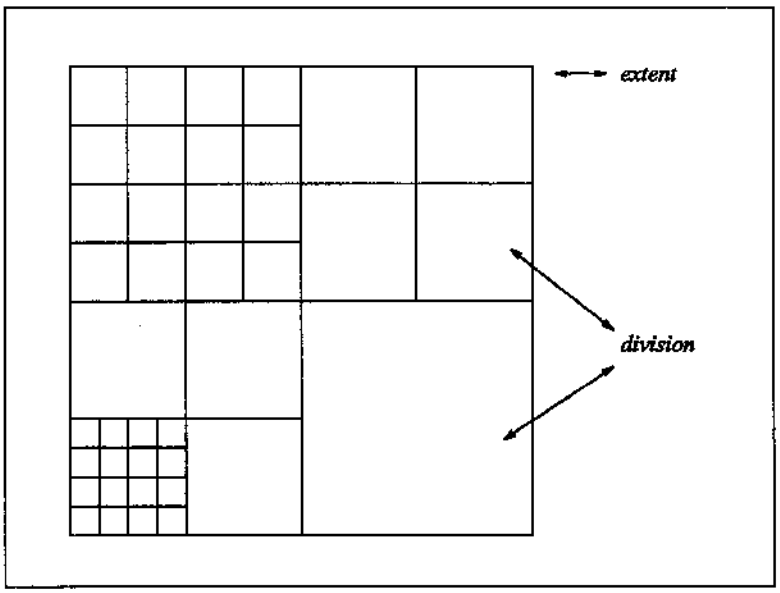

Figura 4.1: Divisão hierárquica do *extent*.

Ao analisar a distribuição de linhas perante uma aplicação real, os fatores forma geométrica, tamanho e complexidade podem variar de acordo com a aplicação e com as características próprias do objeto geográfico (entidade). Pode-se ter entidades cuja característica marcante seja o tamanho enquanto que, em outras, seja a complexidade ou mesmo a forma geométrica, ou a combinação destas. Porém o interesse está na caracterização destas entidades e, sendo assim, pode-se proceder de duas maneiras.

A primeira delas seria utilizar o conceito de *extent* e *division* como comentado anteriormente (utilizando o método *quadtree*) e assim obter a caracterização de todo o banco de dados. Tal fato caracterizaria cada região do *extent* (*divisions*) do banco de dados e poderia ser feito de maneira geral (englobando todas as entidades pertencentes àquela aplicação em estudo), ou em relação a uma entidade específica.

Por outro lado, na análise de uma aplicação, pode surgir o interesse de se caracterizar apenas parte do *extent* dessa aplicação (devido ao tamanho do *extent*, regiões mais concentradas etc). Neste caso, uma segunda maneira seria utilizar ainda o conceito de *extent* e *divisions* porém especificando a região de pesquisa. A caracterização, a partir daí, se procederia de maneira análoga à anterior.

Ainda, tem-se que observar que o conceito de *division* poderá influenciar nos resultados obtidos uma vez que, provavelmente, linhas poderão ultrapassar a definição da *division* sendo considerada. Em geral, para aplicações que utilizam dados sintéticos no processo de geração/análise dos dados descarta-se tal possibilidade. Neste caso, o problema deve ser resolvido de acordo com a especificação de qual procedimento deve ser adotado durante a análise da aplicação real:

- considera-se qualquer ocorrência da entidade incluída ou que "passa" pela *division* em questão, ou
- considera-se apenas aquelas ocorrências da entidade totalmente incluída na *division*.

No processo de análise do *extent*, critérios para determinar o término de iterações devem ser adotados. Poderia ser adotado o tamanho do lado de cada *division* (determinaria uma área mínima baseada nos lados), a área da mesma, a seletividade dos dados em cada *division*, ou mesmo a combinação destes. Dependendo do critério adotado, um maior ou menor refinamento das distribuições dos dados pode ser verificado.

Um exemplo é apresentado abaixo para consolidar os conceitos apresentados. Os dados utilizados fazem parte de um subconjunto do projeto SAGRE - Sistema Automatizado de Gerenciamento de Rede Externa - Telebrás S/A, que será abordado no próximo capítulo.

A figura 4.2 ilustra a distribuição das entidades do mapeamento urbano (MU) pelas coordenadas máximas ($(x_{inicial}, y_{inicial})$ e (x_{final}, y_{final})) do banco de dados (*extent*).

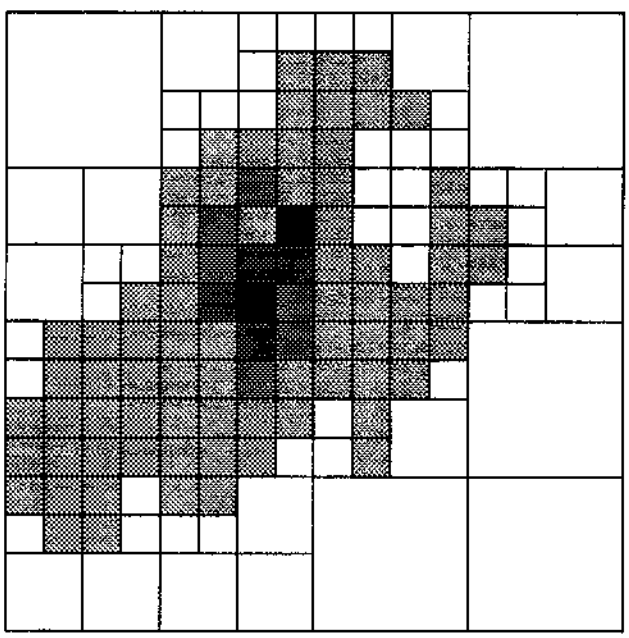

Figura 4.2: Distribuição das entidades do MU sobre o *extent*.

Neste exemplo, a distribuição utiliza a divisão do *extent* em *divisions* segundo o método *quadtree* para todos os objetos geográficos do banco de dados. Considerou-se que a iteração se ocorre até que o tamanho do lado seja maior ou igual a 1.000 metros. Ainda, na contagem dos valores de cada *division* foram contabilizadas todas as entidades que estão incluídas e que passam pela região em análise (primeiro procedimento comentado anteriormente).

Para auxiliar na visualização dos valores obtidos, tons de cinzas foram relacionados às *divisions* conforme exemplificado na figura 4.2. Os tons de cinza refletem faixas de porcentagens dos dados em cada *division* em relação aos valores mínimo e máximo encontrados na última iteração. Este critério possibilita uma visão global da distribuição/seletividade dos dados sobre o *extent*. Estes valores (tons de cinza adotados e faixas de valores) visam facilitar o entendimento, podendo ser adaptadas de acordo com a aplicação e necessidades do usuário. Nesta dissertação, os valores e faixas adotadas, estão presentes na figura 4.3. A cor branca reflete sempre a não ocorrência de valores. Os demais tons representam faixas na escala de 20%.

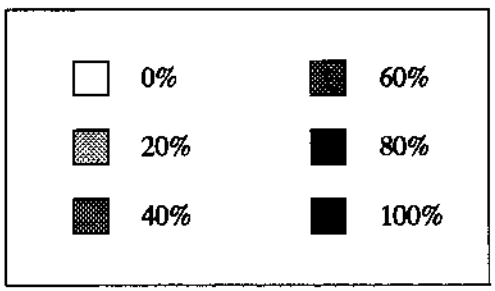

Figura 4.3: Tons de cinza e faixas de valores adotados para destacar a seletividade dos dados.

As *divisions* foram identificadas por um número (1 para o *extent* inteiro) da esquerda para a direita, de baixo para cima. Para cada *division* obtém-se a quantidade de dados presentes e sua seletividade. Assim, inicialmente, tem-se:

Division	Coordenadas		Qtde	Selet.(%)
1	286.000, 7.447.500	309.000, 7.467.000	39.883	100.00

Primeira iteração:

11	286.000, 7.447.500	297.500, 7.457.250	10.259	25.72
12	286.000, 7.457.250	297.500, 7.467.000	26.437	66.28
13	297.500, 7.457.250	309.000, 7.467.000	3.047	7.63
14	297.500, 7.447.500	309.000, 7.457.250	328	0.82

Segunda iteração. Inicia-se com a primeira *division* gerada na primeira iteração:

111	286.000, 7.447.500	291.750, 7.452.375	85	0.21
112	286.000, 7.452.375	291.750, 7.457.250	534	1.33
113	291.750, 7.452.375	297.500, 7.457.250	9.038	22.66
114	291.750, 7.447.500	297.500, 7.452.375	666	1.66

Terceira iteração. Inicia-se com a primeira *division* gerada na segunda iteração:

1111	286.000, 7.447.500	288.875, 7.449.937	0	-
1112	286.000, 7.449.937	288.875, 7.452.375	19	0.04
1113	288.875, 7.449.937	291.750, 7.452.375	68	0.17
1114	288.875, 7.447.500	291.750, 7.449.937	0	-

Quarta iteração. Inicia-se com a primeira *division* gerada na terceira iteração. Como foi obtido o valor zero na iteração anterior, esta *division* não precisa ser verificada. Neste caso, o processo segue com a segunda iteração gerada na terceira iteração e obtem-se os valores :

11121	286.000, 7.449.937	287.437, 7.451.156	0	-
11122	286.000, 7.451.156	287.437, 7.452.375	1	0.00
11123	287.437, 7.451.156	288.875, 7.452.375	11	0.02
11124	287.437, 7.449.937	288.875, 7.451.156	10	0.02

O processo segue recursivamente, de acordo com o critério de término adotado. Através destes dados pode-se analisar de maneira geral onde se encontra a maior concentração de objetos geográficos.

Observa-se que, neste exemplo, a mesma ocorrência de uma entidade pode ser contabilizada várias vezes. Considerando o segundo caso onde apenas os objetos geográficos totalmente incluídos são verificados, a mesma ocorrência seria contabilizada somente naquelas *divisions* onde ela estivesse totalmente incluída. Ainda, no segundo caso, dependendo do tamanho da *division* considerada (ou a iteração em questão), muitas ocorrências poderiam ser desprezadas, o que poderia ilustrar *divisions* com conteúdo zero não ilustrando a realidade. Tal fato pode afetar o processo de análise de desempenho consideravelmente. Neste caso, numa consulta por exemplo, os resultados de desempenho podem não refletir o ambiente de interesse, ou mesmo os resultados esperados.

De acordo com o discutido acima, o primeiro procedimento mostra-se mais adequado quando buscamos observar o número máximo de ocorrências em uma determinada *division*, ou seja, encontrar a maior concentração de um objeto geográfico. Também, permitiria respostas a perguntas como: quais as áreas de maior concentração de trechos de logradouros, rodovias, prédios etc; quais as áreas sem qualquer ocorrência desses objetos geográficos.

Já o segundo procedimento possibilita controlar a localização e a seletividade dos dados com maior precisão.

[Cif95] sugere que a determinação da quantidade de pontos, linhas e polígonos de cada *division* pode ser efetuada com base no conceito de densidade: densidade relativa não geográfica e densidade relativa geográfica. A densidade relativa não geográfica (apresentada até então nesta seção) descreve o porcentual dos dados que está contido em uma dada *division* em relação ao total de dados contidos no *extent*. Este é o ponto chave para se obter uma seletividade controlada pois a densidade pode sempre ser determinada, uma vez que se conhece a quantidade total de dados no *extent* e na *division*. A densidade relativa geográfica consiste na razão (total de dados)/(área do *extent*). A figura 4.4 ilustra tais densidades, mostrando a mesma densidade geográfica e não geográfica para *divisions* de tamanhos distintos.

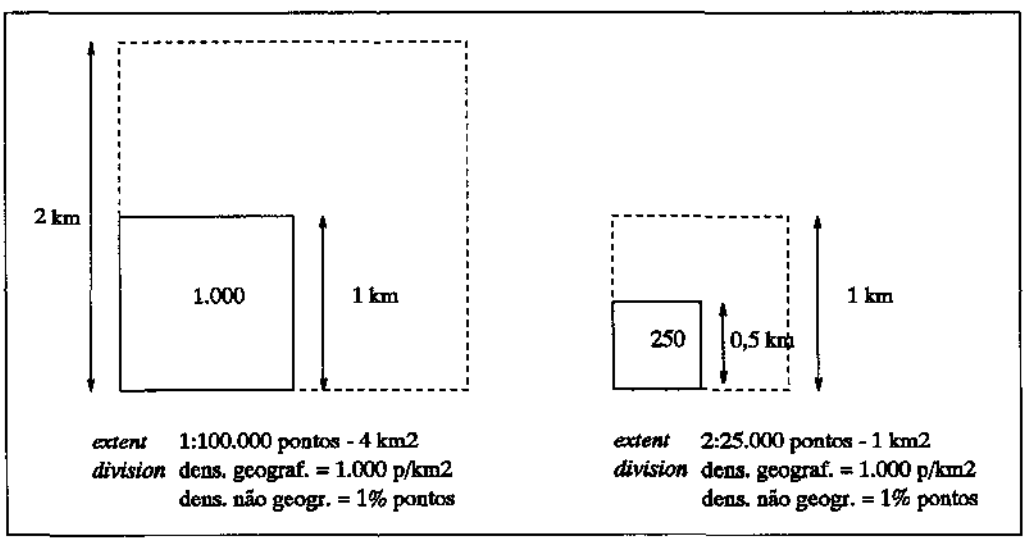

Figura 4.4: Conceitos de densidade relativa geográfica e não geográfica.

4.4 Conteúdo do banco de dados

Os dados de uma aplicação devem ser caracterizados em relação aos tipos, quantidade (de ocorrências, valores máximos e mínimos), distribuição, seletividade e propriedades do

banco de dados.

Quanto aos tipos de dados, estes poderão ser classificados em:

- **dados convencionais**: dados comumente encontrados em SGBDs convencionais. A seguir, é efetuada uma breve descrição de alguns destes tipos de dados:
  - tipo *integer*: corresponde ao armazenamento de números inteiros que podem variar entre 0 e 65.535, sem sinal. A quantidade de *bytes* necessária para o armazenamento de dados deste tipo é de 2 *bytes*.
  - tipo *long integer*: corresponde ao armazenamento de números inteiros que podem variar entre 0 e 44.294.967.295, sem sinal. A quantidade de *bytes* necessária para o armazenamento de dados deste tipo é de 4 *bytes*.
  - tipo *float*: corresponde ao armazenamento de números reais que podem variar entre $3,4^{-38}$ e $3,4^{+38}$, com seis dígitos de precisão. A quantidade de *bytes* necessária para o armazenamento de dados deste tipo é de 4 *bytes*.
  - tipos *char* e *string*: um dado do tipo *char* corresponde ao armazenamento de um caracter, ou seja, ocupa 1 *byte* em geral. O tipo *string*, uma generalização do tipo *char*, consiste no armazenamento de n caracteres, representados por *string[n]*, ocupando n *bytes*.
  - tipo *date*: corresponde ao armazenamento de uma data completa (dia, mês e ano - 06/11/1968). Supondo o uso de números para o armazenamento deste tipo, uma vez que isto facilita a manipulação numérica de datas, faz-se necessário 2 *bytes* para o seu armazenamento.
  - tipo *enum*: corresponde a uma enumeração de valores inteiro. A quantidade de *bytes* requerida para o armazenamento de um dado desse tipo varia de acordo com a quantidade de elementos da enumeração. Caso o tipo enumerado seja formado a partir do tipo *integer*, é necessário 2 *bytes* para o seu armazenamento.
- **célula**: dado armazenado no formato *raster* (quadrático/retangular), correspondente a uma área específica do espaço geográfico. Para este tipo de dado deve-se associar uma categoria e armazenar as suas coordenadas centrais x e y. A derivação de coordenadas geográficas é efetuada de acordo com a posição das células na matriz, as quais possuem implicitamente associada à sua posição uma localização geográfica.
- **ponto**: dado espacial de dimensão zero armazenado no formato vetorial. Para este tipo de dado deve-se armazenar as suas coordenadas x e y.
- **linha**: dados espaciais unidimensionais armazenados no formato vetorial. Uma conexão retilínea entre dois pontos é denominada segmento de linha. A interligação

de vários segmentos de linha forma uma linha. Deve-se armazenar para cada linha, ordenadamente, as coordenadas de todos os pontos que a compõem, sendo necessário fixar o número de pontos e comprimento.

- **polígono**: dado espacial bidimensional armazenado no formato vetorial. Para este tipo de dado deve-se armazenar ordenadamente as coordenadas de todos os pontos que o compõem. Deve-se fixar o número de pontos (vértices) e área.

Ainda, dados que representam topologia de redes e dados gráficos são largamente encontrados em SIGs. 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.

- **redes**: as informações espaciais são usualmente armazenadas em forma de grafo que armazena informações sobre recursos que fluem entre localizações geográficas distintas. Os grafos usam topologias arco-nó, onde os arcos têm um sentido de fluxo e os nós (objetos geográficos) podem ser fontes ou sorvedores ([CCH+96] e [Vas96]).

Dados gráficos, apesar de muito importante para alguns tipos de aplicações georeferenciadas, são mais utilizados para efeitos de visualização, e são tratados principalmente por funções de processamento de imagens. Sendo assim, nexto contexto, a 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 necessária discretizá-la tanto no espaço - amostragem, quanto na amplitude - quantização [CCH+96]. Assim, tem-se:

- **imagem digital**: consiste em uma matriz de números digitais chamados de *pixels*. Cada *pixel* corresponde a um retângulo na superfície da imagem original, não digital.

Por via de regra, dados gráficos são implicitamente relacionados aos dados espaciais e convencionais, através de cada objeto geográfico contido no banco de dados. Quando dados gráficos existem isoladamente, estes geralmente adicionam apenas uma informação visual a uma consulta específica, inexistindo no sentido de fazer parte de um predicado em uma consulta geográfica. Assim, uma consulta geográfica do tipo "Quais as casas próximas à Unicamp que possuem mais de quatro quartos e estão à venda a um preço entre 30.000 e 400.000 reais?", pode retornar como resposta, adicionalmente à visualização dos dados espaciais e convencionais, uma foto em perspectiva de cada casa que satisfaça a consulta, para que se possa ter uma noção da arquitetura e estilo do imóvel. Consultas em SIGs dificilmente envolveriam algo como "Quais as casas perto da Unicamp parecidas com a foto Casa-1?". Além disso, alguns SIGs permitem a omissão do armazenamento de

dados relativos a dados convencionais e não convencionais, uma vez que estes podem ser derivados a partir destes últimos [Cif95].

A descrição destes tipos de dados tem por objetivo padronizar o conteúdo dos dados. O fato destes serem caracterizados aqui desta forma em relação ao tipo, não significa sua existência em todas as aplicações reais.

Os valores de quantidade de ocorrências e valores máximos e mínimos dos dados serão obtidos diretamente sobre a aplicação em análise. Quando se utiliza dados sintéticos, estes valores devem ser pré-estabelecidos e cuidados devem ser tomados para que reflitam a aplicação sendo analisada. No caso de aplicações reais, pode-se desenvolver algoritmos específicos ou mesmo utilizar a linguagem de consulta disponível no SGBD em questão para se obter a quantificação dos dados.

A distribuição dos dados depende do tipo e domínio dos dados sendo analisados. Quanto a dados convencionais, duas formas de distribuição são verificadas: distribuição uniforme e distribuição não uniforme segundo a lei de *Zipf* (apresentadas no capítulo 3). A utilização destas distribuições busca refletir a não uniformidade dos dados quando da geração de dados sintéticos. Ao serem aplicadas sobre valores obtidos junto a aplicações reais, busca-se uma caracterização mais precisa dos mesmos. O objetivo, neste último caso, é obter uma distribuição de *Zipf* (o valor do fator de decaimento Z) o mais próxima possível dos valores reais e que possam ser utilizados na geração de dados sintéticos.

No caso, estas distribuições devem ser aplicadas para cada atributo de uma entidade. Logicamente, atributos que possuem características semelhantes precisam ser analisados apenas uma vez e seu resultado generalizado. Também, alguns atributos podem apresentar pouca relevância no contexto da aplicação, não precisando ser caracterizados quanto à distribuição. Por exemplo, atributos considerados chaves primárias tem distribuição uniforme uma vez que cada valor diferente do mesmo deve ocorrer somente uma vez. Por outro lado, a distribuição do mesmo como chave estrangeira vai depender da sua cardinalidade. Relacionamentos que diferem de (1:1) devem ser verificados.

Quanto a dados não convencionais, vários fatores devem ser levados em consideração quando se pretende trabalhar dados no formato vetorial. Além do seu domínio (valores que podem ser gerados para as coordenadas x e y), é preciso considerar aspectos relacionados à geometria do dado espacial, tais como: forma geométrica, tamanho e complexidade. Tais fatores devem ser considerados na caracterização de uma aplicação real.

De acordo com o apresentado, a caracterização da distribuição de pontos é simples, basta verificar a quantidade de valores contidos no interior de cada *division*.

Já a caracterização da distribuição de linhas será efetuada com base nos fatores: forma geométrica, tamanho e complexidade. A forma geométrica visa caracterizar a orientação das linhas (vertical, horizontal, diagonal descendente e diagonal ascendente,

segundo [Cif95]) para o cálculo do MBR [6]. O MBR é um tipo de *container* que consiste no menor retângulo que circunscreve o objeto geográfico em questão. A complexidade caracteriza o número de pontos de uma linha.

Novamente, observa-se que ao analisar a distribuição de linhas perante uma aplicação real, os fatores forma geométrica, tamanho e complexidade podem variar de acordo com a aplicação e com as características próprias da entidade. Pode-se ter entidades cuja característica marcante seja o tamanho, enquanto que em outras seja a complexidade ou mesmo a forma geométrica. Ainda, pode acontecer a combinação dessas. Por outro lado, o interesse está em caracterizar estas entidades e, sendo assim, para os fatores tamanho e complexidade, basta obter estes valores diretamente sobre o banco de dados da aplicação.

Em relação à forma geométrica de linhas, procura-se obter o MBR. Quanto maior a porcentagem de área relativa ao *dead space* dentro do MBR, maior a probabilidade de se escolher linhas que não satisfazem a resposta a uma consulta, fazendo necessário sua remoção na fase de filtragem.

Em aplicações reais deve-se caracterizar a forma geométrica de acordo com a porcentagem do MBR em relação ao *extent*. Em [Pap95], o MBR é classificado em pequeno, médio e grande. Ainda em [ND97] e [TP95], o valor do lado do MBR é utilizado para classificar o MBR. Os valores dos lados (L) iguais a $0,5\%$, $2,5\%$ e 5% em relação ao lado do *extent* corresponde à classificação pequena, média e grande, respectivamente.

Para o cálculo do MBR para cada linha procedemos da seguinte forma:

- obtém-se os menores valores das coordenadas x e y entre todas os pontos da linha. Estes valores correspondem ao primeiro ponto do MBR;
- obtém-se os maiores valores das coordenadas x e y entre todas os pontos da linha. Estes valores correspondem ao segundo ponto do MBR.

Estes pontos caracterizam o MBR, a partir do qual a área pode ser calculada.

Em seu trabalho, [Cif95] se preocupou na geração das linhas (vertical, horizontal e diagonal) dentro de cada *division* sendo, neste caso, o MBR máximo o tamanho da *division* considerada. Ainda, a porcentagem de área relativa ao *dead space* dentro do MBR era máxima quando se gerava linha com o formato diagonal. Também, todos as linhas foram geradas com tamanho entre 5 a 95 e complexidade de 2 ou 50 pontos. Com os valores obtidos com a caracterização dos dados junto a uma aplicação real como comentado anteriormente, os mecanismos de geração de dados de [Cif95] podem ser utilizados de maneira a gerarem valores de dados sintéticos próximos a aplicações reais.

A análise de polígonos deve ser efetuada com base nos fatores forma geométrica, tamanho e complexidade, semelhante a linhas.

[6] *Minimum Bounding Rectangle - Bounding Box.*

Da mesma forma que na análise de distribuição de linhas, na distribuição de polígonos perante uma aplicação real os fatores forma geométrica, tamanho e complexidade podem variar de acordo com a aplicação e com as características próprias da entidade. Contudo, objetiva-se caracterizar estas entidades e procedemos de maneira semelhante a linhas.

Vale ressaltar que em algumas aplicações SIG a área de um polígono pode estar já armazenada. Neste caso, pode-se facilmente verificar o *dead space*. Em algumas aplicações esta área deve ser calculada para se proceder adequadamente à caracterização da aplicação.

Quanto a dados *raster* [7], a distribuição pode ser baseada exclusivamente na porcentagem que cada categoria ocupa no *extent*. Deste modo, a seletividade de dados *raster* é controlada utilizando-se as porcentagens aplicadas às categorias.

Outra forma de se analisar dados *raster*, de modo que sejam gerados objetos geográficos (2D) bem definidos em todo *extent*, consiste em executar uma transação de conversão de dados - Vetor/Raster. A seletividade dos dados *raster* é derivada diretamente do tamanho dos polígonos convertidos, sendo no entanto sujeita à aproximações decorrentes do processo de conversão.

A seletividade para dados convencionais pode ser calculada de acordo com os tipos de predicados das consultas. Baseando-se em hipóteses teóricas de uniformidade de [Chr84], tem-se as fórmulas abaixo, onde c representa um atributo qualquer. Os valores max e min definem o domínio dos valores do atributo, no caso de valores numéricos.

$$c < v \qquad s = (v - min)/(max - min) \tag{4.1}$$

$$c > v \qquad s = (max - v)/(max - min) \tag{4.2}$$

$$c = v \qquad s = 1/N_{distintos} \tag{4.3}$$

$$v_1 \leq c \leq v_2 \qquad s = (v_2 - v_1)/(max - min) \tag{4.4}$$

Nas combinações de predicados, assume-se independência dos valores dos atributos. No caso de serem dados dois predicados P_1 e P_2 com seletividades S_1 e S_2, respectivamente, tem-se:

$$P_1 \ \ and \ \ P_2 = S_1 \times S_2 \tag{4.5}$$

$$P_1 \ \ or \ \ P_2 = S_1 + S_2 - S_1 \times S_2 \tag{4.6}$$

[7] Dados *raster* são predominantemnete usados para representar dados temáticos e dão ênfase no conteúdo de áreas geográficas mais que nos seus limites.

4.5 Carga de trabalho

A determinação da carga de trabalho é muito importante para o sucesso de qualquer metodologia de análise de desempenho. Um estudo mais apurado da carga de trabalho em sistemas reais pode permitir que os SGBDs sejam testados mais eficientemente e melhor sintonizados para o desempenho.

Entre os parâmetros mais relevantes para a descrição das transações que compõem a carga de trabalho incluem-se:

- **Tipo de transação**: seleção, atualização, projeção, etc ou ainda, transações mais complexas.
- **Predicados**: número, tipo e independência de predicados.
- **Valores finais e intermediários**: tamanho das tuplas e atributos como resultados das transações.
- **Freqüência relativa das transações**: para o conjunto das transações encontradas nas aplicações, organizadas por tipo de transação.

Em [Cam95] e [CCH+96] as operações realizadas sobre dados geográficos são classificadas em:

- **de construção**: criam novos objetos geográficos.
- **de atualização**: modificam objetos geográficos já existentes.
- **escalares**: retornam valores escalares em relação a propriedades e relacionamentos entre objetos geográficos.
- **booleanos**: retornam valores lógicos em relação a relacionamentos entre objetos geográficos.

[Cif95] propõe as seguintes transações para a caracterização de um *benchmark* voltada a SIG:

- reclassificação: permite que categorias de um tema sejam agrupadas, gerando novas categorias.
- superposição: permite a criação de temas formados pela superposição de outros temas. No novo tema, cada categoria é gerada pela combinação das categorias dos temas superpostos. Esta transação possui algumas variantes, depedendo do tipo de algoritmo utilizado. Pode ser assim classificada:

  - convencional: cria uma nova categoria a partir da sobreposição de duas categorias, ou seja, a nova categoria é gerada indicando a interseção das categorias dos temas originais (junção das categorias de cada um dos temas envolvidos).
  - numérica: superposição convencional acrescida de operações aritméticas e relacionais. As principais operações efetuadas neste tipo de superposição são: adição, subtração, multiplicação, divisão, exponenciação, minimização e maximização.
  - booleana: aplica operações booleanas às categorias originais. Típicas operações booleanas são: *And,Or, Xor* e *Not.*
  - análise de ponderação: variante da transação de superposição que combina aspectos das superposições convencional e numérica. As categorias do tema superposto são geradas por superposição convencional e seus pesos são obtidos a partir da aplicação de operadores numéricos (por exemplo, adição) às categorias dos temas originais. Ela pode ser classificada ainda em:
    * simples: considera as categorias dos temas originais como sendo parte de um único conjunto onde para cada elemento desse conjunto é associado um peso. Para se evitar ambigüidade na determinação das categorias originais, as categorias geradas são ponderadas com pesos distintos, normalmente potências de 2.
    * múltipla (ou tabelada): os relacionamentos entre as categorias dos temas originais são ponderados e dispostos em uma tabela (categorias do tema 1 em relação a categorias do tema 2). O peso de cada categoria resultante é obtido buscando a entrada correspondente na tabela de pesos.
  - transações baseadas em conjunto: são generalizações da superposição booleana. Neste tipo de transação, são aplicadas operações matemáticas de conjunto sobre os objetos geográficos e o resultado é também um conjunto de objetos geográficos. As principais transações deste tipo são: união, interseção e diferença.

- decomposição: aplica-se a polígonos e consiste em separar dois componentes básicos: seu interior e sua fronteira. Esta operação é usada para se determinar a pertinência ou não de um ponto ao interior de um polígono. Primeiramente, deve-se obter o interior do objeto para depois efetuar-se a transação topológica de inclusão.

- transações topológicas booleanas: verificam se um determinado relacionamento topológico existe entre dois objetos geográficos. Juntamente com as transações de busca topológicas, estas transações determinam a existência de relacionamentos entre objetos geográficos. Relacionamentos topológicos identificados:

- cruzamento: verifica se objetos geográficos [não] atravessam (cruzam) um objeto fonte;
- interseção: verifica se objetos geográficos [não] intersectam um objeto fonte;
- disjunção: verifica se objetos geográficos [não] são disjuntos a um objeto fonte;
- adjacência: verifica se objetos geográficos [não] são adjacentes em relação a um objeto fonte;
- inclusão: verifica se objetos geográficos [não] estão contidos em um objeto fonte;
- igualdade geométrica: verifica se objetos geográficos são iguais em forma (geometria).

- transações de busca topológica: tem como resultado objetos topológicos que apresentam um determinado relacionamento topológico em relação ao objeto topológico especificado. Variações segundo [Cif95]:
  - determinação do OG mais próximo de um OG fonte.
  - determinação dos OGs que [não] são adjacentes a um OG fonte.
  - determinação dos OGs que [não] estão contidos em um OG fonte.
  - determinação dos OGs que [não] contêm um OG fonte.
  - determinação dos OGs que [não] intersectam um OG fonte.
  - determinação dos OGs que [não] atravessam um OG fonte.

  De acordo com [CdFvO93], citado por [Cer96], o conjunto mínimo de relacionamentos topológicos necessários para representar a topologia entre objetos geográficos são: *cross, in , touch, disjoint* e *overlap*. Observa-se que enquanto as operações de [CdFvO93] são binárias, as definidas por [Cif95] são orientadas a conjuntos, tornando necessário definir operações distintas para verificar a ocorrência ou não de relacionamentos topológicos entre os objetos geográficos envolvidos na transação. De acordo com [CdFvO93], a determinação de seus relacionamentos é auxiliada por operações para separar a fronteira e interior das entidades geográficas.

- transações topológicas escalares: determinam as coordenadas geográficas onde ocorre um determinado relacionamento topológico entre dois objetos geográficos. Transações topológicas escalares típicas são a interseção e o cruzamento.

- transações escalares: retornam valores escalares referentes às características intrínsecas de um objeto geográfico ou de relacionamentos topológicos entre objetos geográficos. Exemplos:

  - cálculo da distância mínima entre dois objetos geográficos;
  - cálculo da área de um objeto geográfico (polígono);
  - cálculo do perímetro de um objeto geográfico (polígono);
  - cálculo do comprimento de um objeto geográfico (linha).

- análise de proximidade: efetuda a partir da criação de uma região de *buffer* em torno de uma área analisada, que pode ser interna ou externa à sua fronteira. A análise de proximidade pode ser caracterizada como sendo:

  - simples: consiste em gerar um objeto geográfico na forma de um "corredor", cujos limites externos possuem uma distância k em relação a um objeto geográfico fonte e cujos limites internos são formados pelos limites do próprio objeto geográfico fonte a partir de onde a distância k começou a ser medida.
  - múltipla: generalização da análise de proximidade simples onde consiste em gerar múltiplas zonas de *buffer* ao redor de um mesmo objeto geográfico fonte.

- transações de conversão de formatos de dados.

- transações específicas de dados no formato *raster*: ocorre devido à natureza particular de armazenamento em células providas por este formato. As principais transações são descritas a seguir:

  - agrupamento: junção de um conjunto de células de mesma categoria, adjacentes entre si, para formar objetos geográficos;
  - mudança de resolução espacial: modifica o tamanho de armazenamento das células;
  - manipulação de matrizes de células: visa verificar a diferença de conteúdo de matrizes - por exemplo, uma matriz produzida a partir de uma observação de solo e uma matriz por sensoriamento remoto.

- transações diversas: são transações comumente usadas, tais como:

  - determinação das coordenadas geográficas de um conjunto de OGs.
  - geração de um OG-1D a partir de dois OGs-0D.
  - carga do sistema, ou seja, inicialização do SIG, a qual inclui a construção de índices espaciais e convencionais.
  - seleção de OGs a partir dos valores de seus atributos convencionais, denominada de seleção convencional.

  - armazenamento de OGs em memória secundária.

- Transações baseadas em redes: atuam sobre grafos. São elas ([Vas96], [Cer96] e [CCH+96]):
  - vizinhança;
  - caminho ótimo;
  - caminho crítico;
  - segmentação dinâmica;
  - caminho mais curto;
  - fecho transitivo (fluxo máximo).

- transações baseadas em orientação: são aquelas relacionadas ao posicionamento geográfico relativo de dois ou mais objetos. Segundo [Her93], os principais relacionamentos de orientação são:
  - em frente;
  - atrás;
  - à direita;
  - à esquerda;
  - à direita-atrás;
  - à esquerda-atrás;
  - à direita-em frente;
  - à esquerda-em frente;

- transações aplicadas sobre imagens: são de natureza heterogênea e dependem muito da aplicação em questão. [CCH+96] divide as operações sobre imagem em dois grupos:
  - transformações radiométricas: os valores dos *pixels* da imagem são alterados sem modificar a geometria da imagem;
  - transformações geométricas: a geometria da imagem é alterada.

  Deste modo, [CCH+96] destaca as principais transações sobre imagem, como sendo:
  - realce;

- filtragem;
- classificação;
- segmentação.

Realce e filtragem são operações que servem tanto para identificar determinadas características quanto para pré-processar uma imagem. Classificação e segmentação permitem identificar objetos a partir de imagens.

Vale ressaltar que estas transações são predominantemente orientadas aos dados não convencionais, sendo independente do formato de dados utilizado (*raster* ou vetorial). A caracterização de transações independentemente do formato é aceitável desde que atualmente existem algoritmos nos dois formatos padrões (para a maioria das transações) e, também, existem algoritmos de conversão entre estes formatos.

Ainda, as transações de um *benchmark* deve seguir o padrão da aplicação sendo analisada, ou seja, dentro das transações primitivas relacionadas tem-se, na análise de desempenho, de adequá-las à aplicação sendo analisada. Nem todas as aplicações SIGs apresentam as mesmas características e as transações relevantes podem variar de aplicação para aplicação. Também, transações que não envolvam predicados espaciais podem ser definidas.

A carga de trabalho a ser utilizada de ser caracterizada quanto a freqüência de execução de cada transação primitiva, ordem de execução, tipo de "bufferização" (se o *buffer* deve ou não ser "limpo" após a execução de cada transação), composição dos dados (sobre quais arquivos cada transação irá atuar e critérios de seletividade).

4.6 Conclusões

Este capítulo apresentou um esquema de caracterização de sistemas de banco de dados espaciais para análise de desempenho. Paralelamente, o mesmo esquema foi elaborado de forma que possa ser aplicado a uma aplicação real e assim obter características que possibilita a geração de dados sintéticos que melhores representam aplicações reais. Também apresentou um conjunto expressivo de transações comumente encontradas em aplicações SIGs que poderão compor a carga de trabalho de um *benchmark*.

De forma geral, aplicações SIGs têm em comum a necessidade de manipularem tanto dados convencionais como dados não convencionais, fomentando a existência de SGBDs cada vez mais adequados a estas aplicações. O desempenho destes SGBDs exerce papel relevante para o sucesso comercial dessas aplicações.

Sendo assim, estas caracterizações são úteis na validação de propostas e estudos na área de análise de desempenho de sistemas de banco de dados espaciais.

Para auxiliar na obtenção do esquema de caracterização do banco de dados, utilizou-se um modelo de representação de dados estruturado em dois níveis: lógico e físico.

Em nível lógico o interesse está na especificação das estruturas/características mais abstratas que não são representadas diretamente sobre o SGBD, mas que podem ser transformadas e refinadas em informações mais concretas no nível físico.

Em relação ao nível físico, as informações obtidas no nível lógico são utilizadas para caracterizar o banco de dados. O objetivo é obter valores para as relações (seus atributos), contidas no esquema obtido anteriormente, através da caracterização mais detalhada dos dados, em termos dos domínios relacionados e de descrições suplementares dos atributos: a distribuição dos dados, a disposição sobre as relações, a ordem de colocação no agrupamento (se ele participa de algum) e se o atributo aceita valores nulos. Além das informações sobre os dados, as estruturas de índices e dos agrupamentos devem também ser informadas.

Resumidamente, a caracterização proposta segue os seguintes passos:

- determinação do esquema do banco de dados.

  Ao final desta fase deve-se identificar:

  - Entidades (relações) que compõem o banco de dados.
  - Tipos das entidades.
  - Relacionamentos entre as entidades.
  - Atributos das entidades.
  - Domínios dos atributos e domínios em comum.
  - Identificação das chaves primárias.
  - Identificação das chaves estrangeiras.
  - Identificação das restrições de mapeamento (propriedades).

- determinação do conteúdo do banco de dados quanto às propriedades:

  - tipos de dados.
  - quantidade.
  - distribuição.
  - seletividade.

- ainda, o ambiente da aplicação em análise deve ser considerado. No caso de aplicações espaciais, um esquema de análise de distribuição espacial dos dados é apresentado.

Este esquema baseia-se nos conceitos de *extent* e *division*. Um *extent* consiste na área total abrangida pela aplicação (coordenadas máximas). Uma *division* consiste em uma área contida inteiramente no *extent*. Para cada *division* é possível determinar a distribuição e a seletividade dos dados não convencionais.

Finalizando, um conjunto de transações possíveis para caracterizar a carga de trabalho são apresentadas. As transações que comporão o *benchmark* devem seguir o padrão da aplicação sendo analisada, ou seja, dentro das transações primitivas relacionadas tem-se, na análise de desempenho, que adequá-las à aplicação em estudo. Nem todas as aplicações SIGs apresentam as mesmas características e as transações relevantes podem variar de aplicação para aplicação.

Um estudo de caso é apresentado no próximo capítulo.

Capítulo 5

Estudo de caso e análise dos resultados

5.1 A aplicação

Este estudo de caso foi realizado junto ao projeto SAGRE - Sistema Automatizado de Gerenciamento de Rede Externa [1], sendo desenvolvido pelo Centro de Pesquisas e Desenvolvimento (CPqD) Telebrás, em cooperação com suas empresas operadoras [Mag93, Mag94, Agu95]. Este projeto é um típico sistema de AM/FM [2] construído sobre um SIG.

O projeto SAGRE tem por finalidade automatizar os processos de planejamento, projeto, cadastro, implantação, operação e manutenção da rede de telecomunicações, visando obter a redução do tempo de implantação das redes telefônicas, a redução do custo do terminal instalado e a melhoria da qualidade dos serviços prestados à comunidade, entre outras vantagens.

Requisitos do SAGRE exigiram uma plataforma computacional que implementa uma arquitetura cliente-servidor. Os clientes podem ser estações UNIX ou micro-computadores com sistema operacional MS/Windows. O servidor é caracterizado por estações de trabalho RISC de alto desempenho e resolução gráfica e sistema operacional UNIX. O ambiente computacional completa-se 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

[1] Por rede externa entende-se o conjunto de cabos, canalizações subterrâneas, postes, equipamentos de sustentação e de proteção a esses cabos, além de dispositivos eletrônicos complementares.

[2] *Atomated Mapping/Facilities Management.*

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. Sendo assim, o projeto SAGRE é composto de um conjunto de módulos que tratam de funções específicas de rede externa, compartilhando uma base de dados única e integrada. Atualmente, o SAGRE está organizado em seis módulos (figura 5.1):

- cadastro: tem por finalidade cadastrar as informações contidas nas plantas que compõem a rede externa (rede existente e mapeamento urbano) e que são utilizadas pelos outros módulos [GEOG94].

- conversão: tem por finalidade prover um meio de converter os dados das plantas de papel para o banco de dados SAGRE [MGS+94].

- administração: este módulo tem como função o controle e configuração da rede; permite o trabalho cooperativo entre operadores e projetistas.

- planejamento: o objetivo desde módulo é utilizar as informações contidas na Rede Externa para planejar mudanças e melhorias a serem feitas na rede externa tais como intalação de novas centrais telefônicas, quantidade de caixas terminais necessárias à instalação, localização dos equipamentos, etc.

- projeto: este módulo tem como função o auxílio no desenvolvimento de um projeto de uma rede, de forma a determinar quais os equipamentos, materiais e mão-de-obra necessários para a realização do projeto, assim como estimar o orçamento do mesmo.

- operação: alocação, reserva, liberação e dedicação de facilidades para serviços de clientes.

As informações manipuladas pelo SAGRE compreendem uma base de dados que descreve a rede externa do ponto de vista dos atributos dos seus elementos (dados alfanuméricos) e também em relação ao posicionamento de sua topologia nos mapas urbanos (dados gráficos). Para possibilitar tal tratamento, o SAGRE se utiliza de um **Sistema de Informações Geográficas - SIG**, o VISION*GIS, da *SHL Systemhouse Incorporated.* O VISION (descrito neste capítulo na próxima seção) possibilita a representação gráfica de rede externa e a associação desta com os atributos alfanuméricos que são armazenados

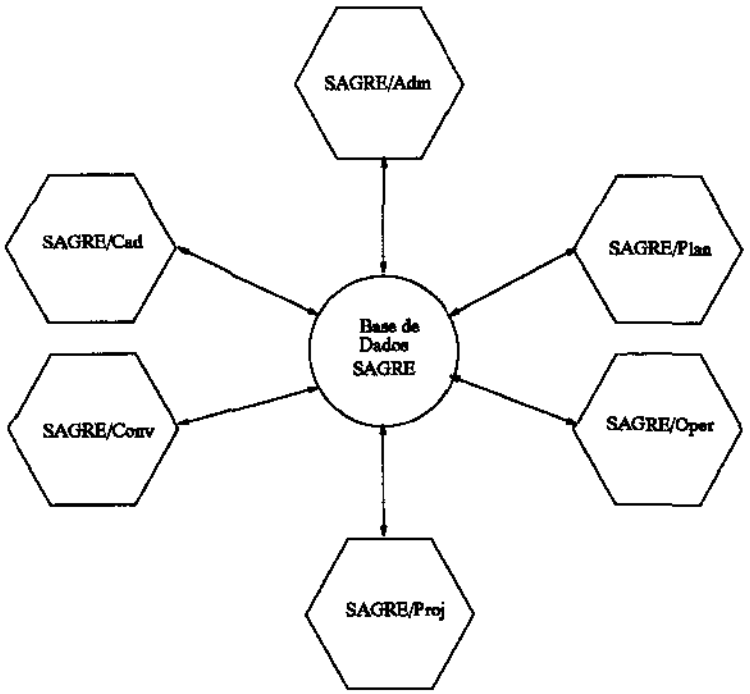

Figura 5.1: Projeto Sagre.

num banco de dados comercial. Para o desenvolvimento do projeto SAGRE adotou-se o ORACLE para gerenciar os dados referentes à rede externa.

Desta forma, o mapeamento urbano é fundamental para o sistema SAGRE sendo, para esta aplicação, denominado mapeamento urbano básico (MUB). O MUB contém mapas das áreas urbana e rural das cidades que são representadas no banco de dados, ou seja, representa o conjunto de informações gráficas e alfanuméricas referentes à base cartográfica das plantas cadastrais.

O modelo de dados do MUB contém um conjunto mínimo de informações necessárias para se relacionar o modelo de rede e suportar os aplicativos que serão desenvolvidos pelo SAGRE. Este modelo fornece uma padronização mínima adequada para que a informação existente possa ser processada pelo SAGRE sem perda ou redundância dos dados.

De acordo com o modelo gráfico do MUB, os elementos básicos de mapeamento urbano são organizados como uma série de níveis. Estes níveis gráficos são: linha central (trecho de logradouro); lotes ou divisas de lotes; numeração predial; nomenclatura dos logradouros; arruamento, face da quadra ou quadra; meio fio; edificações de destaque e nomenclatura; hidrografia e nomenclatura; obras de arte e nomenclatura; acidentes geográficos e nomenclatura; passeios, calçadas e calçadões; limites e nomenclatura; marcos geográficos e grades de coordenadas.

Apenas os níveis de linha central e de lote são imprescindíveis, e devem constar em todas as plantas convertidas, uma vez que a estes elementos gráficos estão relacionados,

respectivamente, os atributos alfanuméricos de nomenclatura de logradouros e numeração predial, os quais serão utilizados para o processamento do SAGRE [TEL93].

5.2 VISION*

VISION* é um SIG desenvolvido pela *SHL Systemhouse Inc.* utilizado para criar, manter, visualizar e analisar informações geográficas. Ele é composto por um sistema de gerenciamento de banco de dados relacional (RDBMS [3]) e outros produtos de *softwares* utilizados para construir e gerenciar informações geográficas [SHL97].

Embora haja várias definições de SIG, sabe-se que o seu coração é, como o nome implica, a informação - armazenada de maneira semelhante aos sistemas de banco de dados convencionais exceto que o dado é referenciado espacialmente, isto é, há elementos de dados que relacionam o dado a uma localização física ou relativa a outras características geográficas de alguma forma.

O banco de dados VISION* consiste de uma coleção de informações que são organizadas e referenciadas geograficamente. A base dos produtos da família VISION* é o sistema de gerenciamento de banco de dados relacional - ORACLE, que contém todos os dados relacionados às entidades geográficas.

Três tipos de informações são armazenadas: dados de coordenadas, atributos e imagem. Os dados são armazenados em uma hierarquia de estruturas que ajudam a simplificar os relacionamentos e facilitam entender as interações dos mesmos.

Muitas estruturas lógicas definidas pelo usuário e estruturas topológicas podem ser desenvolvidas e armazenadas dentro do banco de dados VISION*. Estas estruturas são: *features*, redes, camadas e grupos (Figura 5.2).

Uma entidade geográfica que tem um ou mais pontos de coordenada é chamada de *feature*. A *feature* é a unidade básica no banco de dados e pode ser composta de um número ilimitado de coordenadas *xy* (bidimensional) ou *xyz* (tridimensional). Existem quatro tipos de *features*:

- ponto: uma *feature* do tipo ponto tem no mínimo um ponto de coordenada para localizá-la na superfície da terra. Um poste de telefone ou de transformadores, por exemplo.

- linha: uma *feature* do tipo linha deve ter um ponto inicial e um ponto final; e pode também conter um número ilimitado de outros pontos para definir sua geometria. Condutores em uma rede elétrica, por exemplo.

[3] Relational DataBase Management System.

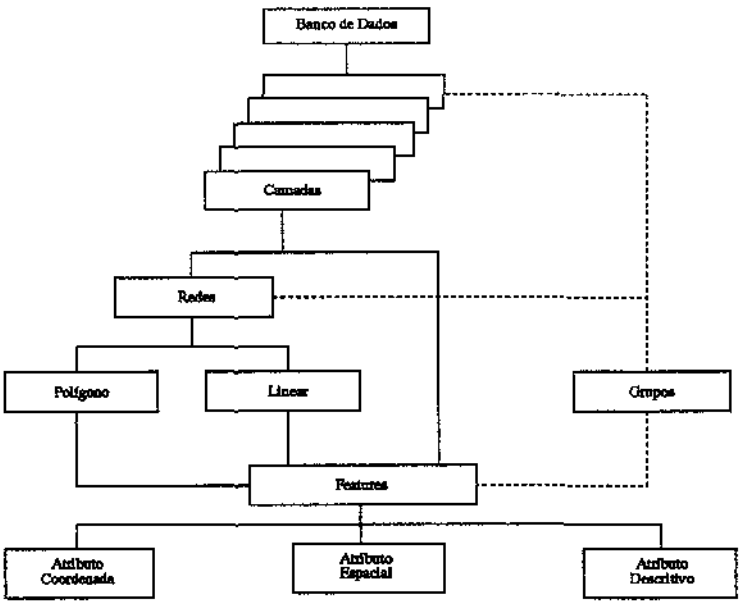

Figura 5.2: Relacionamentos das Estruturas do Banco de Dados Vision*

- polígono: *feature* envolvendo uma área, sendo identificada por um ponto chamado centróide localizado em algum lugar dentro da área que ocupa. Para definir a área, linhas de fronteiras (arestas) são conectadas extremo a extremo para formar um polígono.

- nó: *feature* que existe somente como parte de uma rede. Existem duas categorias: nós do sistema e nós do usuário. Nós do sistema são gerados pelo sistema e existem implicitamente para fins de linhas e rede. Nós do usuário são *features* definidas pelo usuário e são um tipo especial de *feature* do tipo ponto.

Uma rede é uma estrutura que contém *features* topologicamente relacionadas. Os relacionamentos topológicos são armazenados e mantidos dentro do banco de dados e são usados pelo sistema em consultas avançadas e funções de análise. Muitas redes podem existir num mesmo banco de dados, cada qual identificada por um número e um nome. Existem dois tipos de redes: poligonal e linear.

Uma rede de polígono contém um número ilimitado de polígonos conectados. Cada linha em uma rede de polígonos tem um nó do sistema para cada ponto terminal.

Uma rede linear contém *features* de linha que são conectadas em seus extremos ou são separadas geograficamente. Para cada interseção de *feature* de linha, uma *feature* nó (do sistema ou definida pelo usuário) existe. Nós do usuário podem existir independente da existência de uma linha conectando-os. Há dois tipos de rede linear: rede definida pelo sistema e rede definida pelo usuário. A rede definida pelo sistema usa uma tabela de conectividade *default* gerada pelo sistema. Nas redes definidas pelo usuário é ele quem

define a tabela de conectividade. Isto permite personalizar a conectividade das redes, adaptando as reais necessidades da aplicação.

Camadas são estruturas do banco de dados que tematicamente agrupam *features* e redes que estão relacionados logicamente (rios, lagos e oceanos podem ser agrupados em uma camada de hidrologia).

A estrutura grupo permite a criação de uma estrutura temporária para realizar manipulações. Grupos são freqüentemente utilizados para executar alguma manipulação sobre um número de *features* que de outro modo não seriam associados.

A riqueza dos tipos de dados em VISION* e a possibilidade de construir novos tipos mais complexos permite uma modelagem eficiente do fenônemo a ser analisado. Tipos como camadas e redes ampliam o leque e permitem que o usuário possa utilizá-los diretamente.

Aplicações típicas de VISION* incluem gerenciamento de redes de distribuição de gás e energia elétrica, gerenciamento de planta externa de telefonia, gerenciamento de redes de saneamento básico, administração de cidades e países, gerenciamento de recursos naturais e exploração de mapas, entre outras.

5.3 Caracterização do banco de dados

Para o estudo de caso, tomou-se como base o banco de dados da cidade de Valinhos - SP, cidade piloto do projeto SAGRE.

Como mencionado anteriormente, o projeto SAGRE trata-se de uma aplicação real, exigindo cuidados especiais quanto à divulgação de suas informações. Com isto, inicialmente, todo o banco de dados foi analisado. Todavia, o estudo aqui apresentado retrata somente aquelas informações relevantes ao trabalho proposto sem prejudicar o nível de sigilo exigido por aplicações reais.

Neste contexto, além de serem apresentadas entidades do mapeamento urbano - MU, também são apresentadas entidades de rede de canalização (LANCE_DUTO) e rede aérea (POSTE).

No caso, as coordenadas máximas do banco de dados (*extent*), ($x_{inicial}$,$y_{inicial}$) e (x_{final}, y_{final}), correspondem aos valores (286.000, 7.447.500) e (309.000, 7.467.000), respectivamente.

Ainda, como comentado no capítulo 4 desta dissertação, todos os valores podem ser obtidos sobre o próprio banco de dados da aplicação. Vale ressaltar que a aplicação tem que ser "congelada", ou seja, ao final do processo de análise deve-se ter os mesmos dados observados antes de iniciar o processo. Qualquer alteração deve ser cuidadosamente verificada.

5.3.1 Esquema do Banco de Dados

A tabela 5.1 mostra as entidades contidas no banco de dados, a serem analisadas.

Entidade	Tipo
ENDEREÇO	convencional
INDICAÇÃO_LOTE	ponto
LANCE_DUTO	linha
LOGRADOURO	convencional
LOGRADOURO_ALTER	convencional
POSTE	ponto
QUADRA	polígono
REGIÃO	polígono
TRECHO_LOGRADOURO	linha

Tabela 5.1: Entidades do banco de dados a serem caracterizadas.

O modelo físico da aplicação em relação às entidades em estudo é apresentado na figura 5.3.

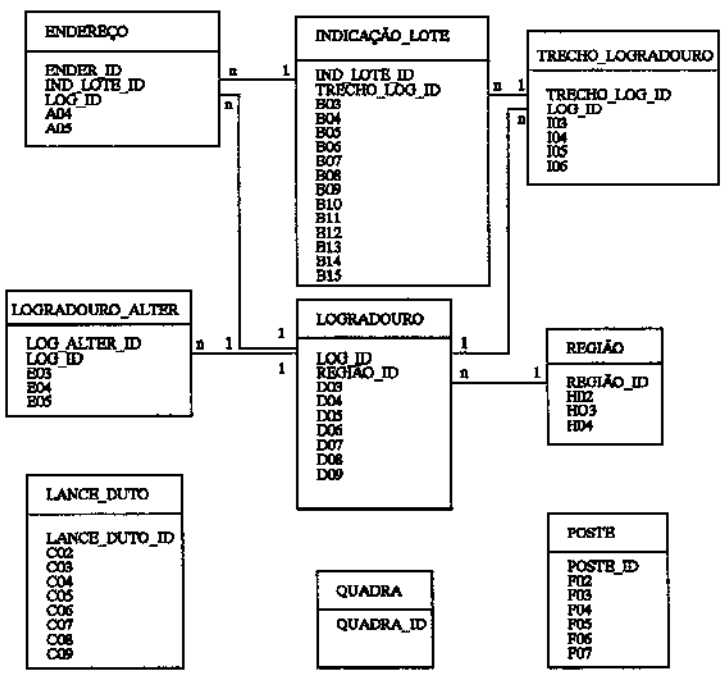

Figura 5.3: Modelo físico de dados.

Um resumo das entidades a serem analisadas é apresentado na tabela 5.2. A figura 5.4 enfatiza as ocorrências dessas entidades no banco de dados. No caso, percebe-se a maior

ocorrência das entidades ENDEREÇO, INDICAÇÃO_LOTE e POSTE, constituindo cerca de 85% dos valores a serem analisados. A entidade QUADRA apresenta apenas um atributo convencional por ser considerada uma entidade gráfica e o valor de apenas uma ocorrência da entidade REGIÃO denota a análise de apenas uma região (Valinhos - SP) como comentado anteriormente.

Entidade	Tipo	Núm. Atrib.	Qtde	Ocorrências
ENDEREÇO	convencional	5	14.228	28,79%
INDICAÇÃO_LOTE	ponto	15	14.083	28,50%
LANCE_DUTO	linha	9	376	0,76%
LOGRADOURO	convencional	9	1.210	2,45%
LOGRADOURO_ALTER	convencional	5	107	0,22%
POSTE	ponto	4	13.814	27,95%
QUADRA	polígono	1	2.310	4,67%
REGIÃO	polígono	4	1	0,01%
TRECHO_LOGRADOURO	linha	6	3.293	6,66%
Total		61	49.422	100%

Tabela 5.2: Resumo geral das entidades do banco de dados.

Em seguida, as entidades da aplicação serão caracterizadas detalhadamente (tabelas 5.3 a 5.11). O caracter "*" (asterisco) ou "+" (adição), em frente ao atributo, serve para ilustrar chave primária e chave estrangeira, respectivamente. Os valores em branco nas tabelas indicam a não existência do mesmo sobre o banco de dados. Ainda, para se manter o sigilo dos dados, os nomes dos atributos foram "mascarados" conforme apresentados nas tabelas.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	ENDER_ID *	num.	10	0	N	S	S	109.797	124.024
2	IND_LOTE_ID +	num.	10	0	S	S	N	59.784	73.886
3	LOG_ID +	num.	10	0	N	S	N	73.869	75.076
4	A04	car.	30	-	N	S	N	1	6
5	A05	car.	1	-	S	S	N		

Tabela 5.3: Caracterização da entidade ENDEREÇO.

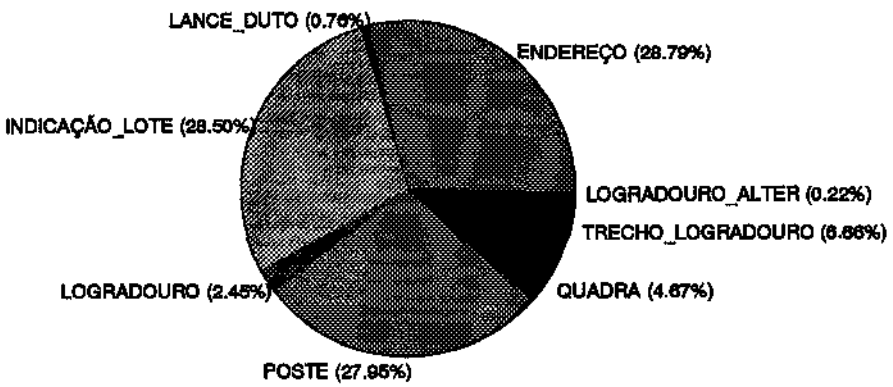

Figura 5.4: Ocorrências de cada entidade em estudo no banco de dados.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	IND_LOT_ID *	num.	10	0	N	S	S	59.784	73.866
2	TR_LOG_ID +	num.	10	0	S	S	N	56.491	59.782
3	B03	num.	1	0	N	N	N	0	0
4	B04	car.	10	-	S	N	N	1	5
5	B05	num.	2	0	S	N	N	0	5
6	B06	car.	1	0	S	N	N		
7	B07	num.	1	0	N	N	N	0	0
8	B08	num.	5	2	S	N	N		
9	B09	num.	5	2	S	N	N		
10	B10	num.	5	2	S	N	N		
11	B11	num.	5	2	S	N	N		
12	B12	num.	5	2	S	N	N		
13	B13	num.	5	2	S	N	N		
14	B14	num.	5	2	S	N	N		
15	B15	num.	5	2	S	N	N		

Tabela 5.4: Caracterização da entidade INDICAÇÃO_LOTE.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	LAN_DUTO_ID *	num.	10	0	N	S	S	4.693	9.302
2	C02	num.	6	2	S	N	N	0	251,9
3	C03	num.	6	2	S	N	N	0,1	254,2
4	C04	num.	4	0	S	N	N	0	1996
5	C05	num.	1	0	N	N	N	5	6
6	C06	num.	10	0	S	N	N	951	2.974
7	C07	num.	4	2	S	N	N	0	0
8	C08	num.	10	0	S	N	N		
9	C09	num.	10	0	S	N	N		

Tabela 5.5: Caracterização da entidade LANCE_DUTO.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	LOG_ID *	num.	10	0	N	S	S	73.868	75.077
2	REGIÃO_ID +	num.	10	0	N	S	N	73867	73868
3	D03	num.	10	0	S	S	N		
4	D04	num.	10	0	N	N	N	-1000	99.999
5	D05	car.	30	-	N	S	N	1	26
6	D06	car.	6	-	N	N	N	2	5
7	D07	car.	14	-	S	N	N	2	6
8	D08	num.	1	0	N	N	N	0	0
9	D09	num.	1	0	N	N	N	1	1

Tabela 5.6: Caracterização da entidade LOGRADOURO.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	LOG_ALT_ID *	num.	10	0	N	S	S	75.078	75.184
2	LOG_ID +	num.	10	0	N	S	N	73.889	75.076
3	E03	car.	30	-	N	S	N	1	21
4	E04	car.	6	-	S	N	N	2	5
5	E05	car.	14	-	S	N	N		

Tabela 5.7: Caracterização da entidade LOGRADOURO_ALTER.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	POSTE_ID *	num.	10	0	N	S	S	10.369	24.182
2	F02	car.	10	0	S	N	N	1	1
3	F03	num.	1	0	S	N	N	0	2
4	F04	num.	1	0	S	N	N	0	1
5	F05	num.	1	0	N	N	N	5	5
6	F06	num.	10	0	S	N	N	2.581	2.989
7	F07	num.	10	0	S	N	N		

Tabela 5.8: Caracterização da entidade POSTE.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	QUADRA_ID *	num.	10	0	N	S	S	75.821	98.658

Tabela 5.9: Caracterização da entidade QUADRA.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	REGIÃO_ID *	num.	10	0	N	S	S	73.867	73.867
2	H02	num.	5	0	S	N	N	11.686	11.686
3	H03	car.	5	-	S	N	N	3	3
4	H04	car.	30	-	N	N	N	8	8

Tabela 5.10: Caracterização da entidade REGIÃO.

Seq	Nome Atributo	Tipo	Tam	Dec	Nls	Idx	Unq	V/T Mín	V/T Máx
1	TR_LOG_ID *	num.	10	0	N	S	S	56.491	59.783
2	LOG_ID +	num.	10	0	N	S	N	73.869	75.077
3	I03	num.	1	0	N	N	N	1	1
4	I04	num.	1	0	N	N	N	0	0
5	I05	car.	10	-	S	N	N		
6	I06	car.	10	-	S	N	N		

Tabela 5.11: Caracterização da entidade TRECHO_LOGRADOURO.

5.4 Caracterização das entidades

Na caracterização de cada tipo de entidade, o importante é a descrição de cada entidade, ou seja, a caracterização precisa dos tipos de dados e quantidade de ocorrências das mesmas. A quantidade de *bytes* pode ser calculado para cada entidade, servindo apenas como uma estimativa do espaço de armazenamento requerido.

Em seguida, caracteriza-se todas as entidades. Porém, inicialmente, é ilustrado a distribuição/seletividade das entidades do tipo ponto, linha e polígono em cada *division* de acordo com as coordenadas máximas do banco de dados (*extent*).

Neste caso, para mostrar esta seletividade, o *extent* foi dividido em *division* segundo o método *quadtree* e apenas as entidades totalmente incluídas na *division* são consideradas (segundo procedimento comentado no capítulo 4). Além disso, no processo de divisão do *extent*, a iteração ocorre até que se tenha o lado da *division* menor ou igual a 250 metros. Um fragmento desses dados é mostrado abaixo.

Division	Coordenadas		Qtde	Selet.(%)
1	286.000, 7.447.500	309.000, 7.467.000	33.747	100.000
11	286.000, 7.447.500	297.500, 7.457.250	10.298	30.515
...				
112	286.000, 7.452.375	291.750, 7.457.250	1.252	3.709
...				
1123	288.875, 7.454.812	291.750, 7.457.250	667	1.97
...				
11234	290.312, 7.454.812	291.750, 7.456.031	247	0.731
112341	290.312, 7.454.812	291.031, 7.455.421	47	0.139
1123411	290.312, 7.454.812	290.671, 7.455.116	1	0.002
1123412	290.312, 7.455.116	290.671, 7.455.421	16	0.047
1123413	290.671, 7.455.116	291.031, 7.455.421	9	0.026
1123414	290.671, 7.454.812	291.031, 7.455.116	20	0.059

Através da figura 5.5 e dos valores obtidos pode-se perceber onde os dados estão concentrados e quais *division* devem ser analisadas com maior precisão. A mesma análise aplicada aqui deve ser executada para cada entidade não convencional separadamente para se obter sua caracterização.

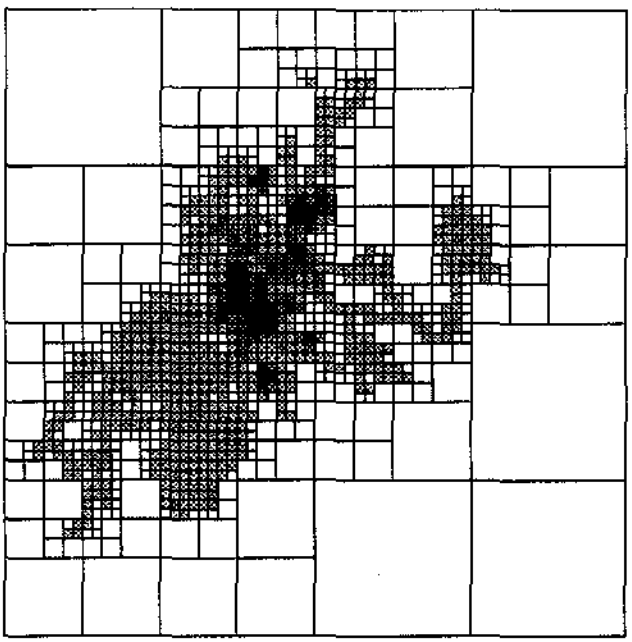

Figura 5.5: Distribuição das entidades do tipo ponto, linha e polígono sobre o *extent*.

5.4.1 Caracterização das entidades do tipo convencional

Na caracterização das entidades convencionais, leva-se em consideração: tipos de dados, ordem de colocação do atributo na entidade, se o atributo pode ou não assumir valores nulos, chave primária, chave estrangeira e a quantidade (de ocorrências, valores máximos e mínimos) dos atributos.

Sendo assim, a tabela 5.3 caracteriza a entidade ENDEREÇO em relação aos fatores acima. O total de ocorrências no banco de dados é 14.228.

No caso, apenas a título de exemplo, tem-se que estes dados ocupam 32 *bytes* a cada ocorrência, sendo 4 *bytes* para a chave (ENDER_ID), 8 *bytes* para as chaves estrangeiras (IND_LOTE_ID e LOG_ID), 10 *bytes* para A04 e 10 *bytes* para A05. O total de dados a serem armazenados é de 455.296 *bytes*.

A entidade LOGRADOURO é caracterizada na tabela 5.6. Cada ocorrência da entidade ocupa 68 *bytes*. Têm-se 1.210 ocorrências, totalizando 82.280 *bytes* a serem armazenados.

Já a entidade LOGRADOURO_ALTER é caracterizada na tabela 5.7. Neste caso, cada ocorrência da entidade ocupa 58 *bytes*. Tem-se 107 ocorrências, totalizando 7.308 *bytes* a serem armazenados.

Quanto à distribuição, atributos que apresentam as mesmas características tanto em entidades convencionais como em não convencionais, serão discutidos aqui. Salienta-se que o interesse está em obter uma distribuição tão próxima quanto possível de valores

encontrados em aplicações reais. No caso, de acordo com a fórmula de distribuição de *Zipf*, procura-se obter o valor do fator de decaimento (Z) que torna os valores gerados sintéticamente o mais próximos possíveis de valores reais.

Atributos que são chaves primárias possuem distribuição uniforme (Z=0) dentro de determinadas faixas de valores e não serão analisados. Por outro lado, estes atributos devem ser analisados quanto a suas ocorrências nas entidades em que aparecem como chaves estrangeiras.

Desta forma, o atributo LOG_ID, chave primária da entidade LOGRADOURO,possui 14.228 ocorrências sobre a entidade ENDEREÇO. A tabela 5.12 e a figura 5.6 mostram a distribuição deste atributo perante a aplicação em relação à entidade ENDEREÇO. Para possibilitar a análise deste dado, devido ao seu volume, adota-se que o *rank* é dado por faixas de valores de ocorrências. Inicialmente, verifica-se a quantidade de atributos diferentes que possuem entre 100 e 90 ocorrências, depois entre 90 e 80 e assim por diante. De acordo com a figura 5.6, o fator de decaimento $Z = 0,5$ é o que mais se aproxima dos valores reais.

A distribuição do atributo D06 presente na entidade LOGRADOURO é mostrada na tabela 5.13 - figura 5.7. O *rank*, neste caso, é dado pela quantidade de ocorrências por atributo diferentes (valor do elemento), sem acumulação de valores por faixa. Por sua vez, o melhor fator de decaimento consiste em $Z = 2,5$.

Já o atributo F06 da entidade Poste é apresentado na tabela 5.14 - figura 5.8. Neste caso, o valor de decaimento $Z = 3,0$ pode ser utilizado.

Ainda, quanto à distribuição, o atributo TR_LOG_ID, chave primária da entidade TRECHO_LOGRADOURO, possui 14.083 ocorrências sobre a entidade INDICAÇÃO-_LOTE. Já a entidade TRECHO_LOGRADOURO possui 3293 ocorrências no banco de dados. Sua distribuição é mostrada na tabela 5.15 e figura 5.9. Neste caso, o *rank* é realizado de acordo com o número de ocorrências de cada atributo diferente no banco de dados. Percebe-se pela figura 5.9 que o fator de decaimento $Z = 0,7$ é o mais indicado.

Nota-se, que nem sempre é possível obter valores ideais segundo a distribuição de *Zipf*. Tal fato deve ser observado quando da caracterização dos valores e a melhor aproximação adotada. Na geração de dados sintéticos, estes valores devem ser identificados para fundamentar os resultados obtidos.

Durante as análises, percebeu-se a tendência de chaves estrangeiras possuírem fator de decaimento entre os valores 0,4 e 0,9. Nos exemplos apresentados, figura 5.6 e 5.9, os melhores valores são Z=0,5 e Z=0,7, respectivamente. Para atributos que possuem apenas dois ou três elementos diferentes em uma quantidade muito grande de ocorrências (tipo de logradouro, tipo de poste), observou-se que o fator de decaimento está entre 2,5 e 3. Para atributos com variações de valores diferentes em sua maioria (como nome de cidades, nome de ruas), indica-se um fator de decaimento de Z=0 na geração de dados

Núm. Elem.	Núm. Ocor.	% um Elem.
3	100	0,70
9	90	0,63
10	80	0,56
7	70	0,49
21	60	0,42
27	50	0,35
43	40	0,28
91	30	0,21
181	20	0,14
350	10	0,07
460	0	-

Tabela 5.12: Distribuição atributo LOG_ID - entidade LOGRADOURO em relação à entidade ENDEREÇO.

sintéticos.

A seletividade de dados convencionais no processo de consulta depende do predicado presente na consulta a ser realizada. Por exemplo, no caso do atributo numérico LOG_ALT_ID - entidade LOGRADOURO_ALTER, se ocorrer: $LOG\_ALT\_ID > 75.100$; tem-se $s = (max - v)/(max - min) = 84/107 = 0,78$. Já o atributo caracter D06 - entidade LOGRADOURO, acontecendo: $D06 = Pca$, tem-se $s = (1/N_{distintos}) = 1/12 = 0,83$. Neste último caso deve-se levar em consideração o fator de repetição de cada valor distinto (tabela 5.13).

5.4.2 Caracterização das entidades do tipo ponto

Na caracterização de entidades do tipo ponto, deve-se levar em consideração as coordenadas x e y que as constituem. Adicionalmente, cada entidade também deve ser caracterizada em relação aos seus dados convencionais.

A tabela 5.8 caracteriza a entidade POSTE em relação aos dados convencionais. Estes dados ocupam 24 *bytes*. Já em relação aos dados não convencionais, cada ocorrência da entidade ocupa 8 *bytes*, 4 *bytes* para cada uma das coordenadas x e y. Assim cada ocorrência da entidade POSTE, ocupa 32 *bytes*. Como há 13.814 ocorrências no banco de dados, no total há 442.048 *bytes* para todo o banco de dados.

A distribuição/seletividade dos dados não convencionais da entidade POSTE pode ser visualizado sobre a figura 5.10. Para mostrar a seletividade, o *extent* foi dividido em *division* segundo o método *quadtree* e a iteração ocorre até que se tenha o lado da

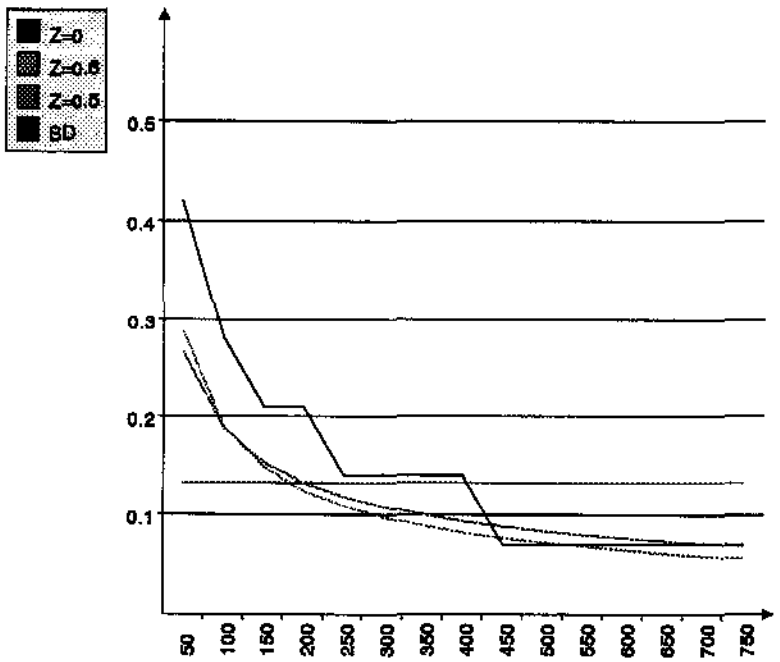

Figura 5.6: Distribuição atributo LOG_ID - entidade LOGRADOURO em relação à entidade ENDEREÇO.

Vlr. Elem.	Núm. Ocor.	% Elem.
Rua	913	75,45
Estr	107	8,84
Al	86	7,10
Av	52	4,29
Pça	20	1,65
Cam	15	1,23
Tv	9	0,74
Vel	3	0,24
Via	2	0,16
Lg	1	0,08
Pq	1	0,08
s/n	1	0,08
total	1210	99,94

Tabela 5.13: Distribuição atributo D06 - entidade LOGRADOURO.

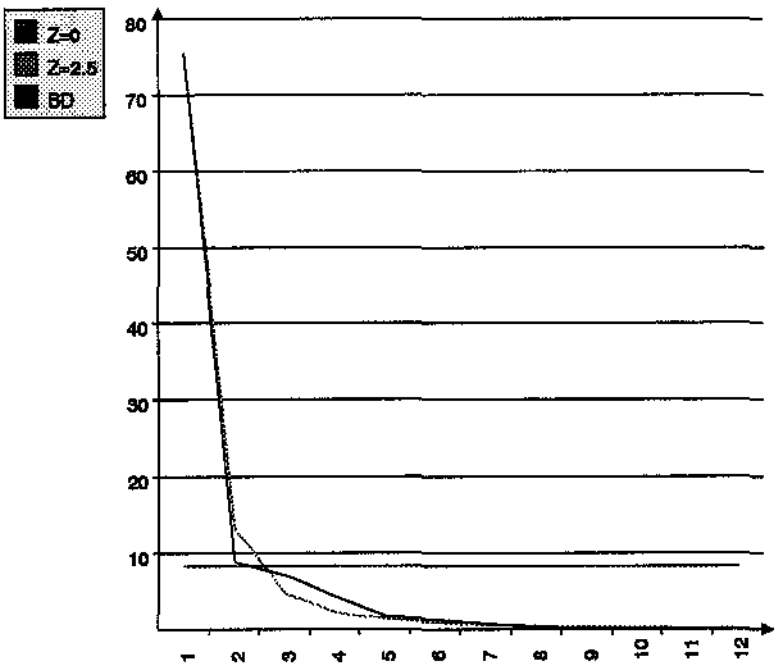

Figura 5.7: Distribuição atributo D06 - entidade LOGRADOURO.

Vlr. Elem.	Núm. Ocor.	% Elem.
2.951	7.414	53,67
2.952	6.346	45,93
2.953	25	0,180
2.983	12	0,086
2.982	5	0,036
2.581	4	0,028
2.977	3	0,026
2.989	2	0,018
-	2	0,018
2.641	1	0,007
total	13.814	99,99

Tabela 5.14: Distribuição atributo F06 - entidade POSTE.

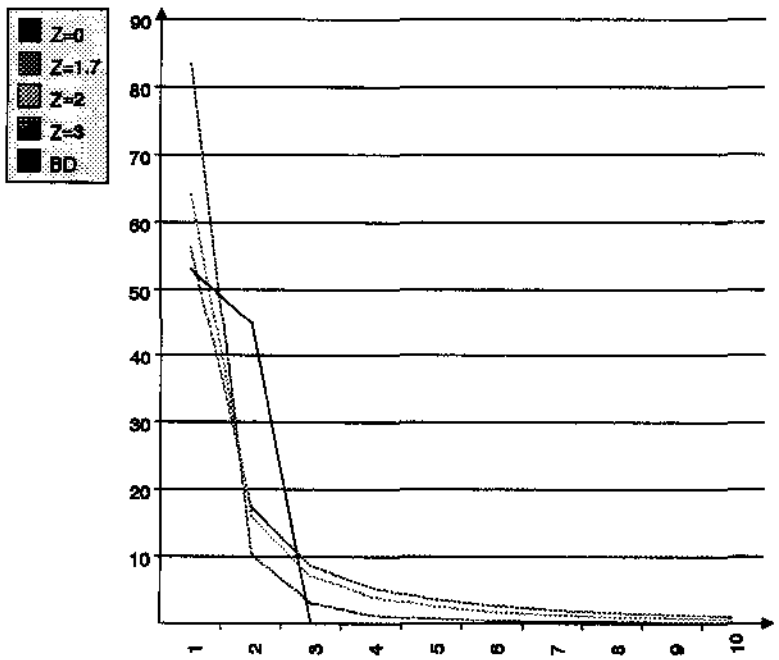

Figura 5.8: Distribuição atributo F06 - entidade POSTE.

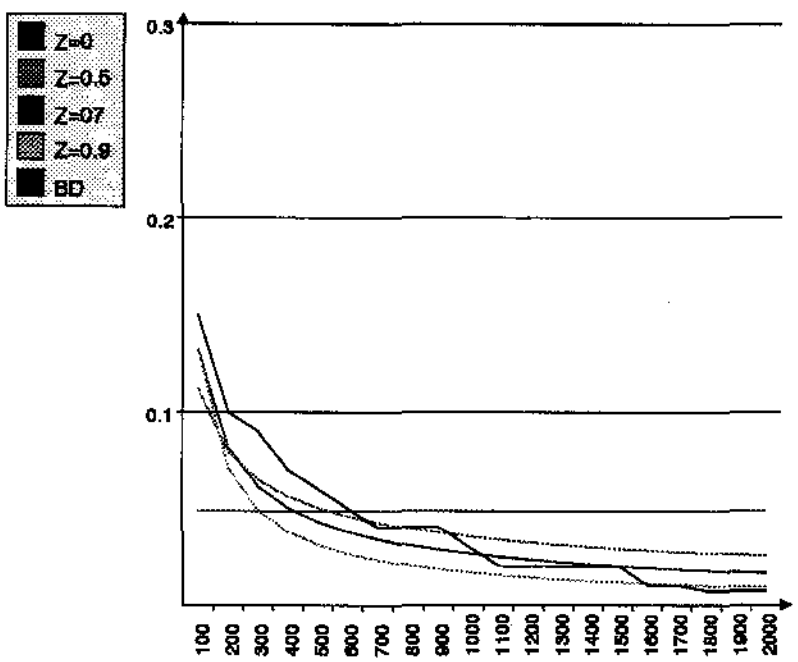

Figura 5.9: Distribuição atributo TR_LOG_ID - entidade TRECHO_LOGRADOURO em relação à entidade INDICAÇÃO_LOTE.

Núm. Elem.	Núm. Ocor.	% um Elem.
105	Outros	-
13	20	0,14
15	19	0,13
18	18	0,12
20	17	0,12
25	16	0,11
27	15	0,10
41	14	0,09
38	13	0,09
51	12	0,08
50	11	0,07
74	10	0,07
86	9	0,06
90	8	0,05
115	7	0,04
154	6	0,04
154	5	0,03
178	4	0,02
247	3	0,02
255	2	0,01
302	1	0,007
1.235	0	-

Tabela 5.15: Distribuição atributo TR_LOG_ID - entidade TRECHO_LOGRADOURO em relação à entidade INDICAÇÃO_LOTE.

division maior ou igual a 250 metros, ou seja, que a área seja superior a $62.500m^2$. Um fragmento desta caracterização dos dados para cada *division* é mostrado abaixo. No caso, pegando como exemplo a *division* 1123, temos: 545 ocorrências, coordenadas (288.875, 7.454.812)(291.750, 7.457.250), seletividade 3,945%, ocupando 17.440 *bytes*.

Division	Coordenadas		Qtde	Selet.(%)
1	286.000, 7.447.500	309.000, 7.467.000	13.814	100.000
11	286.000, 7.447.500	297.500, 7.457.250	5.432	39.322
...				
112	286.000, 7.452.375	291.750, 7.457.250	1.060	7.673
...				
1123	288.875, 7.454.812	291.750, 7.457.250	545	3.945
...				

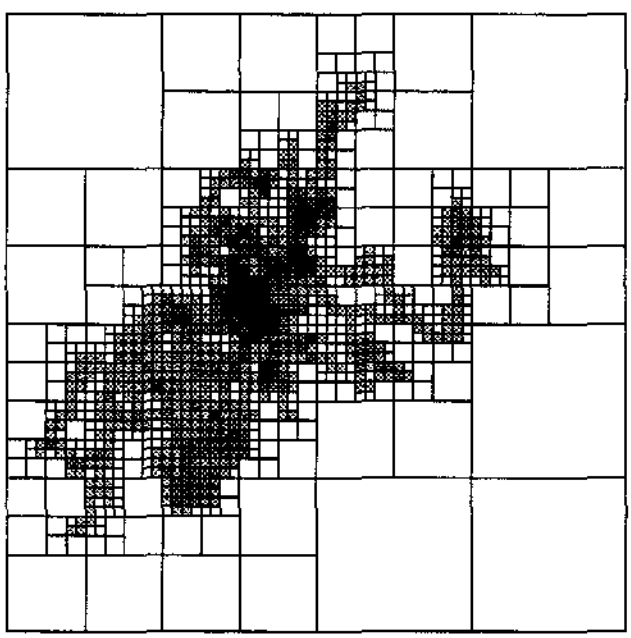

Figura 5.10: Distribuição da entidade POSTE sobre o *extent*.

A caracterização da entidade INDICAÇÃO_LOTE em relação aos dados convencionais é mostrada na tabela 5.4. No caso, estes dados ocupam 41 *bytes*. Já os dados não convencionais de cada entidade ocupam 8 *bytes*, 4 *bytes* para cada uma das coordenadas

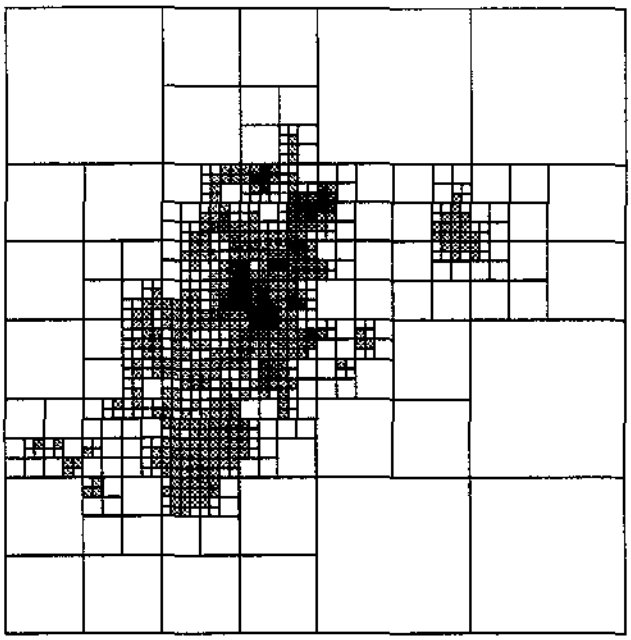

Figura 5.11: Distribuição da entidade INDICAÇÃO_LOTE sobre o *extent*.

x e y. Assim cada ocorrência da entidade INDICAÇÃO_LOTE, ocupa 49 *bytes*. Como tem-se 14.083 ocorrências no banco de dados, tem-se 690.067 *bytes* para todo o banco.

A distribuição/seletividade dos dados não convencionais da entidade INDICAÇÃO_LOTE é apresentada sobre a figura 5.11. Esta caracterização baseou-se nas mesmas condições assumidas quanto à entidade POSTE.

Os valores encontrados para cada *division* destas entidades devem ser utilizados para a geração de dados sintéticos.

5.4.3 Caracterização das entidades do tipo linha

Na caracterização de entidades do tipo linha, leva-se em consideração os fatores complexidade, tamanho e forma geométrica, conforme discutido no capítulo 4. Também, cada entidade deve ser caracterizada em relação aos dados convencionais.

Sendo assim, inicialmente, caracteriza-se a entidade TRECHO_LOGRADOURO em relação aos dados convencionais. Esta caracterização encontra-se na tabela 5.11. Para esta entidade, os dados convencionais ocupam 32 *bytes*.

Já em relação aos dados não convencionais, deve-se levar em consideração a não uniformidade dos valores obtidos sobre o banco de dados, bem como a distribuição dos mesmos.

A figura 5.12, ilustra a distribuição/seletividade dos dados da entidade TRECHO_LOGRADOURO em cada *division* do *extent*. Para mostrar esta seletividade, o *extent* foi dividido em *division* segundo o método *quadtree* e apenas as entidades totalmente incluídas

na *division* são consideradas (segundo procedimento comentado no capítulo 4). Além disso, no processo de divisão do *extent*, a iteração ocorre até que se tenha o lado da *division* maior ou igual a 250 metros, ou seja, que a área seja superior a $62.500m^2$.

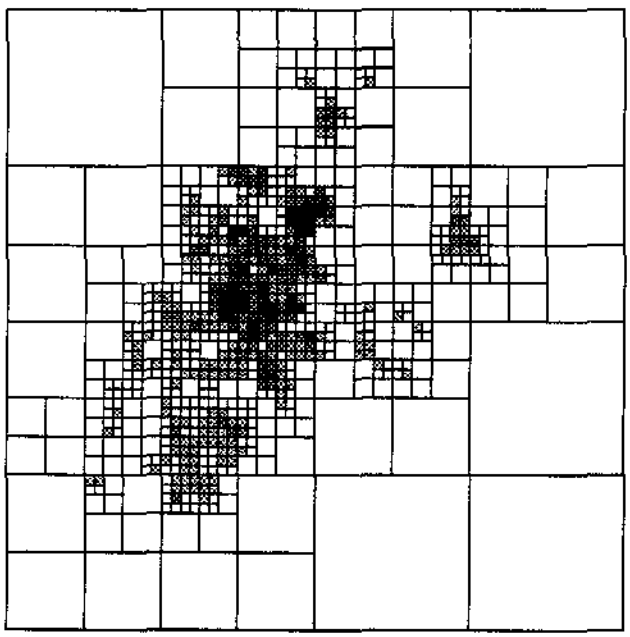

Figura 5.12: Distribuição da entidade TRECHO_LOGRADOURO sobre o *extent*.

Em relação à complexidade, ou seja, número de pontos possíveis em uma linha, a tabela 5.16 sumariza estes dados para a entidade TRECHO_LOGRADOURO em relação ao banco de dados completo. A figura 5.13 ilustra a quantidade de ocorrências de pontos por faixa de valores, a média de tamanho dentro de cada faixa e a média em relação ao banco de dados todo (visualizado como uma linha).

A mesma análise pode ser aplicada na caracterização de cada *division* do *extent*. Sendo assim, a título de exemplo, no caso da *division* 11 abaixo, obtém-se os valores da tabela 5.17:

Division	Coordenadas		Qtde	Selet.(%)
11	286.000, 7.447.500	297.500, 7.457.250	962	29,21

Em seguida, a entidade TRECHO_LOGRADOURO é caracterizada em relação ao tamanho. A tabela 5.18 (figura 5.14) sumariza as ocorrências da entidade no banco de dados.

Complexidade	Qtde	%	Média Ptos Fx	Média Tam. Fx
02	2.269	68,90%	2,00	95,93
04	604	18,34%	3,30	155,03
08	280	8,50%	6,03	255,43
16	92	2,79%	11,27	415,70
32	36	1,09%	23,02	828,09
64	11	0,33%	42,27	1.173,03
Outros	1	0,03%	66,00	2.739,54
Total	3.293	99,98%	-	-

Tabela 5.16: Caracterização da entidade TRECHO_LOGRADOURO em relação à complexidade.

Complexidade	Qtde	%
02	584	50,70%
04	194	20,16%
08	116	12,05%
16	47	4,88%
32	19	1,97%
64	1	0,10%
Outros	1	0,10%
Total	962	99,96 %

Tabela 5.17: Caracterização da entidade TRECHO_LOGRADOURO em relação à complexidade - *division* 11.

Ao caracterizar a entidade TRECHO_LOGRADOURO em relação à forma geométrica, busca-se a área ocupada devido a forma geométrica da entidade no processo de visualização ao usuário - MBR. A tabela 5.19 (figura 5.15) sumariza os dados presentes no banco de dados.

Como mencionado, as mesmas análises efetuadas acima podem ser aplicadas na caracterização de cada *division* do *extent* de interesse.

Em seguida, a entidade LANCE_DUTO é caracterizada de maneira análogo a TRECHO_LOGRADOURO. Porém, para mostrar a seletividade, o *extent* foi dividido em *division* segundo o método *quadtree* e todas as ocorrências que estão totalmente incluídas ou que "passam" na *division* são consideradas (primeiro procedimento comentado no capítulo 4).

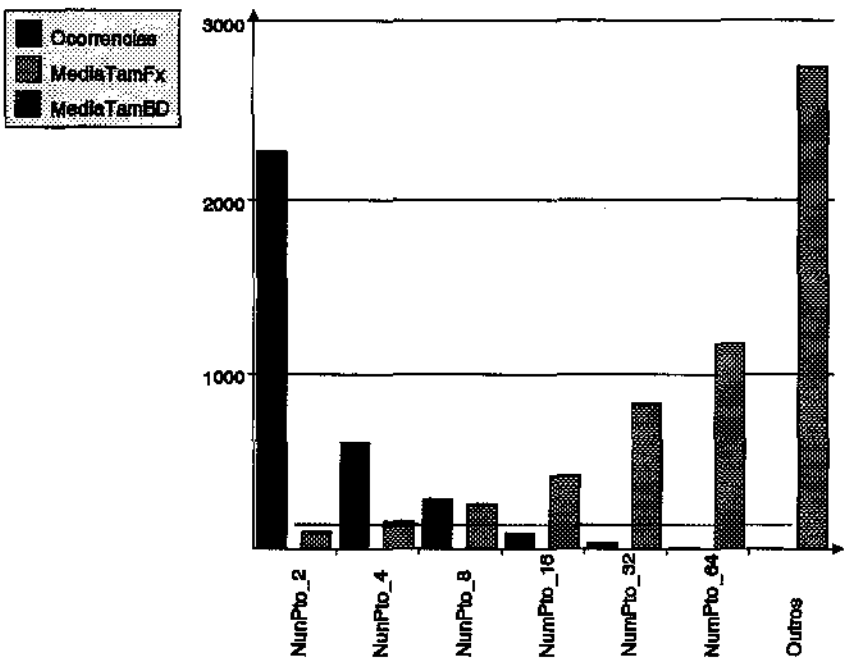

Figura 5.13: Caracterização da entidade TRECHO_LOGRADOURO em relação à complexidade.

Tamanho (m)	Qtde	%	Média Tam. Fx	Média Ptos Fx
100	1.720	52,23%	62,88	2,17
200	1.018	30,91%	140,51	2,84
300	305	9,26%	238,56	4,16
400	101	3,07%	339,12	6,49
500	46	1,40%	448,49	9,08
600	25	0,75%	550,85	8,12
700	21	0,63%	648,79	11,23
Outros	57	1,73%	1.057,75	21,21
Total	3.293	99,98%	-	-

Tabela 5.18: Caracterização da entidade TRECHO_LOGRADOURO em relação ao tamanho.

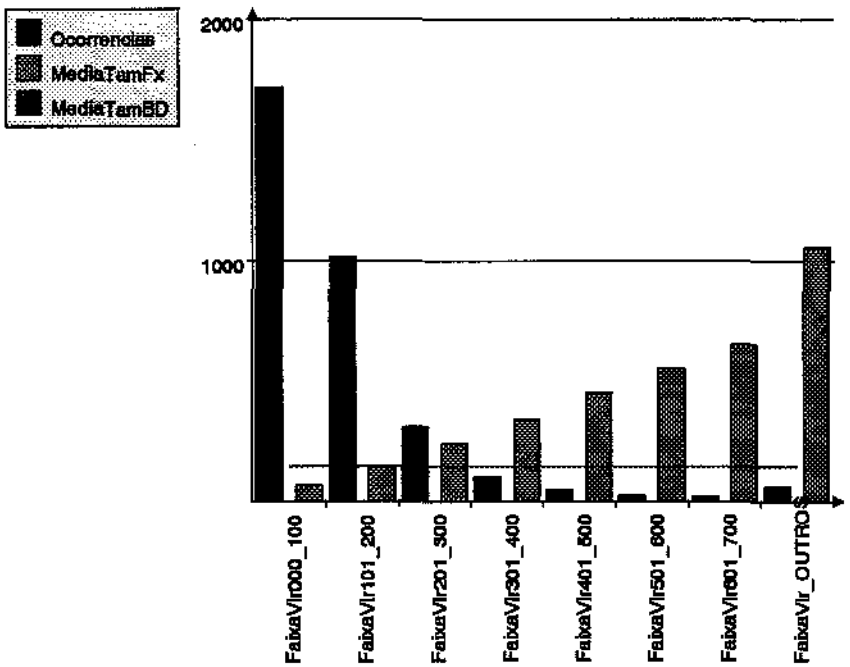

Figura 5.14: Caracterização da entidade TRECHO_LOGRADOURO em relação ao tamanho.

MBR (m^2)	Qtde	%	Média Area Fx
5000	2.135	64,83%	1.807,23
10000	452	13,72%	7.058,39
20000	355	10,78%	13.975,75
40000	179	5,44%	27.692,37
80000	67	2,03%	57.572,06
160000	50	1,52%	116.658,90
320000	35	1,06%	225.876,62
640000	14	0,42%	445.131,75
Outros	6	0,18%	1.650.358,49
Total	3.293	99,98%	-

Tabela 5.19: Caracterização da entidade TRECHO_LOGRADOURO em relação ao MBR.

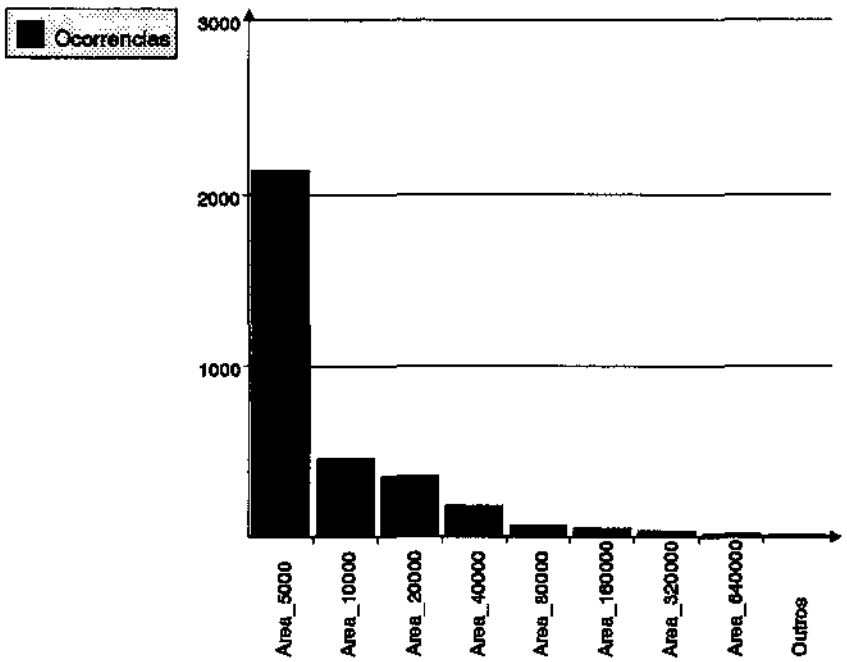

Figura 5.15: Caracterização da entidade TRECHO_LOGRADOURO em relação ao MBR.

A caracterização em relação aos dados convencionais encontra-se na tabela 5.5. Para esta entidade, os dados convencionais ocupam 32 *bytes* e tem-se 376 ocorrências no banco de dados.

A figura 5.16 ilustra a distribuição/seletividade dos dados não convencionais em cada *division* do *extent*.

Em relação à complexidade, a tabela 5.20 e figura 5.17 sumariza estes dados para a entidade LANCE_DUTO em relação ao banco de dados completo.

Em seguida, caracteriza-se a entidade LANCE_DUTO quanto ao tamanho. A tabela 5.21 e figura 5.18 sumariza as ocorrências da entidade no banco de dados.

A caracterização da entidade LANCE_DUTO em relação ao MBR é mostrado na tabela 5.22. A figura 5.19 ilustra os dados presentes no banco de dados graficamente.

Os valores mostrados na caracterização destas entidades do tipo linha mostram as diversidades possíveis de ocorrências quando analisamos os fatores complexidade, tamanho e forma geométrica. Neste caso, o ideal seria obter a caracterização de cada entidade separadamente e generalizar seus valores para outras entidades somente quando houver certeza de que estas possuem as mesmas características.

A seguir, os exemplos apresentados serão discutidos quanto à distribuição de *Zipf*.

No caso da entidade TRECHO_LOGRADOURO, para o atributo complexidade, é

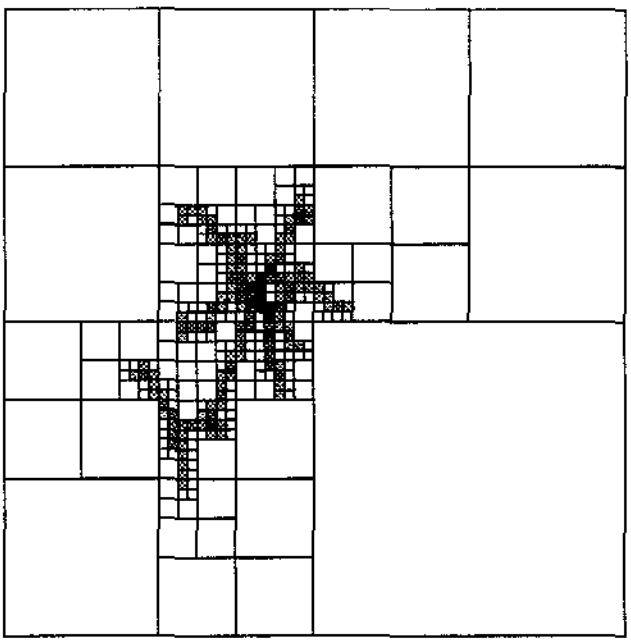

Figura 5.16: Distribuição da entidade LANCE_DUTO sobre o *extent*.

Complexidade	Qtde	%	Média Ptos Fx	Média Tam. Fx
02	223	59,31%	2	94,00
04	16	4,26%	3,43	119,04
08	62	16,49%	6,41	115,90
16	58	15,43%	12,06	124,46
32	15	3,99%	21,13	137,74
64	2	0,53	37,50	143,42
Outros	0	-	-	-
Total	376	99,98%	-	-

Tabela 5.20: Caracterização da entidade LANCE_DUTO em relação à complexidade.

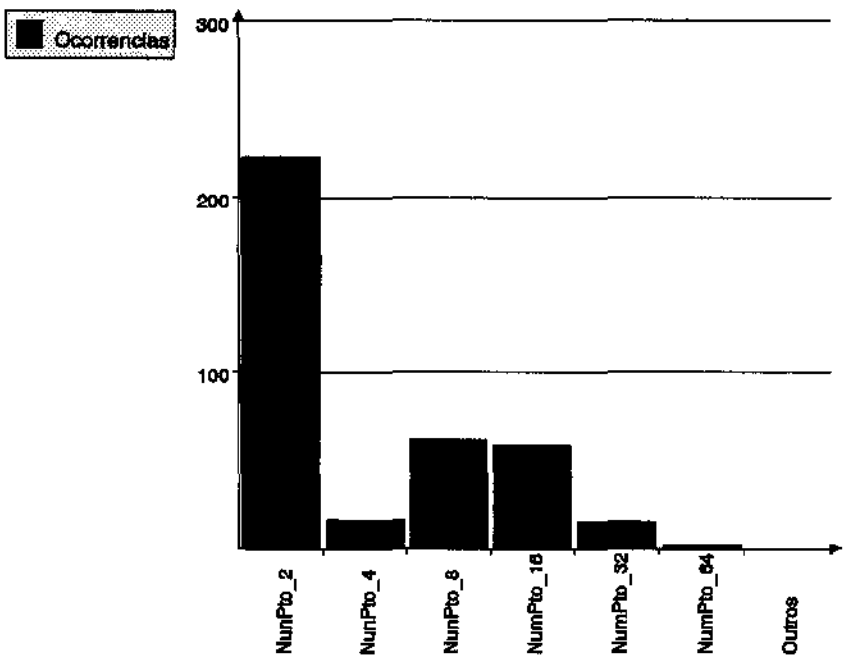

Figura 5.17: Caracterização da entidade LANCE_DUTO em relação à complexidade.

Tamanho (m)	Qtde	%	Média Tam. Fx	Média Ptos Fx
100	168	44,68%	64,23	4,09
200	201	53,46%	135,92	6,22
300	7	1,86%	219,99	7,28
400	0	-	-	-
500	0	-	-	-
600	0	-	-	-
700	0	-	-	-
Outros	0	-	-	-
Total	376	100,00%	-	-

Tabela 5.21: Caracterização da entidade LANCE_DUTO em relação ao tamanho.

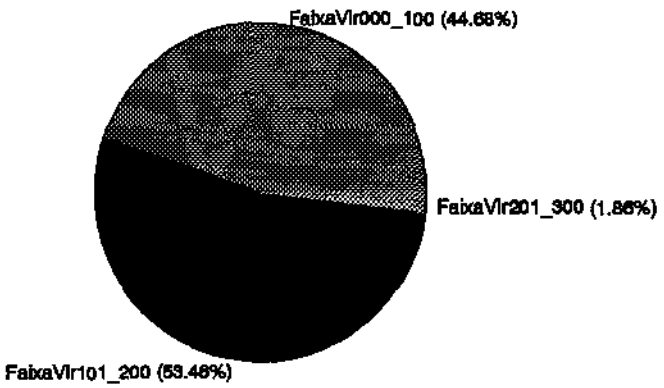

Figura 5.18: Caracterização da entidade LANCE_DUTO em relação ao tamanho.

MBR (m^2)	Qtde	%	Média Area Fx
5000	249	66,22%	2.030,44
10000	94	25,00%	6.909,09
20000	30	7,96%	13.090,61
40000	3	0,80%	23.181,94
80000	0	-	-
160000	0	-	-
320000	0	-	-
Outros	0	-	-
Total	376	99,98%	-

Tabela 5.22: Caracterização da entidade LANCE_DUTO em relação ao MBR.

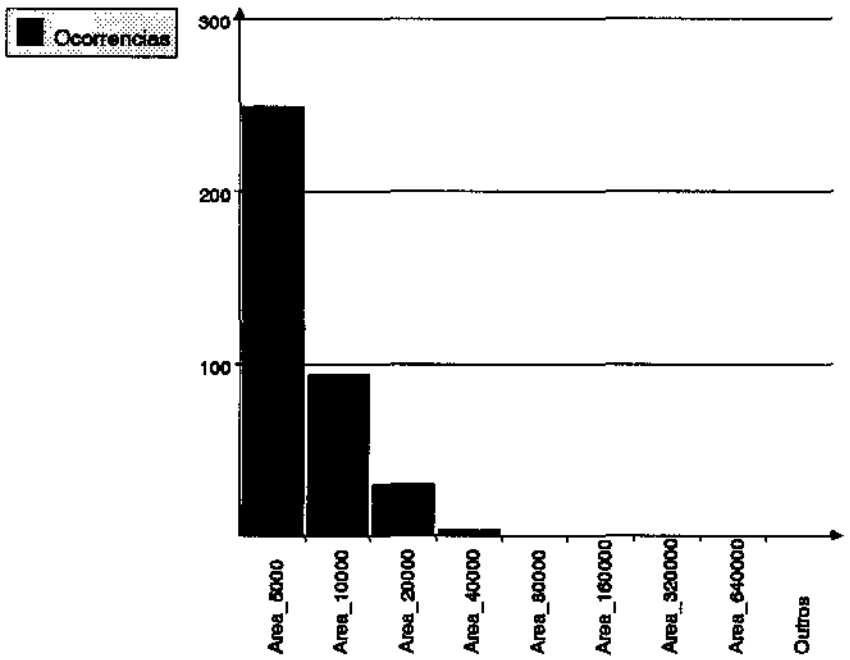

Figura 5.19: Caracterização da entidade LANCE_DUTO em relação ao MBR.

indicado o valor do fator de decaimento Z=2,2. A figura 5.20 demonstra tal tendência. Nesta análise, considerou-se o *rank* como sendo a quantidade de ocorrências dentro de cada faixa de valores de complexidade considerada na verificação dos dados reais.

Em relação ao tamanho, observou-se que o valor do fator de decaimento Z=1.9 é indicado. A figura 5.21 demonstra tal tendência. Para a obtenção deste valor, considerou-se que o *rank* é dado pelas faixas de tamanho utilizadas na análise dos dados reais, no caso, 8 faixas diferentes (o valor 1 da figura corresponde à faixa de 100, e assim sucessivamente). Se forem considerados todos os valores diferentes do banco, têm-se 3.293 valores diferentes de tamanho.

A complexidade da entidade LANCE_DUTO é analisada semelhante à entidade TRECHO_LOGRADOURO. A figura 5.22 ilustra a ordem dos *ranks* de acordo com o número de ocorrências de entidades nas faixas. Em relação ao tamanho, esta entidade deve ser caracterizada apenas em duas faixas de valores devido a sua concentração nestas.

Quanto à forma geométrica, considerando as classificações de [Pap95], [ND97] e [TP95], tem-se a tendência destas entidades possuirem MBRs considerados pequenos.

Na geração destes dados, se forem consideradas as médias de valores dentro de cada faixa (expressas na tabela 5.18) obtém-se valores mais significativos.

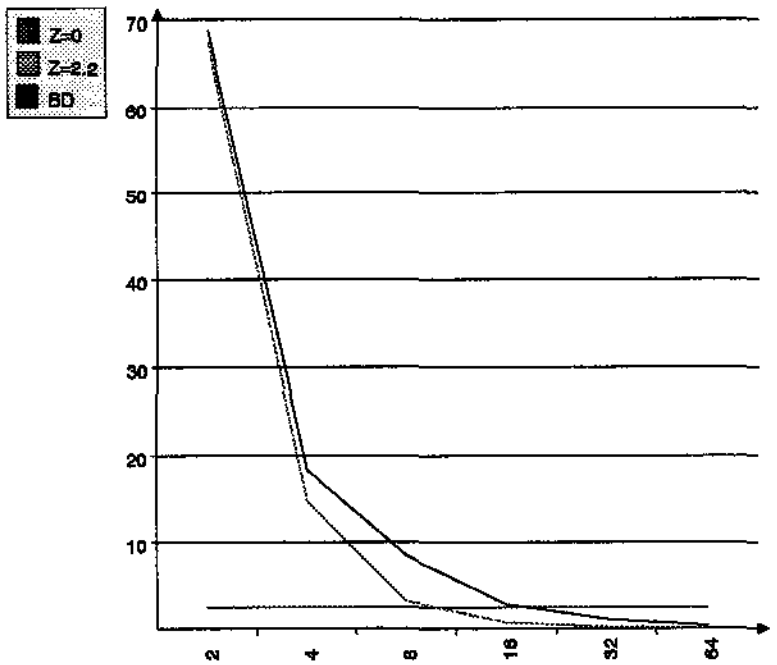

Figura 5.20: Distribuição da complexidade em relação à entidade TRECHO_LOGRADOURO.

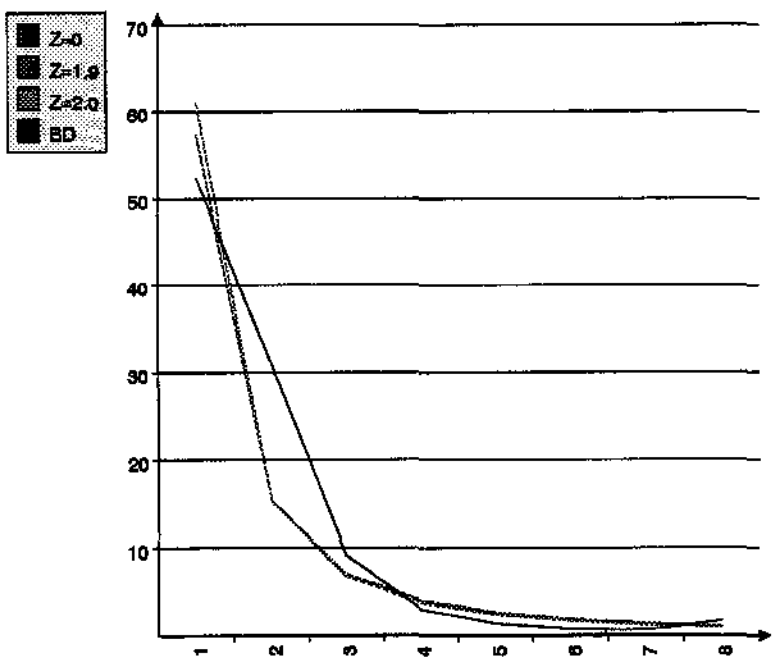

Figura 5.21: Distribuição do atributo tamanho em relação à entidade TRECHO_LOGRADOURO.

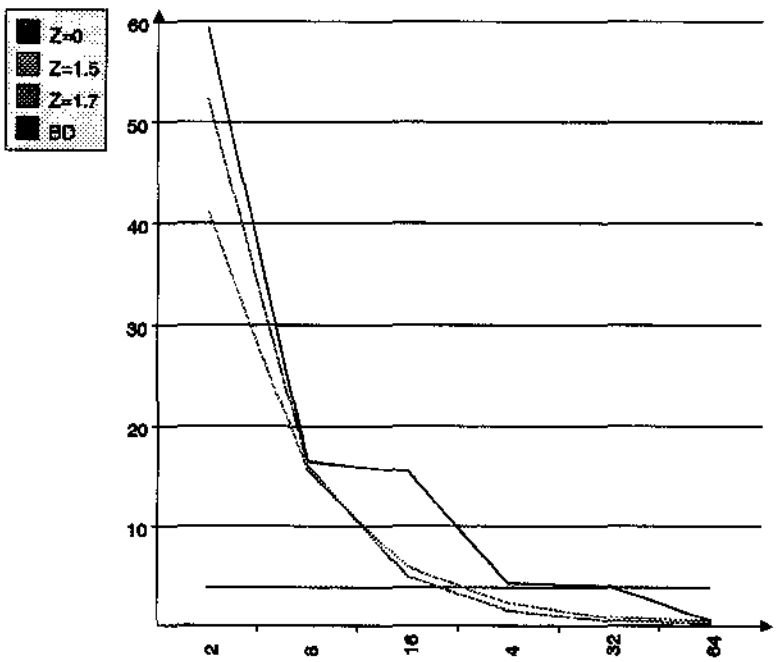

Figura 5.22: Distribuição da complexidade em relação à entidade INDICAÇÃO_LOTE.

5.4.4 Caracterização das entidades do tipo polígono

Na caracterização de entidades do tipo polígono, deve-se levar em consideração, além dos dados convencionais, os fatores complexidade, tamanho e forma geométrica, semelhante a linhas.

Sendo assim, inicialmente, caracteriza-se a entidade QUADRA em relação aos dados convencionais. Neste caso, os dados convencionais ocupam 4 *bytes* pois trata-se de uma entidade gráfica - tabela 5.9.

Quanto aos dados não convencionais, a figura 5.23 ilustra a distribuição/seletividade dos dados em cada *division* do *extent*. Para o cálculo da seletividade, o *extent* foi dividido em *division* segundo o método *quadtree* e todas as ocorrências que estão totalmente incluídas ou que "passam" na *division* são consideradas (primeiro procedimento comentado no capítulo 4). Também, a iteração ocorre até que se tenha o lado da *division* maior ou igual a 250 metros, ou seja, que a àrea seja superior a $62.500m^2$.

Em relação à complexidade, a tabela 5.23 sumariza estes dados para a entidade QUADRA em relação ao banco de dados completo. A figura 5.24 ilustra estes dados graficamente.

A caracterização desta entidade em relação ao tamanho é sumarizada na tabela 5.24 e figura 5.25 sumariza as ocorrências da entidade no banco de dados.

Em seguida, caracteriza-se a entidade QUADRA em relação ao MBR. A tabela 5.25

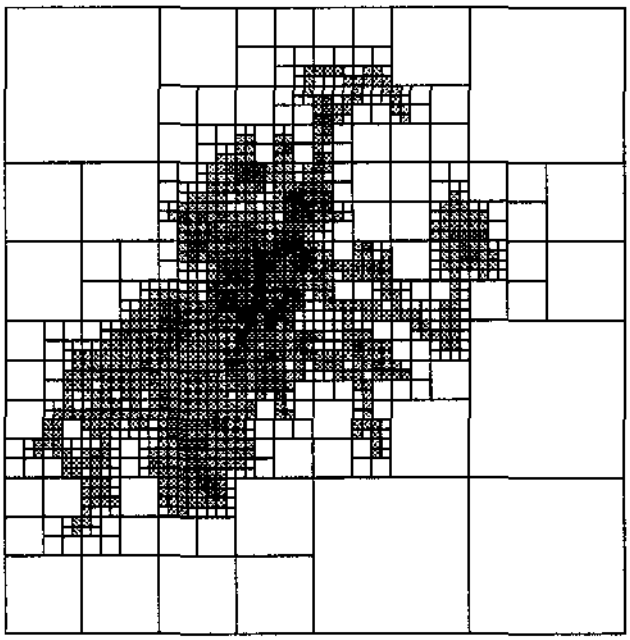

Figura 5.23: Distribuição da entidade QUADRA sobre o *extent*.

e figura 5.26 sumariza os dados presentes no banco de dados.

Novamente, verificou-se a diversidade de valores possíveis de ocorrências quanto aos fatores complexidade, tamanho e forma geométrica. Para esta entidade polígono apresentada, a distribuição dos valores quanto à complexidade e tamanho apresentam-se bastante uniformes dentro de faixas de valores, não se mostrando adequado a distribuição de *Zipf* conforme mostra a figura 5.27. Por sua vez, a forma geométrica pode ser classifica em 2.005 MBRs pequenos, 261 MBRs médio e 44 MBRs grandes.

5.5 Ambiente de execução e testes

O projeto SAGRE vem sendo desenvolvido no seguinte ambiente operacional:

- *Hardware*:
  - *Workstations* SUN e IBM.
  - Microprocessadores PC.
  - Mesa digitalizadora.
  - *Plotters* e impressoras *laser*.
  - Rede *Ethernet* realizando a interligação entre os equipamentos.

Complexidade	Qtde	%	Média Ptos Fx	Média Tam. Fx
02	-	-	-	-
04	117	5,06%	3,52	66,33
08	179	7,75%	6,59	133,71
16	434	18,79%	12,44	245,66
32	669	28,96%	23,49	398,55
64	509	22,03%	45,73	571,74
128	280	12,12%	88,83	844,48
Outros	122	5,28%	198,20	1422,26
Total	2.310	%	-	-

Tabela 5.23: Caracterização da entidade QUADRA em relação à complexidade.

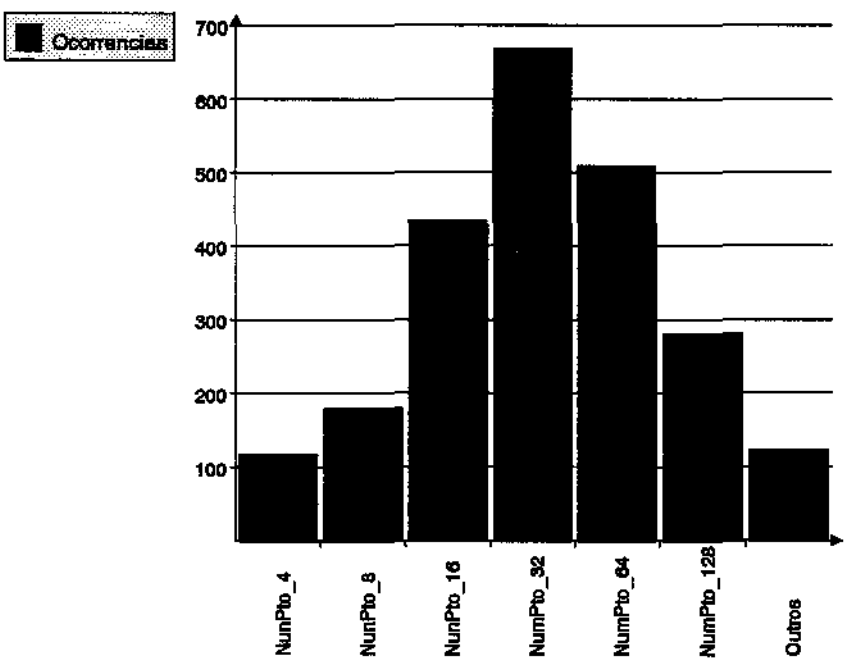

Figura 5.24: Caracterização da entidade QUADRA em relação à complexidade.

Tamanho (m)	Qtde	%	Média Tam. Fx	Média Ptos Fx
200	657	28,44%	94,28	14,84
400	727	31,47%	300,56	35,15
800	604	26,15%	540,26	48,54
1600	221	9,57%	1.118,46	69,64
3200	86	3,72%	2.162,33	126,76
6400	15	0,65%	4.401,85	274,46
Outros	0	-	-	-
Total	2.310	%	-	-

Tabela 5.24: Caracterização da entidade QUADRA em relação ao tamanho.

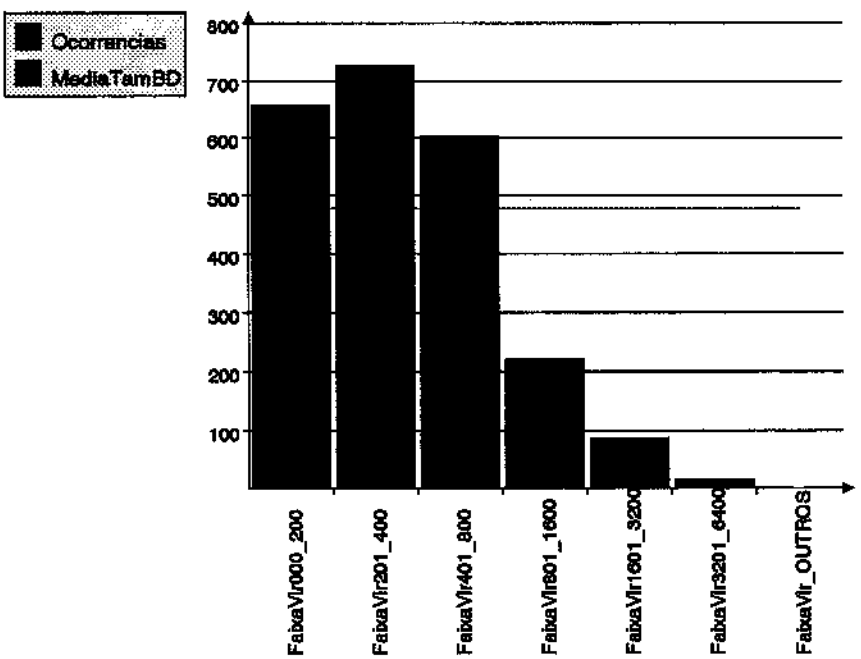

Figura 5.25: Caracterização da entidade QUADRA em relação ao tamanho.

MBR (m^2)	Qtde	%	Média Area Fx
5000	606	26,33%	1.601,82
10000	322	13,94%	7.539,76
20000	487	21,08%	14.572,92
40000	346	14,98%	28.120,44
80000	166	7,19%	55.885,33
160000	163	7,06%	112.511,52
320000	91	3,94%	235.608,99
640000	61	2,64%	463.082,72
Outros	68	2,94%	1.800.584,06
Total	2.310	100,00%	

Tabela 5.25: Caracterização da entidade QUADRA em relação ao MBR.

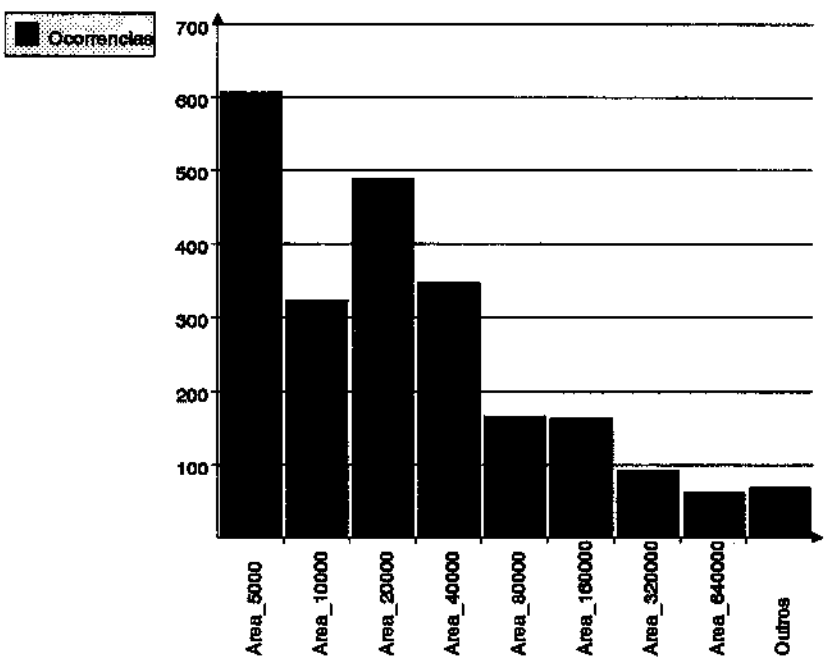

Figura 5.26: Caracterização da entidade QUADRA em relação ao MBR.

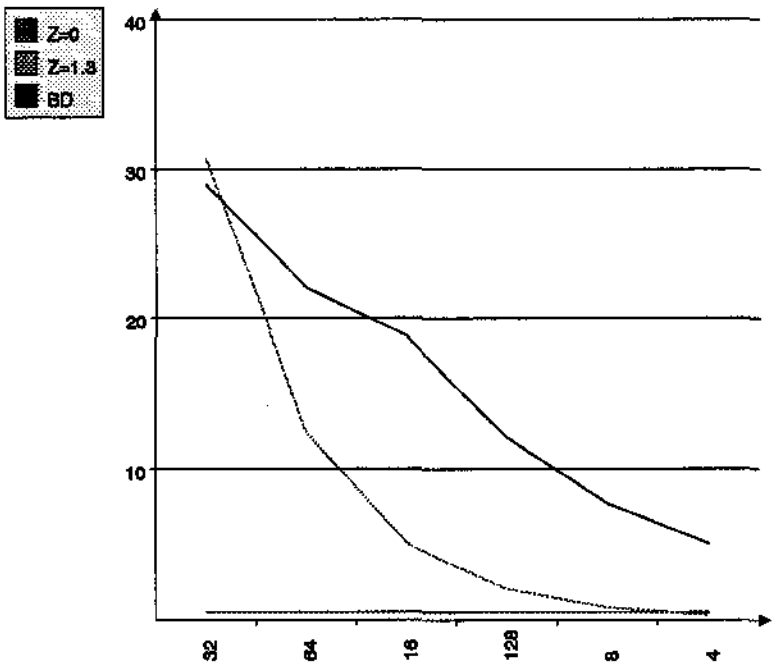

Figura 5.27: Distribuição do atributo complexidade em relação à entidade Quadra.

- *Software*:
  - Sistema operacional UNIX.
  - Programas desenvolvidos em linguagem C.
  - SGBD relacional padrão SQL (ORACLE).
  - Ferramentas CASE para especificação do sistema, modelagem de dados, documentação etc.
  - Sistema de informações geográficas VISION*, da SHL.

Além do conhecimento necessário do ambiente da aplicação onde a caracterização dos dados deve ser efetuada (como comentado acima), outros fatores devem ser levados em consideração. Alguns deles são:

- O banco de dados da aplicação tem que ser "congelado", ou seja, ao final do processo de análise deve-se ter os mesmos dados observados antes de iniciar o processo. Qualquer alteração deve ser cuidadosamente verificada. Tal fato sugere uma interação constante com o usuário.

- Os trabalhos devem ser desenvolvidos de forma a não prejudicar o desenvolvimento e operação da aplicação em análise. No caso, indica-se horários fora do expediente do usuário.

- A execução dos programas de análise pode acarretar sobrecarga de execução ao ambiente do usuário devido ao volume dos dados. Mais uma vez, indica-se a execução desses programas em horários fora do expediente do usuário.

Ainda, nesta dissertação os programas para a geração das informações foram definidos em GML que trata-se de uma linguagem de quarta geração utilizada para definição de programas no ambiente de desenvolvimento do VISION*. A escolha desta linguagem de implementação refere-se simplesmente ao fato de que a aplicação alvo trabalha com a mesma. O *software Business Object* foi utilizado para a geração dos gráficos.

5.6 Carga de trabalho

Consultas visando caracterizar a carga de trabalho de um *benchmark* devem apresentar os seguintes componentes:

- representatividade: ilustra o contexto comercial em que a consulta deve ser usada (motivo), ou seja, sua real existência frente ao a uma aplicação;

- definição da consulta: especifica a consulta de acordo como uma linguagem de programação/consulta adequada ao SGBD. Para cada consulta cria-se um programa executável. A execução dessas consulta retorna uma ou mais linhas que são chamadas de dados de saída e são posteriormente utilizados na validação da consulta;

- parâmetros de substituição: descreve como gerar os valores necessários para completar a sintaxe da consulta. Ainda, muitas das consultas poderiam ser efetuadas diretamente sobre o banco de dados via linguagem de consulta executável (SQL, por exemplo).

- validação da consulta: descreve como validar a consulta de acordo com a qualificação do banco de dados.

Devido ao atual estágio de desenvolvimento do projeto analisado, o usuário não dispõe de consultas expressivas que possam refletir a caracterização proposta nesta dissertação e assim reportar resultados de desempenho. Salienta-se que esta carga de trabalho pode ser definida pelo usuário de acordo com a aplicação em estudo.

5.7 Estatísticas gerais

A figura 5.28 mostra a média dos relacionamentos existentes entre as entidades analisadas. Os relacionamentos contínuos da figura denotam os mesmos relacionamentos do modelo físico enquanto que os relacionamentos tracejados denotam correspondência a nível lógico. Ainda, são indicados os valores de ocorrências de cada entidade presente no banco de dados e os valores de ocorrências das chaves estrangeiras.

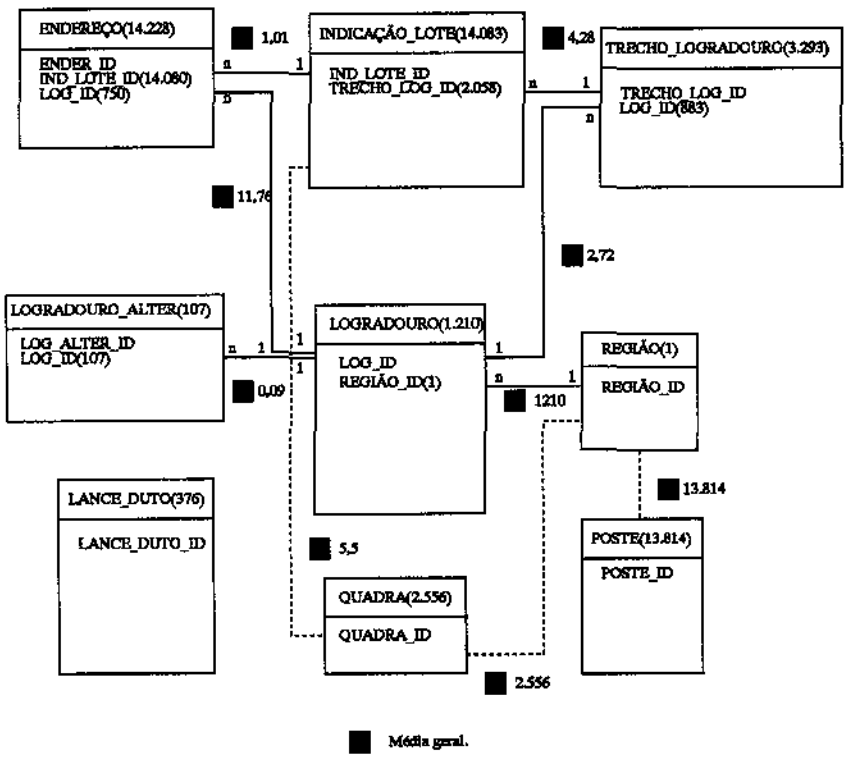

Figura 5.28: Média dos relacionamentos das entidades da aplicação.

As médias desses relacionamentos foram calculadas para fornecerem dados estatísticos. Exemplos sobre a figura 5.28 e de acordo com a aplicação em análise podem ser:

- Média de endereços (ENDEREÇO) por logradouro (LOGRADOURO).

  Neste caso, tem-se que cada logradouro possui em média 11,76 endereços.

- Média de endereços (ENDEREÇO) por indicação de lote (INDICAÇÃO_LOTE).

  Aqui, tem-se a média de 1,01 enderecos por testada de lote. Percebe-se a existência de um relacionamento muito próximo de 1 : 1.

- Média de logradouros alternativoss (LOGRADOURO_ALTER) por logradouro (LOGRADOURO).

Neste caso, tem-se a média de 0,09 logradouros alternativos por logradouro; valor este muito pequeno.

- Média de testadas de lote (INDICAÇÃO_LOTE) por trecho de logradouro (TRECHO_LOGRADOURO).

  Obtem-se a média de 4,28 indicações de lote por trecho de logradouro.

- Média de logradouros (LOGRADOURO) por região (REGIÃO).

  O valor aqui encontrado de 1.210 logradouros por região fica prejudicado visto que analisa-se apenas uma região. Tal média foi ilustrada para ressaltar sua importância no contexto desse tipo de aplicação.

- Média de trechos de logradouro (TRECHO_LOGRADOURO) por logradouro (LOGRADOURO).

  Neste caso, tem-se uma média de 2,72 trechos de logradouro por logradouro.

Ainda, algumas médias podem ser obtidas a nível lógico, ou seja:

- Média de quadras (QUADRA) por região (REGIÃO).

  Mais uma vez estes valores (2.556 quadras por região) ficam prejudicados pois analisa-se apenas uma região.

- Média de lotes (INDICAÇÃO_LOTE) por quadra (QUADRA).

  Neste caso, a média de 5,5 indicações de lote por quadra é encontrada.

- Média de trechos de logradouro (TRECHO_LOGRADOURO) por região (REGIÃO).

  Neste caso, tem-se a média de 3.293 trechos de logradouro por região.

Estes valores, além de caracterizarem a aplicação em questão, podem também sugerirem valores para a geração de dados sintéticos; sugerirem estruturas de armazenamento de dados adequados e validarem a aplicação em questão.

No caso, por exemplo, a média de logradouros alternativos por logradouro (média = 0,09), sugere que a modelagem efetivada à aplicação foi adequada visto que, se colocados os atributos da entidade logradouro alternativo (LOGRADOURO_ALTER), na entidade logradouro (LOGRADOURO) poderia-se acarretar num desperdício de espaço de armazenamento em torno de $1.103 * 30 = 33.099$ *bytes* [4].

[4] O primeiro valor (1.103) corresponde ao número de ocorrências de logradouro menos número de ocorrências de logradouros alternativos e, o segundo valor (30), ao tamanho do nome alternativo, se considerarmos apenas este campo.

5.8 Rotação de entidades do tipo linha

Em relação à forma geométrica, um estudo para verificar os valores gerados dos MBRs foi realizado sobre entidades do tipo linha.

Inicialmente, os MBRs foram calculados de acordo com os valores existentes no banco de dados.

Posteriormente, cada ocorrência da entidade foi rotacionada de acordo com o ângulo obtido entre dois pontos que expressam a maior distância dentre todos aqueles que compõem a linha. Escolheu-se este ângulo pelo fato de que a maioria das ocorrências de linhas nesta aplicação é constituída por dois pontos e também pelo fato destes pontos representarem a maior largura do MBR na maioria das vezes. Foi verificado que se a linha contém a disposição dos pontos na forma de um quadrado ou retângulo, o MBR aumenta no esquema aqui proposto de rotação.

No contexto dessa aplicação, a entidade TRECHO_LOGRADOURO é utilizada como exemplo. Sendo assim, no primeiro momento, obteve-se:

Área banco de dados: $448.500.000m^2$

Área MBRs calculada: $50.697.117,35m^2$

Densidade : $0,11\%$

A densidade, área total dos MBRs dividida pela área total do banco, foi calculada consistindo de $0,11\%$ da área total do banco.

Após a rotação, obteve-se os seguintes valores:

Área MBRs calculada: $12.853.163,59m^2$

Densidade : $0,028\%$

No caso, a área dos MBRs diminuiu em aproximadamente 75% e a densidade que inicialmente consistia de $0,11\%$ passou a $0,028\%$.

Os valores obtidos com a rotação do MBR mostram uma grande variação dos valores e ainda retrata com maior precisão a área realmente ocupada na região de estudo em relação a este tipo de entidade.

5.9 Conclusões

Este capítulo apresentou um estudo de caso frente a uma aplicação real - projeto SAGRE. Este projeto trata-se de um típico sistema de AM/FM construído sobre um SIG.

Através deste projeto, procurou-se aplicar o esquema de caracterização sugerido no capítulo 4. Os resultados obtidos ao longo do capítulo vão ao encontro da proposta desta dissertação visto que um conjunto expressivo de características foram determinadas que possibilitam a geração de dados sintéticos mais próximos de aplicações reais.

Estas características são úteis na validação de propostas e estudos na área de análise de desempenho de sistemas de banco de dados espaciais. Ponto fundamental aqui, é que os resultados obtidos com este esquema de caracterização possibilita a geração de aplicações sintéticas em outros ambientes computacionais, para em seguida, analisar seu desempenho. Tais resultados podem indicar tendências e necessidades futuras nestes ambientes computacionais.

Ainda, devido ao volume dos dados, alguns mecanismos de visualização dos dados foram determinados (tabelas e figuras). Tal fato pode auxiliar na análise de outras aplicações.

Capítulo 6

Conclusões

Esta dissertação apresentou um esquema de caracterização de sistemas de bancos de dados espaciais para análise de desempenho utilizando dados sintéticos.

No contexto de análise de desempenho de sistemas de banco de dados, o conceito de *benchmark* de banco de dados é empregado. Esta técnica consiste na execução de um conjunto de transações sobre um banco de dados conhecido para reportar características de desempenho. Neste caso, o esquema de caracterização proposto visa dar uma "identidade" à aplicação, ou seja, caracterizá-la tanto quanto a dados convencionais quanto a dados não convencionais em relação a fatores como: tipos de entidade, tipos de dados, quantidade, distribuição, controle da localização e seletividade, possibilitando, posteriormente, que os dados sejam gerados sintéticamente. A partir desta caracterização, propostas e estudos na área de análise de desempenho de banco de dados espaciais podem ser validados. Por sua vez, estes dados podem ser gerados em outros ambientes computacionais e estes analisados em relação ao desempenho.

Paralelamente, o mesmo esquema foi elaborado para ser aplicado perante uma aplicação real e obter as características que melhor representam uma aplicação real.

Ainda, um conjunto expressivo de transações que poderão compor a carga de trabalho da aplicação e, conseqüentemente, do *benchmark*, foi apresentado. Devido ao atual estágio de desenvolvimento do projeto analisado, não foi possível dispor de consultas expressivas que pudessem ser utilizadas na caracterização proposta nesta dissertação e assim reportar resultados de desempenho

Observa-se que o objetivo não é criar um *benchmark* padrão como acontece com os *benchmarks* de Wisconsin e Débito e Crédito e sim, possibilitar a avaliação de SGBDs frente a uma aplicação mais próxima daquela que seria implementar sem ainda dispor dos dados desta.

O esquema foi verificado perante uma aplicação real - o projeto SAGRE - Sistema Automatizado de Gerenciamento de Rede Externa da Telebrás Telecomunicações Brasileiras

S/A.

Ainda, os trabalhos de [Per90] e [Cif95] foram validados em relação à geração de dados convencionais e não convencionais, respectivamente. Quanto a dados convencionais, [Per90] propôs uma metodologia que possibilita a análise de aplicações do usuário obtendo resultados adequados e próximos do esquema de caracterização aqui proposto. Quanto a dados não convencionais, o trabalho de [Cif95] foi validado com os resultados deste estudo.

Resumidamente, as principais contribuições desta dissertação são:

- Estudo experimental.
- Caracterização de aplicações típicas de sistemas AM/FM.
- Validação da caracterização proposta junto a uma aplicação real.
- Estudo preliminar para analisar variações de rotação para cálculo do MBR para objetos geográficos do tipo linha.

Com relação a extensões que poderiam enriquecer este trabalho, propõe-se:

- verificar o esquema de caracterização proposto junto a outros tipos de aplicações (Ambientais, por exemplo).
- aplicar as características obtidas no capítulo 5 perante outros ambientes computacionais e verificar o comportamento dos mesmos. Pode-se até mesmo, caracterizar outros modelos de SGBD que não sejam o relacional (por exemplo, modelo OO ou com suporte a dados temporais).
- Validar o esquema proposto para a análise de desempenho de banco de dados espaciais em relação a transações dos usuários (carga de trabalho), ou seja, transações da própria aplicação. No caso, os resultados das consultas deveriam refletir as caracterizações aqui sugeridas e o tempo de execução destas poderia ser analisado.
- Analisar outras hipóteses de seletividade de dados convencionais. Assume-se nesta dissertação a hipótese de uniformidade apresentadas em [Bou88].
- Analisar variações de rotação para cálculo do MBR para objetos geográficos do tipo polígono, podendo-se verificar a possibilidade de outras hipóteses.

Bibliografia

[Agu95] C. D. Aguiar. Integração de Sistemas de Banco de Dados Heterogêneos em Aplicações de Planejamento Urbano. Tese de Mestrado, DCC-IMECC-UNICAMP, 1995.

[Bou88] P. Boursier. Analysis of Urban Geographic Queries. In *New Trends in Computer Graphics*, pp. 601–610. N. M. Thalmann e D. Thalmann, 1988. Spring - Verlag.

[Cam95] G. Camara. *Modelos, Linguagens e Arquiteturas para Bancos de Dados Geográficos.* Tese de Doutorado, INPE, dezembro de 1995.

[CCH+96] G. Câmara, M. A. Casanova, A. S. Hemerly, G. C. Magalhães, e C. M. Bauzer Medeiros. *Anatomia de Sistemas de Informações Geográficas.* UNICAMP, julho de 1996.

[CdFvO93] E. Clementini, P. di Felice, e P. van Oosterm. A small set of formal tipological relationship suitable for end-user interaction. *In 3rd Synposium on Spatial Dabase Systems*, pp. 227–295, 1993.

[Cer96] N. Cereja. Visões em Sistemas de Informações Geográficas - modelo e mecanismos. Tese de Mestrado, IC-UNICAMP, 1996.

[Cha95] A. B. Chaudhri. An Annoted Bibliography of Benchmarks for Object Databases. *SIGMOD RECORD*, 24(1):50–55, março de 1995.

[Che83] P. A. Chen. A Preliminary Framework for Entity-Relationship Models. *Entity-Relationship Aproach to Information Modeling and Analysis*, pp. 19–28, 1983.

[Chr84] S. Christodoulakis. Implications of Certain Assumptions in Database Performance Evaluation. *ACM TODS 9, 2*, 1984.

[Cif95] R. R. Ciferri. Um Benchmark voltado a Análise de Desempenho de Sistemas de Informações Geográficas . Tese de Mestrado, DCC-IMECC-UNICAMP, 1995.

[Cox91] F. S. Cox. Análise de Métodos de Acesso a Dados Espaciais Aplicados a Sistemas Gerenciadores de Banco de Dados. Tese de Mestrado, Universidade Estadual de Campinas, dezembro de 1991.

[Dat86] C. J. Date. *An Introduction to Database Systems.* Addison-Wesley Publishing Company Inc., USA, 1986.

[Dew91] D. J. Dewitt. *The Wisconsin Bechmark: Past, Present, and Future*, capítulo 3, pp. 119–162. Jim Gray, 1991.

[EN94] R. Elmasri e S. M. Navathe. *Fundaments of Database Systems.* USA, 1994.

[GEOG94] J. Gonçalves, M. Erhardt, F. Obata, e S. Granado. Automação de Cadastros de Rede Externa. In *I Encontro de Qualidade de Redes de Telecomunicações e de Equipamento Terminais*, 1994.

[Gra91] J. Gray. *The benchmark Handbook for Database and Transaction Processing Systems.* Morgan Kaufmann Publishers, 1991.

[Kim95] W. Kim and J. F. Garza. Requirements for a performance benckmark for object-oriented databases systems. In Won Kim, editor, *Modern Database Systems: The Object Model, Interoperability and Beyond*, pp. 203–215, 1995.

[KS89] H. F. Korth e A. Silberschatz. *Sistemas de Bancos de Dados.* 1989.

[Mag93] G. C. Magalhães. Projeto SAGRE - Sistema Automatizado de Gerência de Rede Externa. Fator GIS, n. 3, p.26-28, 1993. Telebrás - Telecomunicações S/A.

[Mag94] G. C. Magalhães. Especificação Técnica de Conversão de Dados - Proposta da Telebrás - Projeto SAGRE. In *GIS Brasil 94 - Congresso e Feira para Usuários de Geoprocessamento (anais)*, pp. 43–52, Curitiba, Pr, Brasil, 1994.

[MGR93] D. Maguire, M. Goodchild, e D. Rhind. *Geographical Information Systems - volume II - Applications.* John Wiley and Sons, 2 edição, 1993.

[MGS+94] G. Magalhães, A. Giglioni, C. Santos, D. Teijero, e E. Argondizio. Especificação Técnica de Conversão de Dados Proposta da Telebrás - Projeto SAGRE. *Anais GIS-Brasil*, pp. 43–52, 1994.

[MM93] C.M.B. Medeiros e G. C. Magalhães. Rule Aplication in GIS - a Case Study. Technical report, DCC - IMECC - UNICAMP, 1993.

[ND97] M. A. Nascimento e M. H. Dunham. Using Parallel B+ trees as a Practical Alternative to the Classical R-tree. *SBBD*, 1997.

[Ooi90] B. C. Ooi. Efficient Query Processing in Geographic Information Systems. In *Lecture Notes in Computer Science*, volume 471, pp. 1–209. Spring Verlag, 1990.

[Pap95] D. Papadias et al. Topological Relations in the World of Minimum Bounding Rectangles: A Study with R-trees. *Procedings of the 1995 ACM SGMOND International Conference on Managment of Data*, pp. 92–103, junho de 1995. Montreal, Canada.

[Per90] R. F. Pereira. Análise de Desempenho de Banco de Dados Utilizando Benchmarks Especializados. Tese de Mestrado, DCC - IMECC - UNICAMP, 1990.

[Sea95] D. Seaborn. Database Managment in GIS - Past, Present, Future: An Enterprise Perspective for Executives. *Technical Paper Series - SHL Vision solutions, Ottawa, Canadá*, pp. 1–12, 1995.

[Ser91] O. Serlin. *The History of DebitCredit and the TPC*, capítulo 2, pp. 19–117. Jim Gray, 1991.

[SHL97] Inc SHL Systemhouse. *Vision* Concepts*, 1997.

[TEL93] TELEBRAS. *Aquisição de Dados de Mapeamento Urbano Básico - Recomendações para o Projeto SAGRE*, 1993.

[Tim94] V. C. Times. Um Modelo Orientado a Objetos para Aplicações Geográficas. Tese de Mestrado, Recife: DI - UFPe, 1994.

[Tom91] C. D. Tomlin. Geographical Information Systems, Principles and Application. In M. F. Goodchild D. J. Maguire e D.W. Rhind, editores, *Cartographic Modelling*, volume 1, pp. 371–384. Longman Group UK Limites, England, 1991.

[TP95] Y. Theodoridis e D. Papadias. Range Queries involving Spatial Relations: a Performance Analysis. *Procedings of the Second International Conference on Spatial Information Theory (COSIT'95)*, setembro de 1995. Semmering, Austria.

[Vas96] R. C. S. Vasconcelos. Análise Comparativa do uso dos Modelos Relacional e Orientado a Objetos em Sistemas de Informações Geográficas. Tese de Mestrado, IC - UNICAMP, 1996.