

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ – UTFPR
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO
DE SISTEMAS

PAULO EDUARDO FERREIRA DE LIMA

ÍNDICES PARA CONSULTAS ESPACIAIS EM BANCO DE DADOS

TRABALHO DE DIPLOMAÇÃO

MEDIANEIRA

2011

PAULO EDUARDO FERREIRA DE LIMA

ÍNDICES PARA CONSULTAS ESPACIAIS EM BANCO DE DADOS

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas da Universidade Tecnológica Federal do Paraná, Campus Medianeira.

Orientador: Prof. MSc. Claudio Leones Bazzi.

MEDIANEIRA – PR

2011

DEDICATÓRIA

A meu pai Sebastião Lima, pois cada gota de suor sua valeu a pena. Minha mãe, mesmo distante... A senhora também foi crucial. Meu redentor Jesus Cristo peça chave nessa conquista...

AGRADECIMENTOS

Primeiramente todos os meus agradecimentos e louvores são a Deus, pois se não fosse a sua intervenção em dado momento da minha vida com certeza eu não estaria concluindo esta etapa, foram suas mãos fortes que me sustentaram em momentos difíceis e na angústia seu amor e carinho invisível aos olhos de muitos aos meus olhos foi revelado. Obrigado Deus por me dar forças quando eu não tinha, obrigado Deus por me animar quando eu estava triste, obrigado Deus pelo seu amor.

A minha família em geral, pai às vezes a distância entre nós parece tão grande mais a sua perseverança sempre me acompanhou e a sua trajetória de vitórias é o meu maior incentivo para lutar e vencer, obrigado pela paciência, pelas broncas enfim por tudo. Mãe minha doce e amada, Deus seja louvado pela sua vida, você é uma guerreira e vencedora, e sua contribuição é oculta aos olhos de muitos, mais eu e Deus sabemos que você foi crucial em todos os momentos. Carlos e Elise meus amados irmãos que sempre torceram para que eu viesse a conseguir essa vitória amo vocês de todo meu coração obrigado por tudo. Vó a senhora também foi importante, pois sempre me acordou quando eu com preguiça não queria levantar para vir estudar.

As amizades verdadeiras que fiz, pois não só de estudos foram esses anos, grandes amizades fiz dentro desta universidade, e o valor delas sempre vou recordar, Marcos, Vinicius e tantos outros irmãos que descobri que tenho, já depois de grande.

A todos os professores da instituição (Universidade Tecnológica Federal do Paraná) pelo rico ensino que me passaram, e ao Claudio Bazzi, que me deu a oportunidade de estar realizando este trabalho (quem sabe um emprego no futuro). Valeu Bazzi - A todos os meus sinceros agradecimentos.

“Porque aonde habitou o pecado, superabundou a graça de Deus.”

Romanos 5: 20b

RESUMO

Com o desenvolvimento das formas de armazenamento de informação nos dias atuais, os bancos de dados vêm sofrendo constantes modificações, novas formas de armazenamento têm surgido juntamente com novas ferramentas, junto a todas essas novidades que estão em ascensão no mercado, estão as formas de armazenamento de atributos geográficos (geoprocessamento).

O objetivo deste trabalho foi realizar estudos sobre índices para realização de consultas espaciais, em banco de dados com suporte a este tipo de dado. O trabalho corresponde a uma pesquisa bibliográfica sobre índices e foram realizados testes utilizando dados reais, visando avaliar o desempenho de cada estrutura utilizada.

Palavras-chave: Geoprocessamento, Consultas Espaciais, Índices Espaciais.

ABSTRACT

With the development of forms of information storage in the present day, the databases have been under constant change, new forms of storage have emerged along with new tools, along with all this new stuff is on the rise in the market, are the storage form attributes of geographic (GIS).

The objective of this study is to conduct studies on indices to perform spatial queries in the database that supports this type of data. The work represents a literature search on indexes and tests will be performed using real data to evaluate the performance of each structure used.

Keywords: Spatial Queries. Geoprocessing.

LISTA DE FIGURAS

Figura 1 - Representação de um mapa e grade (a) e um mapa vetorial (b).....	14
Figura 2 - Sistema de Informação Geográfica.....	17
Figura 3 - Arquitetura de um SIG.....	19
Figura 4 - Arquitetura Dual	20
Figura 5 - Arquitetura Integrada.....	21
Figura 6 - Diferença da Arquitetura Dual para a Arquitetura Integrada	21
Figura 7 - Representação do acesso a um Banco de Dados.....	25
Figura 8 - Processamento de consultas espaciais	26
Figura 9 - Tipos de dados espaciais.....	28
Figura 10 - Exemplo de partial match queries.....	29
Figura 11 - Exemplo de Container Rager Query	29
Figura 12 - Exemplo de Adjacency query	30
Figura 13 - Exemplo de Direction Query	31
Figura 14 - Organização de uma R-Tree	32
Figura 15 - Arvore R e retângulos correspondentes.....	33
Figura 16 – Sequência de execução do Índice GIST.....	34
Figura 17 - Criação de um Índice GIST.	35
Figura 18 - Elementos de uma B-tree.....	36
Figura 19 - SQL Editor.....	39
Figura 20 - ShapeFile carregados no PgAdmin.....	40
Figura 21 - Funções fornecidas pelo PgAdmin	41
Figura 22 - Consulta na tabela sp_municipios.....	41
Figura 23 - select na tabela sp_municipios.....	42
Figura 24 - Consulta aplicada sem utilização de índices.....	43
Figura 25 - Resultado da Consulta sem utilizar índices	43
Figura 26 - Criação do índice GiST.....	44
Figura 27 - Resultado da consulta utilizando índices.....	44
Figura 28 - Consulta aplicada sem índice B-tree.....	45
Figura 29 - Análise da Consulta sem utilização de Índices.....	45
Figura 30 - Criação do índice B-tree	45
Figura 31 - Utilização do índice B-Tree.....	46
Figura 32 - Resultados Obtidos	47

LISTA DE SIGLAS

SIG - Sistema de Informação geográfica.

GPS - Global Positioning System.

SGBD - Sistema de Gerenciamento de Banco de Dados.

BLOB - Binary Large Object.

SQL - Structured Query Language.

SGBDOR Sistema Gerenciador de Banco se Dados Objeto-Relacional.

WMS - Web Map Service.

OGC - Open GeoSpatial Consortium.

XML - Extensible Markup Language.

WFS - Web File Service.

HTTP - Hyper Text Transfer Protocol.

BD - Banco de dados.

MAE - Método de Acesso Espacial.

GIST - Generalized Search Tree.

SUMÁRIO

1.	INTRODUÇÃO	12
1.1.	OBJETIVOS	13
1.1.1.	Objetivo Geral	13
1.1.2.	Objetivos Específicos	13
2.	GEOPROCESSAMENTO	14
2.1	História do Geoprocessamento	15
2.2	SIG - SISTEMA DE INFORMAÇÃO GEOGRÁFICA	17
2.2.1	Estruturas Geral de um SIG	18
2.2.2	Gerencia de Dados em um SIG	19
2.3	Web Map Service & Web Feature Service	22
3	BANCO DE DADOS GEOGRÁFICOS	24
4	CONSULTAS ESPACIAIS	26
4.1	Métodos de Acesso para Dados Espaciais	26
4.2	Tipos de Consultas Espaciais	28
4.2.1	Exact Match Query	28
4.2.2	Partial Match Query & Point Query	28
4.2.3	Containment Ranger Query	29
4.2.3	Enclosure Region Query	30
4.2.4	Adjacency Query	30
4.2.5	Nearest Neighbor Query	30
4.2.6	Direction Query	31
5	ALGORITMOS DE BUSCA APLICADOS A DADOS ESPACIAIS	32
5.1	R-Tree	32
5.2	Gist	34
5.3	B-Tree	35
6	CONSULTAS ESPACIAIS, UTILIZANDO ÍNDICES	37
6.1	PostGreSQL	37
6.2	Gerenciador de Banco de Dados PgAdmin III	38
6.3	Introdução as Consultas realizadas com os Índices	40
6.4	Resultados obtidos	46
7	CONCLUSÃO	48

1. INTRODUÇÃO

O Geoprocessamento, tecnologia aplicada no SIG, pode ser definido como o conjunto de técnicas e metodologias que implicam na aquisição, arquivamento, processamento e representação de dados geo-referenciados (MARBLE & PEUQUET, 1983).

Compreender a distribuição espacial de fenômenos constitui hoje um desafio para a elucidação de questões importantes em diversas áreas do conhecimento, como saúde, ambiente, geologia, agronomia, entre outras. Tais estudos vêm-se tornando cada vez mais comuns, devido à disponibilidade de sistemas de informação geográfica (SIG) de baixo custo e com interfaces amigáveis. Esses sistemas permitem a apresentação espacial de variáveis como população de indivíduos, índices de qualidade de vida ou venda de empresas em uma região por meio de mapas. Para tanto, basta dispor de um banco de dados e de uma base geográfica (como um mapa de municípios), e o **SIG** torna possível apresentar um mapa que permite a visualização do padrão espacial determinado pelo fenômeno em observação (CAMARA et. al, 2004).

Ao se observar o armazenamento, análise e apresentação de dados sobre o determinado espaço geográfico, fez com que se desenvolvessem ambientes informatizados que aliassem mapas digitais as informações sobre os elementos do mapa. Esta operação envolve tecnologia de informática, banco de dados e cartografia digital. As aplicações e usos do SIG dependem da existência de um sistema eficiente e lógico que possa transformar e associar elementos cartográficos a banco de dados (MARBLE & PEUQUET, 1983).

Os sistemas gerenciadores de dados geográficos possuem a capacidade de efetuar verificações denominadas consultas espaciais, as quais permitem a realização de cálculos como área, distância, centróide (ponto no interior de uma forma geométrica que define o seu centro geométrico), e algumas outras operações, e será dando ênfase a este ponto, que esse estudo foi realizado.

1.1. OBJETIVOS

1.1.1. Objetivo Geral

Desenvolver estudos sobre banco de dados geográfico buscando avaliar o desempenho de índices utilizados em consultas espaciais.

1.1.2. Objetivos Específicos

- Realizar um estudo sobre as extensões para armazenamento de dados geográficos e formas de representação;
- Realizar um estudo sobre as consultas espaciais disponíveis;
- Estudar diversos algoritmos de busca, aplicados a dados espaciais;
- Utilizando um banco de dados com informações reais. Realizar consultas espaciais aplicando índices;
- Avaliar o desempenho dos índices espaciais utilizados, indicando-os para tipos de dados específicos.

2. GEOPROCESSAMENTO

Geoprocessamento é o conjunto de técnicas de manipulação de bases analógicas e digitais para a elaboração de mapas-base digitais em programas especialistas. Os mapas digitais podem assumir duas "entidades" primordiais: a grade e o vetor. A grade (raster) é formada por pixels (menor elemento da imagem), sendo a cada um deles atribuído um valor (atributo). A localização de cada pixel e suas características intrínsecas dá origem a uma matriz. O vetor é uma informação georreferenciada, que pode assumir a forma de pontos, linhas ou polígonos, a cada uma destas formas é também dado um atributo (DIAS & YWASAKI, 2007). A Figura 1 mostra a representação de um mapa de grade e um mapa vetorial, a partir de uma localidade hipotética.

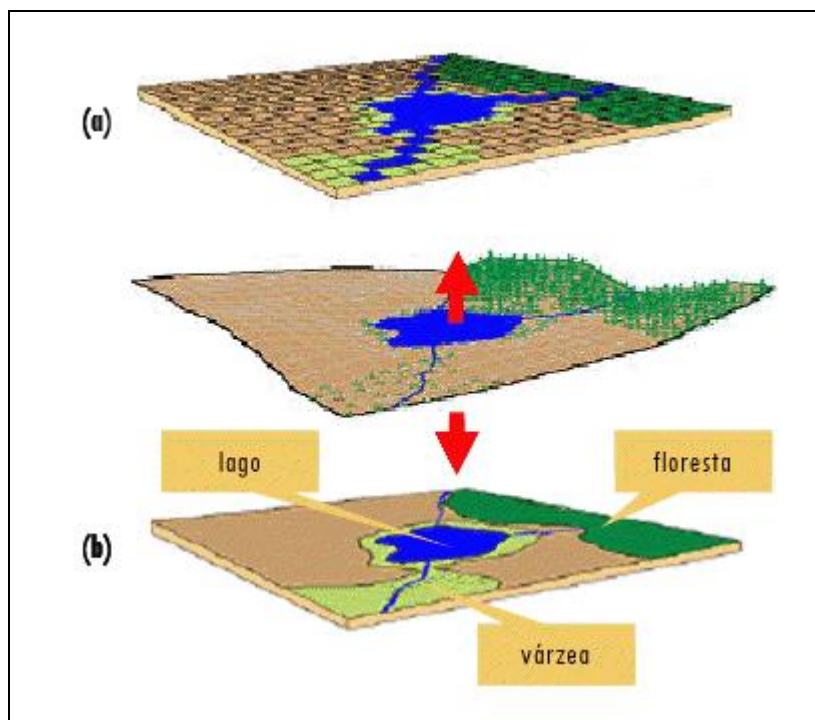


Figura 1 - Representação de um mapa de grade (a) e um mapa vetorial (b)
Fonte: USP - Universidade de São Paulo.

Nesse contexto, o termo Geoprocessamento pode ser descrito como a disciplina do conhecimento que utiliza técnicas matemáticas e computacionais para o tratamento da informação geográfica e que vem influenciando de maneira crescente as áreas de cartografia, análise de recursos naturais, transportes, comunicação, energia e

planejamento urbano e regional. As ferramentas computacionais para geoprocessamento, chamadas de sistemas de informação (SIG), permitem realizar análises complexas, ao integrar dados de diversas fontes e ao criar banco de dados georeferenciados. Tornam ainda possível automatizar a produção de materiais cartográficos (CAMARA & DAVIS, 2005).

Já há algum tempo o termo geoprocessamento tem se tornado muito útil para varias empresas que tem adotado essa tecnologia como ferramenta de trabalho. Segundo Davis e Câmara, de forma genérica, "*Se **onde** é importante para seu negocio, então Geoprocessamento é a sua ferramenta de trabalho*", sempre que o **onde** aparecer, dentre as questões e problemas que precisam ser resolvidos por um sistema informatizado, existe a possibilidade de se usar das ferramentas que estão disponíveis dentro de um sistema de informação geográficos (CAMARA & DAVIS, 2005).

Segundo Rodrigues (1993), Geoprocessamento é um conjunto de tecnologias de coleta, tratamento, manipulação e apresentação de informações espaciais voltado para um objetivo específico. Este Conjunto possui como principal ferramenta o *GIS*.

O Brasil tem carecido nesse aspecto, quando se trata de obter informações com as quais se julga adequadas para a tomada de decisões sobre os diversos problemas ambientais e urbanos. O geoprocessamento pode trazer a solução adequada, pois tem capacidade para suprir essa carência além de ser uma tecnologia que não tem um custo tão alto.

2.1 História do Geoprocessamento

A história do geoprocessamento se iniciou com os EUA e a Inglaterra na década de 50 com o intuito de otimizar a produção e manutenção de mapas. Entretanto, devido ao fato da informática estar ainda pouco desenvolvida, a atividade era muito cara e restrita e ainda nem existia o conceito de GIS que só viria a ser empregado na década de 70 (FARIA, 2008).

Os primeiros Sistemas de Informação Geográfica surgiram na década de 60, no Canadá, como parte de um programa governamental para criar um inventário de recursos naturais. Estes sistemas, no entanto, eram muito difíceis de usar: não existiam monitores gráficos de alta resolução, os computadores necessários eram excessivamente caros, e a mão de obra tinha que ser altamente especializada e caríssima. Não existiam soluções comerciais prontas para uso, e cada interessado precisava desenvolver seus

próprios programas, o que demandava muito tempo e, naturalmente, muito dinheiro (CAMARA, 2006).

Contudo, á partir da década de 80, concomitantemente ao desenvolvimento da tecnologia dos computadores e softwares, o geoprocessamento deu um salto, principalmente após a fundação da NCGIA (National Centre for Geographical Information and Analysis), em 1989, quando o geoprocessamento passou a ser reconhecido oficialmente como uma disciplina científica (CAMARA, 2006).

Nos últimos anos temos presenciado a massificação do geoprocessamento. Com o lançamento de ferramentas como o Google Earth, qualquer pessoa mesmo que não entenda nada de geoprocessamento pode ter acesso a mapas de qualquer região do mundo que aliam imagens de satélite, GPS e modelos em 3D.

Atualmente, o geoprocessamento consiste nas seguintes etapas: coleta, armazenamento, tratamento e análise de dados e uso integrado das informações. Uma ferramenta de geoprocessamento muito conhecida em meio a todos (pelo menos por aqueles que possuem habilitação) é o GPS, que é uma ferramenta básica, muito utilizada para coleta de dados georreferenciados (FARIA, 2008).

O geoprocessamento no Brasil nos dias atuais pode ser dividido em quatro segmentos, são eles: Cadastral, Cartografia, Automatizada, Ambiental e Concessionárias/Redes (CAMARA, 1996).

Cadastral são aplicações de cadastro urbano e rural, as quais são normalmente realizadas por prefeituras, para atender a este setor os SIGs usam funções de consulta a banco de dados relacionados e suas apresentações gráficas. A parte de Cartografia automatizada, é aquela que realiza o mapeamento básico e temático, neste caso é necessário que o SIG seja equipado com a técnica de aerofotogrametria tradicional e moderna e dispor de ferramentas sofisticadas de entrada de dados e de produção de mapas (CAMARA, 1996).

A parte ambiental são as instituições ligadas às áreas de agricultura, meio-ambiente, ecologia e planejamento regional, para os SIGs atenderem a este segmento é necessário a integração de dados (mapas temáticos, modelos de terreno e imagens de sensoriamento remoto), gerenciamento e conversão entre projeções cartográficas. As concessionárias de serviços (água, Telefonia, Energia elétrica) Cada aplicação de rede tem características próprias e com alta dependência de cada usuário. Os SIG's para redes devem apresentar duas características básicas: a forte ligação com bancos de dados relacionais e a capacidade de adaptação e personalização (CAMARA, 1996).

2.2 SIG - SISTEMA DE INFORMAÇÃO GEOGRÁFICA

O termo Sistemas de Informação Geográfica (SIG) é aplicado para sistemas que realizam o tratamento computacional de dados geográficos e recuperam informações não apenas com base em suas características alfanuméricas, mas também através de sua localização espacial; oferecem ao administrador (urbanista, planejador, engenheiro) uma visão inédita de seu ambiente de trabalho, em que todas as informações disponíveis sobre um determinado assunto estão ao seu alcance, interrelacionadas com base no que lhes é fundamentalmente comum – a localização geográfica. Para que isto seja possível, a geometria e os atributos dos dados num SIG devem estar localizados na superfície terrestre e representados numa projeção cartográfica (CAMARA, 2006). Abaixo a Figura 2, demonstra a área de atuação dos SIGs.



Figura 2 - Sistema de Informação Geográfica
FONTE: Alessio, A. Banco de dados Espaciais

Segundo CAMARA, 2006, o requisito de armazenar a geometria dos objetos geográficos e de seus atributos representa uma dualidade básica para SIG's. Para cada objeto geográfico, o SIG necessita armazenar seus atributos e as várias representações gráficas associadas. Devido a sua ampla gama de aplicações, que inclui temas como agricultura, floresta, cartografia, cadastro urbano e redes de concessionárias (água, energia e telefonia), há pelo menos três grandes maneiras de utilizar um SIG:

- Como ferramenta para produção de mapas;
- Como banco de dados geográficos, com funções de armazenamento e recuperação de informações espaciais;
- Como ferramenta para produção de mapas;

Estas três visões do SIG são antes convergentes que conflitantes e refletem a importância relativa do tratamento da informação geográfica dentro de uma instituição.

As principais características de um SIG são:

- Inserir e integrar, numa única base de dados, informações espaciais provenientes de meio físico-biótico, de dados censitários, de cadastros urbanos e rurais, e outras fontes de dados como imagens de satélite, e *GPS* (CÂMARA, 2005).
- Oferecer mecanismos para combinar as várias informações, através de algoritmos de manipulação e análise, bem como para consultar, recuperar e visualizar o conteúdo da base de dados geográficos (CÂMARA, 2005).

2.2.1 Estruturas Geral de um SIG

Numa visão abrangente, pode-se indicar que um SIG tem os seguintes componentes:

- Interface com usuário;
- Entrada e integração de dados;
- Funções de consulta e análise espacial;
- Visualização e plotagem;
- Armazenamento e recuperação de dados (organizados sob a forma de um banco de dados geográfico).

Estes componentes se relacionam de forma hierárquica. No nível mais próximo ao usuário, a interface homem-máquina define como o sistema é operado e controlado. No nível intermediário, um SIG deve ter mecanismos de processamento de dados espaciais (entrada, edição, análise, visualização e saída). No nível mais interno do sistema, um sistema de gerência de bancos de dados geográficos oferece armazenamento e recuperação dos dados espaciais e seus atributos (CAMARA, 1999).

De uma forma geral, as funções de processamento de um SIG operam sobre dados em uma área de trabalho em memória principal. A ligação entre os dados geográficos e as funções de processamento do SIG é feita por mecanismos de seleção e consulta que definem restrições sobre o conjunto de dados (CAMARA, 1999). A Figura 3 explica a arquitetura de um SIG.

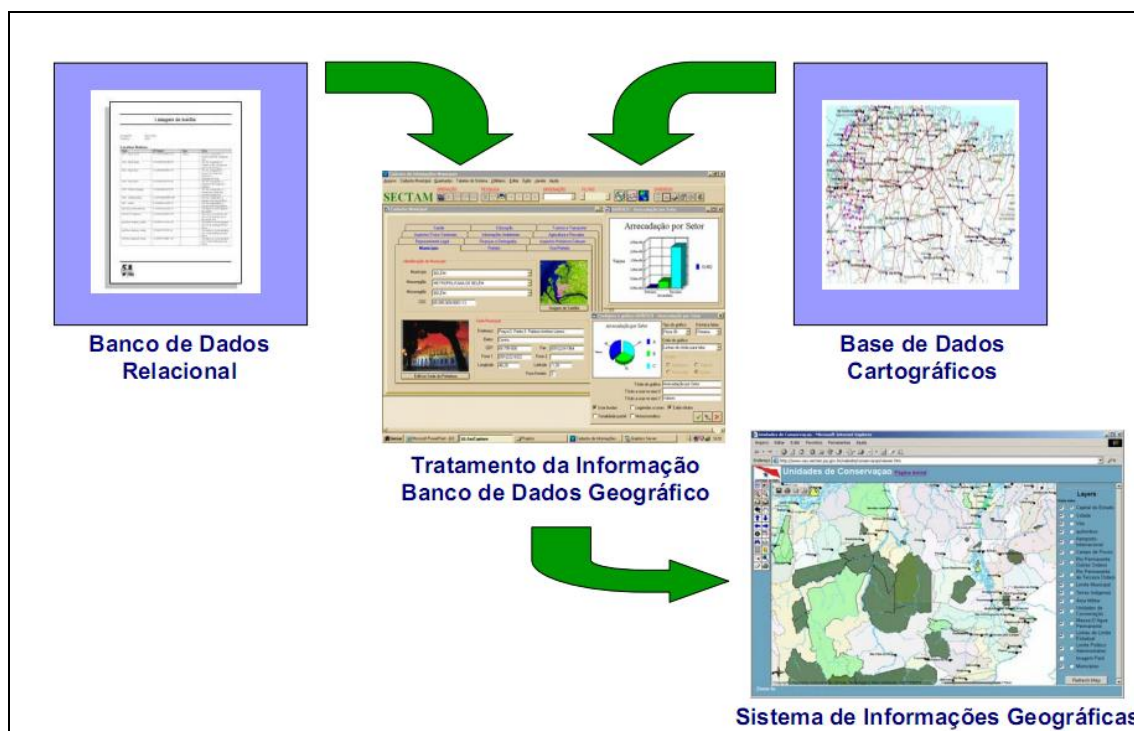


Figura 3 - Arquitetura de um SIG
Fonte: Arquitetura de um SIG, Gilberto Câmara, 2002.

2.2.2 Gerência de Dados em um SIG

Atualmente, a principal diferença entre os SIGs é a forma como os dados geográficos são gerenciados. Há basicamente três diferentes arquiteturas de SIGs que utilizam os recursos de um SGBD: dual, integrada baseada em SGBDs relacionais e integrada baseada em extensões espaciais sobre SGBDs objeto-relacionais (CÂMARA, 2006).

Fazendo-se o uso de um SGDB, os atributos dos objetos geográficos são armazenados, e as formas geométricas dos objetos são salvas dentro de arquivos. A arquitetura Dual usa o modelo relacional para fazer o armazenamento de dados dentro das tabelas da seguinte forma: dentro das colunas são denominados os atributos e nas suas linhas os dados, os atributos recebem identificadores únicos para referenciar os

arquivos geométricos, após isso, uma ligação é entre os atributos não-espaciais que estão salvos no SGBD e o arquivo que contém a geometria.

A principal vantagem de usar esta estratégia é poder utilizar dos SGBDs do mercado, porém deve ser observado que essas representações geométricas dos objetos espaciais não estão sob o controle do SGBD, desta forma dificulta-se o equacionamento das questões de otimização de consultas, gerencia de transações e controle de integridade. Abaixo a Figura 4, demonstra a arquitetura dual:

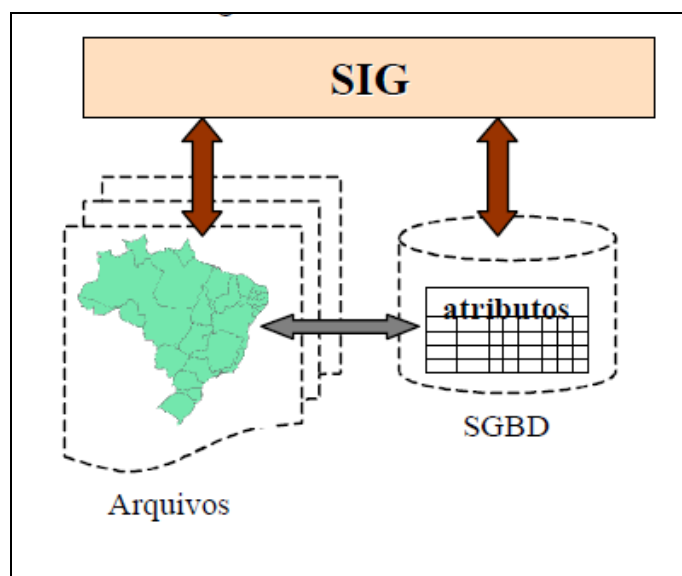


Figura 4 - Arquitetura Dual

Fonte: **Arquitetura de Sistemas de Informação Geográficas. Gilberto Câmara.**

Já as principais desvantagens, ocorrem no momento de controlar e manipular os dados espaciais, manter a integridade entre o componente espacial e o componente alfanumérico. As consultas realizadas são lentas, pois a consulta é processada de forma separada. A parte espacial é processada pelo aplicativo utilizando os arquivos proprietários, e a parte convencional da consulta é executada pelo SGBD de forma separada. Outra desvantagem é a falta de interoperabilidade entre os dados, sendo que cada sistema produz seu próprio arquivo proprietário, desta forma não há um formato padrão a ser seguido, dificultando a integração dos dados.

Já na Arquitetura integrada, os dados alfanuméricos e dados espaciais são armazenados dentro de um SGBD, desta forma pode utilizar-se dos recursos que ele oferece os quais são gerencia de transações, controle de integridade do banco e concorrência em dados espaciais, assim a integridade entre os componentes espaciais e

alfanuméricos é feita pelo SGBD. A Figura 5, demonstra um modelo da arquitetura integrada.



Figura 5 - Arquitetura Integrada

Fonte: Arquitetura de Sistemas de Informação Geográficas. Gilberto Câmara.

Além desses benefícios, com o armazenamento dos dados em uma mesma base a pesquisa fica simplificada, já que apenas é necessária realizar uma única consulta á base de dados para retorno dos dados desejados, ao contrário disso na arquitetura Dual para se retornar a localização de um objeto e sua descrição era necessário acessar a base espacial e convencional (JUNIOR A, 2011). Para melhor compreensão, a Figura 6 demonstra uma comparação de ambas arquiteturas.

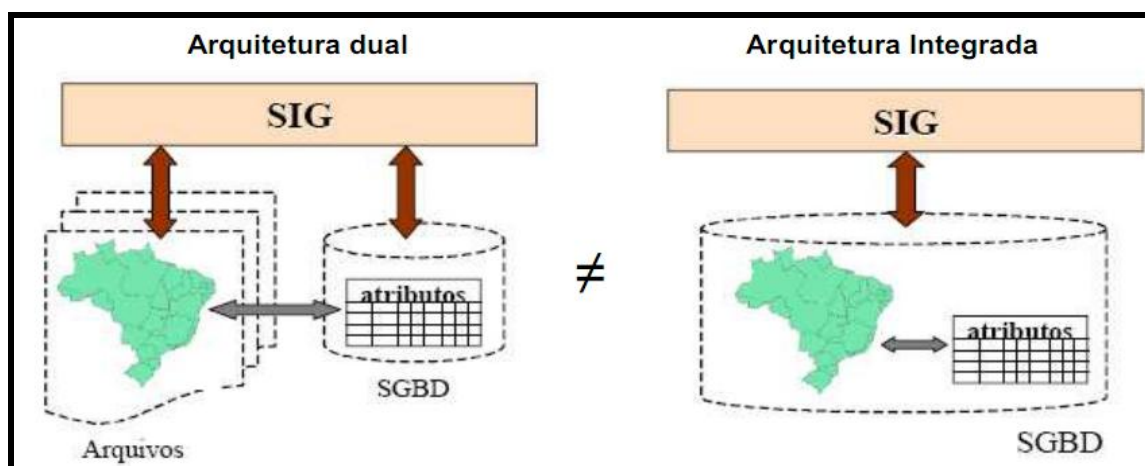


Figura 6 - Diferença da Arquitetura Dual para a Arquitetura Integrada

FONTE: Universidade do Estado do Mato Grosso – Departamento de Engenharia Civil

A arquitetura integrada se divide em dois modelos, o baseado em SGBD's relacionais e o baseado em extensões espaciais sobre SGBD's objeto-relacional.

O modelo baseado em um SGBD relacional utiliza os campos BLOB¹ (*Binary Large Object*) para realizar o armazenamento do dado espacial, porém isso gera alguns inconvenientes já que o SGBD trata os dados espaciais como uma cadeia binária, não é possível conhecer a semântica do seu conteúdo, os métodos de acesso espacial e otimizador de consultas devem ser implementados pelo SIG já que os dados são tratados como uma cadeia binária além das limitações da linguagem SQL para a manipulação dos dados espaciais. (CÂMARA, 2006).

Já o modelo baseado em extensões espaciais sobre SGBDOR funciona acoplando extensões espaciais especialmente desenvolvidas para estes SGBDOR. Estas extensões ampliam o banco de dados adicionando novas funcionalidades e procedimentos que permitem armazenar, acessar e tratar dados espaciais de formato vetorial. As desvantagens deste modelo estão na falta de mecanismos de controle de integridade sobre os dados espaciais e a falta de padronização das extensões da linguagem SQL. (CÂMARA, 2006).

2.3 Web Map Service & Web Feature Service

A especificação WMS é um *standard* do OGC que normaliza a forma como os clientes podem requisitar mapas a servidores de mapas e também o modo como estes servidores devem descrever e devolver mapas (BEAUJARDIERE, 2006).

A especificação WMS define a forma de criação e visualização de mapas georreferenciados a partir de diversas fontes de dados distribuídas e heterogêneas. Os mapas são uma representação visual e bidimensional da informação em formato *raster*. Existem 3 operações definidas para o WMS que são:

- **GetCapabilities**, que se destina a disponibilizar para a aplicação cliente informação sobre os serviços que disponibiliza. Estas informações correspondem aos metadados do serviço e é fornecida no formato XML.
 - **GetMap**, responsável por produzir o mapa, de acordo com os parâmetros especificados e no formato requerido pela aplicação cliente.
-

- *GetFeatureInfo* (opcional), que devolve informação sobre as entidades geográficas apresentadas pelo mapa (NEBERT, 2004).

Para melhor compreensão imagine o seguinte cenário, "você possui um grande banco de dados geográfico com dados de desmatamento (polígonos), e gostaria de permitir o acesso a esses dados a outros pesquisadores, que podem por exemplo, estudar padrões de desmatamento" (QUEIROZ, 2007).

Disponibilizando um servidor WFS como mediador para este banco de dados seria possível que os pesquisadores tivessem acesso aos dados, independente do formato de armazenamento do banco e com um protocolo bem definido (QUEIROZ, 2007).

- Utilizando dos serviços WFS, os usuários podem enviar requisições via HTTP GET ou POST;
- Os parâmetros de uma requisição são expressos em termos de pares (nome/valor) ou XML;
- No caso de requisições codificadas em XML, não se usa o método GET (usa-se apenas POST).
- Além disso, faz-se intensivo uso de XML nas requisições.

2.4 BANCO DE DADOS GEOGRÁFICOS

Os conceitos supracitados expressam a idéia básica de BD, que é uma estrutura onde dados podem ser armazenados e relacionados, de tal modo que seja possível resgatá-los sem qualquer perda ou mesmo alteração do seu valor (LIMA A. 2011).

Representar ações, estados ou mesmo objetos fazendo uso apenas de simbologia às vezes pode-se tornar uma tarefa bastante árdua. Matemática, Física e outras tantas ciências sempre buscam exprimir conceitos/idéias através de símbolos e expressões, sendo que muitas delas possuem um significado compreensível por serem representadas em uma linguagem próxima à nossa, a dos seres humanos. Em computação, o armazenamento dessas idéias não é tão imediato assim. Dados inseridos em um computador são gravados como 0's e 1's, de forma que a leitura direta da informação não traz consigo qualquer contexto semântico. Surge então a necessidade de uma diferenciação entre o que é dado e informação: o primeiro refere-se a uma simbologia que pode ser lida ou armazenada em algum local, enquanto a outra expressa um conceito, atribui uma semântica a um determinado símbolo. Por exemplo, dado o valor "13", nada se pode concluir porque ele não agrega nenhuma informação, é meramente uma simbologia. O conceito semântico surge quando se atribui um significado ao valor (LIMA A. 2011).

Entre os diversos tipos de bancos de dados que são utilizados atualmente, o Modelo Relacional tem se destacado por ter sua estrutura baseada em tabelas, compostas por linhas e colunas. Cada linha de registro esta ligada a um campo de atributos o qual dá valor a propriedade deste conceito, certos registros fazem referência a outros registros que por sua vez estão diretamente referenciando outro registro, todo este processo faz parte da caracterização do modelo adotado pelo Banco de Dados.

A Figura 7 mostra como é realizado o acesso a um banco de dados.



Figura 7 - Representação do acesso a um Banco de Dados
Fonte: Poucodetudo.com, Os Melhores Programas Banco de Dados.

Um banco de dados geográfico tem sua diferença do convencional por armazenar, além de dados alfanuméricos, os dados referentes à localização e geometria de uma entidade, por exemplo, se for introduzido na base de dados uma feição geográfica, este não guardará apenas as descrições desta feição, como nome, descrição, guardará também sua posição (x, y) em um sistema de coordenadas e sua feição geométrica. Além da forma de armazenamento, outra diferença é a possibilidade de aplicar funções desenvolvidas especialmente para tratamento de dados geográficos, das quais, por exemplo, é possível extrair em uma consulta a distância entre dois pontos (IRRIGART, 2006).

Como vantagens deste tipo de armazenamento podem se citar o fato de, evitar os problemas de controle de integridade típicos de ambiente *desktop*, permitindo o acesso concorrente dos dados, essa facilidade de se integrar as bases de dados corporativas já existentes, como sistemas legados que já utilizam de SGBD relacionais, sendo este último fator possível graças às características de um SGBD, que apresenta os dados em uma visão independente dos aplicativos, além de garantir três importantes requisitos que são o acesso e modificações de grandes volumes de dados, o controle de acesso por múltiplos usuários e a manutenção de dados por longo tempo independente dos aplicativos que acessam (CÂMARA, 2004).

3 CONSULTAS ESPACIAIS

Um dos principais componentes de todo sistema de gerência de bancos de dados, o processador de consultas, recebe uma consulta e prepara um plano de execução, consistindo de operações de mais baixo nível, implementadas pelo subsistema de armazenamento. A parte do plano responsável pelo processamento dos objetos geográficos tipicamente começa na fase de filtragem, que identifica quais objetos podem satisfazer a qualificação da consulta, esta fase pode utilizar índices espaciais, em geral definidos sobre aproximações das geometrias dos objetos. A fase de refinamento do plano recupera para memória principal a geometria exata dos objetos identificados na fase anterior, computando quais deles de fato satisfazem a qualificação da consulta. A fase final do plano aplica transformações aos objetos retornados na fase anterior, produzindo a resposta final da consulta (CASANOVA, 2005).

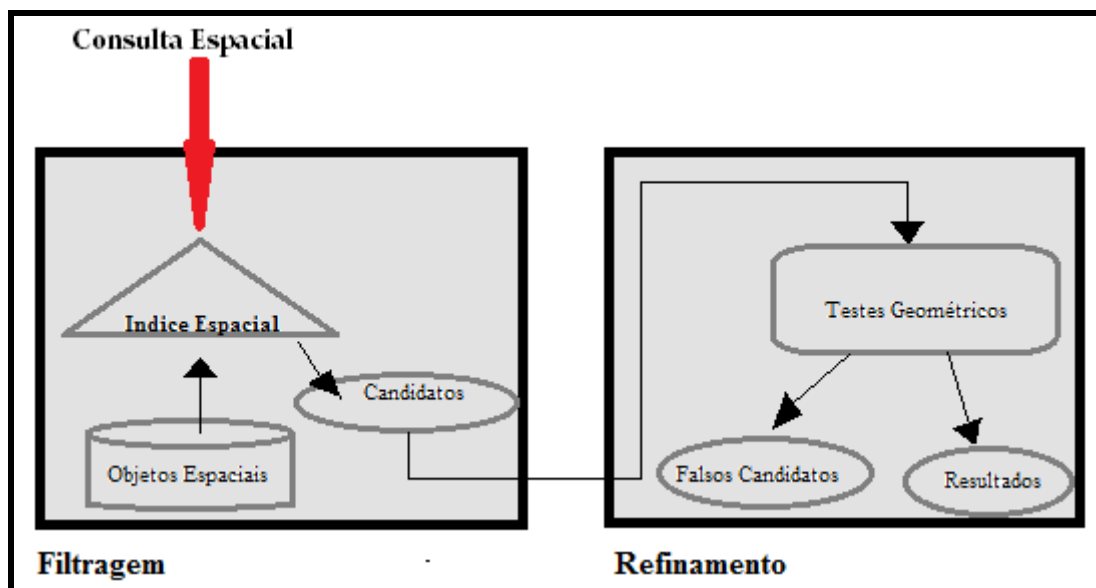


Figura 8 - Processamento de consultas espaciais

Fonte: Banco de Dados Geográficos. Adaptada de Gaede e Gunther (1998).

3.1 Métodos de Acesso para Dados Espaciais

Métodos de acesso espacial, ou índices espaciais, são as estruturas de dados auxiliares, mais essenciais para o processamento eficiente de consultas espaciais, certamente uma consulta espacial normalmente envolve apenas uma parte do banco de dados, sendo assim não se faz necessário percorrer todo o banco de dados, portanto é necessário que se estabeleça um plano de execução para que essa consulta seja efetuada

com sucesso. Esta fase depende da existência de índices espaciais, em geral definidos sobre aproximações das geometrias dos objetos (QUEIROZ, 2005).

Um método de acesso espacial (MAE) é uma estrutura de indexação voltada ao suporte de objetos espaciais. O principal objetivo de um MAE é propiciar uma rápida obtenção dos objetos espaciais que satisfazem um determinado relacionamento topológico, métrico ou direcional. Neste sentido, o espaço indexado é organizado de tal forma que, por exemplo, a recuperação dos objetos espaciais contidos em uma área particular requeira apenas o acesso aos objetos próximos a esta área, em oposição à análise do conjunto completo de objetos armazenados em memória secundária. Pode-se dizer, portanto, que um MAE é projetado como um caminho otimizado aos dados e o seu uso melhora significativamente o desempenho de sistemas gerenciadores de banco de dados (SGBD) espaciais no processamento de consultas (CIFERRI, 2001).

A seguir são descritos os principais tipos de dados espaciais oferecidos por SGBD's espaciais. Um ponto é a menor unidade possível para representar um objeto espacial. Um objeto espacial formado por um único ponto não possui extensão. Uma linha é uma sequência de pontos conectados retilinearmente, ao passo que uma linha poligonal os pontos não estão dispostos de forma retilínea. Para ambas, linhas e linhas poligonais, cada par de pontos conectados corresponde a um segmento de linha. Um polígono é formado por uma sequência de linhas ou de linhas poligonais, sendo que esta sequência é fechada, isto é, o primeiro ponto coincide com o último. Polígonos e linhas poligonais fechadas são funcionalmente equivalentes, diferindo somente em alguns SGBD espaciais com relação ao armazenamento explícito de atributos convencionais derivados de polígonos, como exemplo área e perímetro (CIFERRI, 2001).

De uma forma geral, os pontos são utilizados para apresentar localizações discretas, como uma cidade em um mapa, linhas e linhas poligonais abertas são usadas para representar objetos espaciais lineares (rios, estradas, ferrovias). Polígonos e linhas poligonais fechadas são usados para representar objetos bidimensionais, como exemplo a área ocupada por um bairro. Também existem os poliedros que são utilizados para representar sólidos no espaço (peça mecânica ou um membro humano). A Figura 9 ilustra graficamente os vários tipos de dados espaciais.

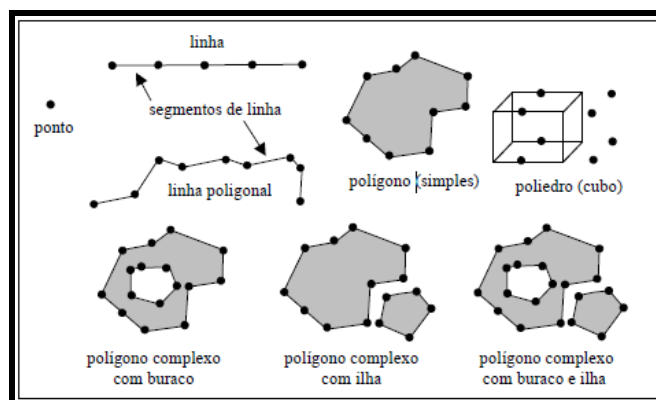


Figura 9 - Tipos de dados espaciais
FONTE: Ricardo R. Ciferri, 2001.

3.2 Tipos de Consultas Espaciais

3.2.1 Exact Match Query

A *Exact Match Query* tem o seguinte objetivo: Quando se recebe dados de um objeto o qual possui atributos espaciais, ela encontrar todos os objetos que tem as mesmas coordenadas de tal objeto, sendo eles de mesma extensão e localização no espaço.

A Execução deste tipo de consulta espacial pode ser realizada em três situações distintas. Consideremos objetos representados por pontos, tais como postes, poços artesianos e edificações de destaque. A primeira situação consiste da busca de um ponto inexistente no arquivo de dados. Neste caso a consulta é capaz de descobrir tal inexistência sem ter que percorrer parte significativa de sua estrutura de dados. Já a segunda situação acontece quando o arquivo de dados não contém duplicatas e a pesquisa é efetuada para um ponto existente. Como resultado é recuperado um único ponto dentre o universo de pontos armazenados no arquivo de dados. A última situação é uma variante da anterior, quando o arquivo de dados contém duplicatas. Nesta situação, podem ser recuperados vários objetos. O armazenamento em separado de duplicatas conduz a um maior número de acessos a disco e, por conseguinte a uma queda no desempenho (TSURUDA, 2009).

3.2.2 Partial Match Query

Para objetos bidimensionais, tal como polígonos, esta consulta espacial determina uma faixa perpendicular com relação à dimensão escolhida, segundo a qual

os objetos devem ter as suas respectivas extensões e localizações limitadas. Para pontos, esta faixa torna uma reta, como demonstrado na Figura 10 (CIAFERRI, 2001).

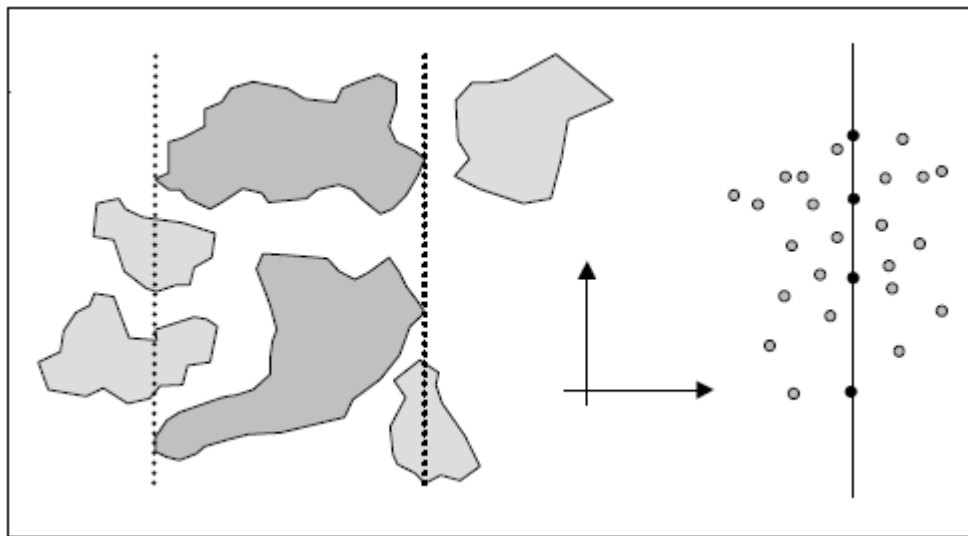


Figura 10 - Exemplo de partial match queries.
FONTE: Banco de Dados Espaciais. Ricardo R. Ciferri, 2001

3.2.3 Containment Ranger Query

Para arquivos de dados do tipo ponto, esta consulta é utilizada para encontrar todos os objetos que tenham pelo menos um ponto em comum. A **Figura 11** demonstra uma consulta considerando como região de pesquisada o bairro Ipiranga, observe que o foco de dengue neste bairro são raros, ao contrário do que ocorre no bairro adjacente de São Luís (CIAFERRI, 2001).

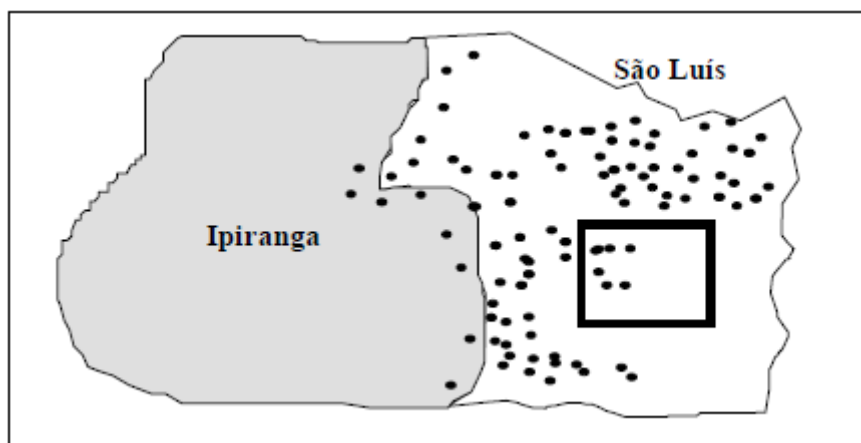


Figura 11 - Exemplo de Container Ranger Query
FONTE: Métodos de Acesso Multidimensionais. Ricardo Ciferri,

3.2.3 Enclosure Region Query

Também conhecida como *enclosure query* a *enclosure region query* é utilizada para encontrar todos os pontos que englobam um ponto a ser pesquisado, por exemplo, dado um ponto o , com um atributo espacial definido, é possível encontrar todos os objetos que englobam o (CIAFERRI, 2001).

3.2.4 Adjacenty Query

A partir de um dado pesquisado, é possível encontrar todos os objetos que são adjacentes a ele. Como por exemplo, pode-se querer identificar todos os estados que tenham fronteira com o estado de Pernambuco como mostra a FIGURA 12 (TSURUDA, 2009).



Figura 12 - Exemplo de Adjacenty query

FONTE: Ricardo Ciferri, Métodos de Acesso Multidimensionais

3.2.5 Nearest Neighbor Query

Este tipo de consulta envolve um cálculo de relacionamento de distância entre os objetos espaciais, retornando o elemento (ou os elementos) mais próximo (MANICA, 2007).

Há basicamente dois subtipos de consultas. O primeiro é baseado no cálculo da menor distância, enquanto o segundo é baseado no cálculo das menores distâncias (CIAFERRI, 2001).

3.2.6 Direction Query

Dado um objeto espacial o' com atributo espacial $o'.G \subseteq Ed$, encontre todos os objetos o posicionados na direção s com relação ao objeto o' (TSURUDA, 2009).

Um exemplo é ilustrado na FIGURA 13, na qual são determinadas todas as propriedades localizadas completamente ao sul da linha do Equador. Para o espaço Euclidiano tridimensional, também são válidos os relacionamentos direcionais de trás de, na frente de, acima, abaixo, além de outros.

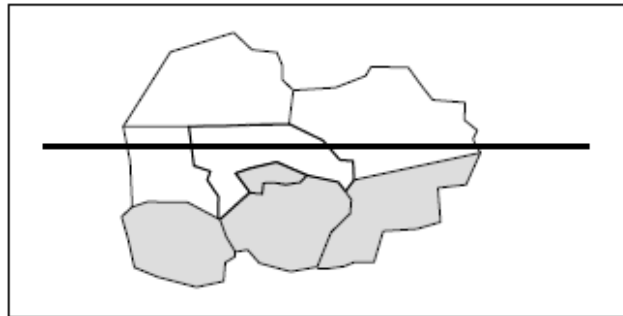


Figura 13 - Exemplo de Direction Query

FONTE: Ricardo Ciferri, Métodos de Acesso Multidimensionais

4 ALGORITMOS DE BUSCA APLICADOS A DADOS ESPACIAIS

4.1 Índice Espacial, R-Tree

A R-Tree é uma árvore balanceada por altura, semelhante à B+-Tree [Comer 1979], com ponteiros para objetos espaciais nos nós folhas. É uma estrutura de dados hierárquica que utiliza retângulos (chamados de MBRs - *Minimum Bounding Rectangle*) para organizar um conjunto dinâmico de objetos espaciais, de maneira que objetos co-localizados fiquem armazenados próximos uns dos outros e que haja uma redução no espaço de busca a cada nível da árvore (GUTTMAN, 1984).

As regras básicas para formação de uma árvore-R são semelhantes às da árvore-B. Todas as folhas aparecem sempre no mesmo nível da árvore. Nós internos contêm a delimitação de retângulos que englobam todos os retângulos dos nós nos níveis inferiores (FIGURA 16). Uma árvore-R de ordem (m, M) conterá entre m e M entradas em cada nó da árvore ($m < M / 2$), exceto a raiz, que terá pelo menos dois nós (a menos que a árvore só tenha uma entrada) (DAVIS Jr, 2005).

Conforme ilustrado na Figura 14, a estrutura hierárquica da *R-Tree* possui um nó raiz, nós internos (N1..2 _ N3..6) e um último nível de nós folha (N3..6 _ a..i). A parte superior da figura retrata o desenho dos MBRs agrupando os objetos espaciais de a a i em subconjuntos, de acordo com sua co-localização. A parte inferior ilustra a representação da árvore *R-Tree*. Cada nó armazena no máximo M e no mínimo m _ M 2 entradas. O valor de M é um fator importante na organização de um índice *RTree*. A escolha deste valor pode gerar mais espaço morto e áreas sobrepostas entre os MBRs, o que degrada o desempenho da operação de busca na árvore (OLIVEIRA, 2011).

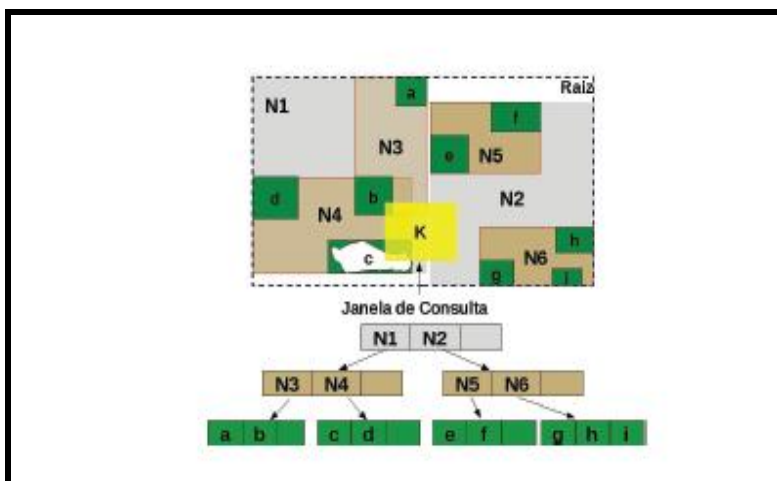


Figura 14 - Organização de uma R-Tree

FONTE: Um Índice R-Tree Escalável Distribuído. OLIVEIRA, T. B.

Um problema com as árvores-R é que a ordem de inserção dos objetos interfere na forma final da árvore, e portanto vai interferir também com o resultado das operações de subdivisão dos nós para manter o balanceamento. Existem algumas técnicas para tentar otimizar este comportamento da árvore-R, mas sempre com algum custo adicional em termos de processamento (QUEIROZ, 2005).

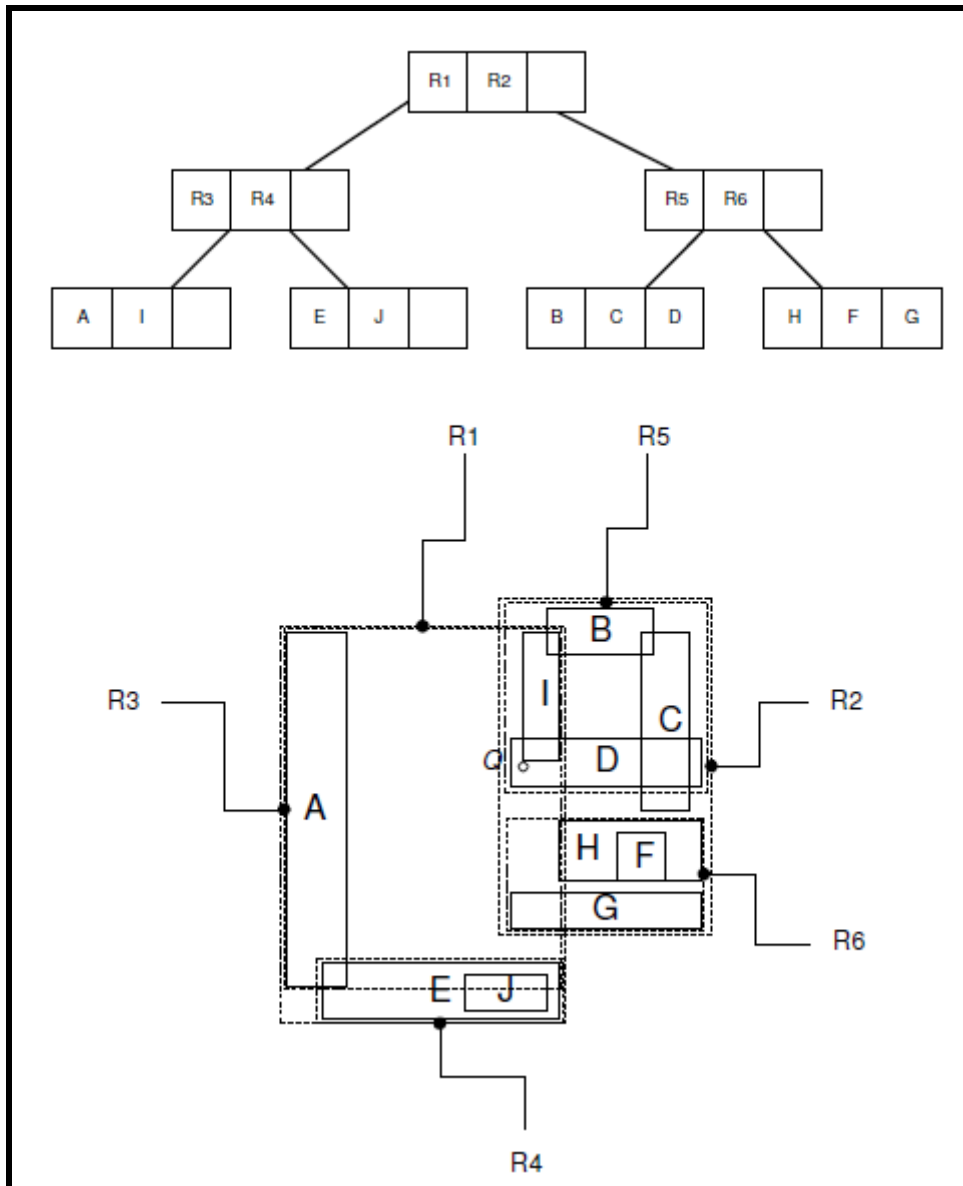


Figura 15 - Arvore R e retângulos correspondentes.

FONTE: Banco de dados Geográfico. Casa Nova, M. Câmara, G. Davis Jr, C. Vinhas, L. Queiroz, G.

Pesquisas na árvore-R são relativamente simples de serem executadas. O único problema é que um grande número de nós pode ter que ser examinado, pois um retângulo pode estar contido em vários outros, mesmo que o objeto esteja contido em apenas um nó folha (QUEIROZ, 2005).

4.2 Índice Espacial, GIST

GIST representa “Generalized Search Tree” e é uma forma generalizada de indexar. Além da indexação de GIS, GIST é usada para acelerar buscas em todos os tipos de estruturas de dados irregulares (arranjos de números inteiros, dados espectrais, etc) que não são amenos a indexação normal B-Tree (SANTILLI, 2010).

Oferecem uma infra-estrutura sobre a qual podem ser implementadas consultas, gera uma árvore de pesquisa balanceada versátil similar a uma árvore B que pode ser utilizada com estruturas de dados definidas pelo usuário e em implementações diversas, como índices para cubos Olap, árvores R (R-Trees) e em busca em texto. Internamente é composta por nós que apresentam pares (Valor, ponteiro para dado indexado ou para outro nó).

Os índices GIST não são um único tipo de índice, mas na verdade uma infra-estrutura nas quais muitas estratégias de indexação diferentes podem ser implementadas. De acordo com esta característica, os operadores com os quais um índice GiST pode ser utilizado variam de acordo com a estratégia de indexação. A distribuição habitual do PostgreSQL inclui as classes de operadores GiST equivalentes às classes de operadores da R-Tree, no entanto muitas outras classes estão disponíveis em outros projetos da equipe do PostgreSQL (PostgreSQL, 2011). A Figura 16 demonstra como o índice GIST trabalha dentro de uma consulta.

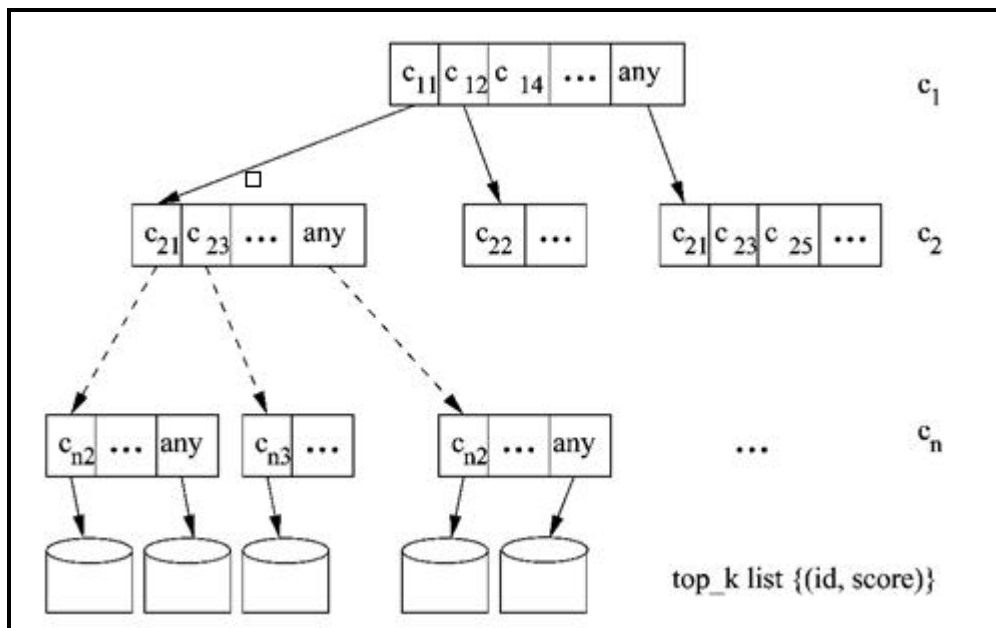


Figura 16 – Sequência de execução do Índice GIST.
FONTE: Tiny, 2009.

Tradicionalmente, implementar um novo método de acesso a índice significa muito trabalho: é necessário entender os processos internos do SGBD. GiST tem uma interface com um alto nível de abstração, requerendo do implementador do método de acesso somente implementar a semântica do tipo de dado a ser acessado. A própria camada da GIST cuida do controle de concorrência, logging e pesquisas na estrutura da árvore (TANAKA, 2011).

A sintaxe para construir um índice de GiST em uma coluna de geometria é como segue:

```
CREATE INDEX [indexname] ON [tablename]
  USING GIST ( [geometryfield] GIST_GEOMETRY_OPS );
```

Figura 17 - Criação de um Índice GIST.

FONTE: Manual PostGIS

4.3 B-Tree

São árvores de pesquisa balanceada feitas para trabalhar bem em discos magnéticos ou outros dispositivos de armazenamento secundários com acesso direto. B-Trees são eficientes pois minimizam operações de entrada e saída no disco (OLIVEIRA, 2001).

São usadas para dados que podem ser ordenados ao longo de um eixo; por exemplo, números, cartas, datas. Dados de GIS não podem ser racionalmente ordenados ao longo de um eixo (o qual é maior, (0,0) ou (0, 1) ou (1,0)?) assim indexação B-tree é inútil para nós (SANTILLI, 2010).

Apresentam bons resultados para pesquisas de intervalo e de igualdade, em dados que podem ser ordenados segundo um determinado atributo. O PostgreSQL usa o algoritmo de Lehman e Yao [10] para criar e gerir as B+Trees. O algoritmo de Lehman e Yao apresenta as seguintes características: nós folha de igual profundidade, cada nó (exceto a raiz e nós folha) tem de $k/2$ a k filhos, os nós folhas têm entre $(k-1)/2$ e $k-1$ valores, as chaves são de tamanho fixo, os algoritmos não usam trincos para leitura e usa uma ligação entre nós folha unidirecional da esquerda para a direita (LEHMAN, 1984).

Veja na Figura 18 uma B-Tree onde os elementos são consoantes. Um nodo interno x contendo $n[x]$ elementos tem $n[x] + 1$ filhos. Todos com a mesma altura na

árvore. Note que a altura de uma B-Tree cresce logaritmicamente com o número de nodos que ela contém (OLIVEIRA, 2001).

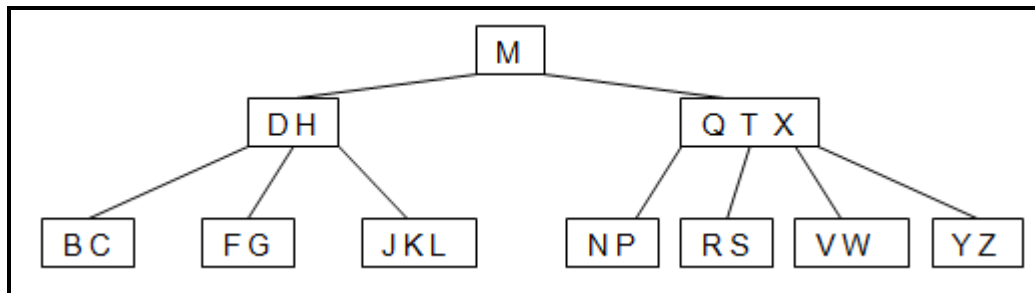


Figura 18 - Elementos de uma B-tree

Fonte: ESTRUTURAS AVANÇADAS DE DADOS. Oliveira, B. 2001.

Em particular, o otimizador do PostgreSQL considera usar uma B-Tree sempre que uma coluna indexada se apresenta numa comparação envolvendo um dos seguintes operadores: $<$, $<=$, $=$, $>=$ e $>$. Construções semelhantes aos operadores apresentados também podem ser implementados utilizando uma procura por índice B-Tree (por exemplo, BETWEEN e IN, mas não IS NULL) (LEHMAN, 1984).

5 CONSULTAS ESPACIAIS, UTILIZANDO ÍNDICES

Dentro de todo contexto abordado até este presente momento, tendo como base a aplicação dos índices dentro das consultas espaciais, foi possível observar que quando aplicados, os índices auxiliam em um melhor desempenho no resultado final de cada consulta realizada dentro de um banco de dados, foi realizada uma pesquisa buscando arquivos que contém dados reais para que os índices sejam testados, foi possível reunidos tais dados e com base neles os testes foram realizados.

É necessário que sejam utilizados alguns softwares que auxiliam no desenvolvimento de tais consultas. Os softwares seguintes foram obtidos e instalados na máquina que os testes foram feitos.

5.1 PostGreSQL

O sistema de gerenciamento de banco de dados objeto-relacional hoje conhecido por *PostgreSQL*, é derivado do pacote *POSTGRES* escrito na Universidade da Califórnia em *Berkeley*. Com mais de uma década de desenvolvimento, o *PostgreSQL* é atualmente o mais avançado banco de dados de código aberto disponível (POSTGRESQL, 2009).

O sistema obteve várias versões durante os anos, sendo que uma das mais importantes versões foi a 8.0 lançada em janeiro de 2005 e entre outras novidades, foi a primeira a ter suporte nativo para *Microsoft Windows* (tradicionalmente, o *PostgreSQL* só rodava de forma nativa em sistemas *Unix* e, em sistemas *Windows* - através da biblioteca *Cygwin*). Dentre as muitas novidades da versão 8.x, pode-se destacar o suporte a *tablespaces*, *savepoints*, *point-in-time recovery*, *roles* e *Two-Phase Commit* (2PC).

O *PostgreSQL* é um dos SGBDs de código aberto mais avançado, contando com recursos de consultas complexas, chaves estrangeiras, integridade transacional, controle de concorrência multi-versão, suporte ao modelo híbrido objeto-relacional, gatilhos, visões, linguagem procedural em várias linguagens (*PL/pgSQL*, *PL/Python*, *PL/Java*, *PL/Perl*) para procedimentos de armazenagem, indexação por texto e estrutura para guardar dados georeferenciados *PostGIS*.

5.2 Gerenciador de Banco de Dados PgAdmin III

PgAdmin é o mais popular recurso de administração de fonte aberta para a Plataforma de Desenvolvimento de código aberto o banco de dados PostgreSQL, o aplicativo pode ser usado em Linux, FreeBSD, Solaris, Mac OSX e plataformas Windows (PGADMIN.COM, 2011).

Foi projetado para responder às necessidades de todos os usuários, desde escrever consultas SQL simples, até o desenvolvimento de bases de dados complexas. A interface gráfica, suporta todos os recursos do PostgreSQL e é de fácil administração. O aplicativo também inclui um destaque de sintaxe o SQL editor, um editor de código do lado do servidor, conforme é demonstrado na Figura 19 (PGADMIN.COM, 2011).

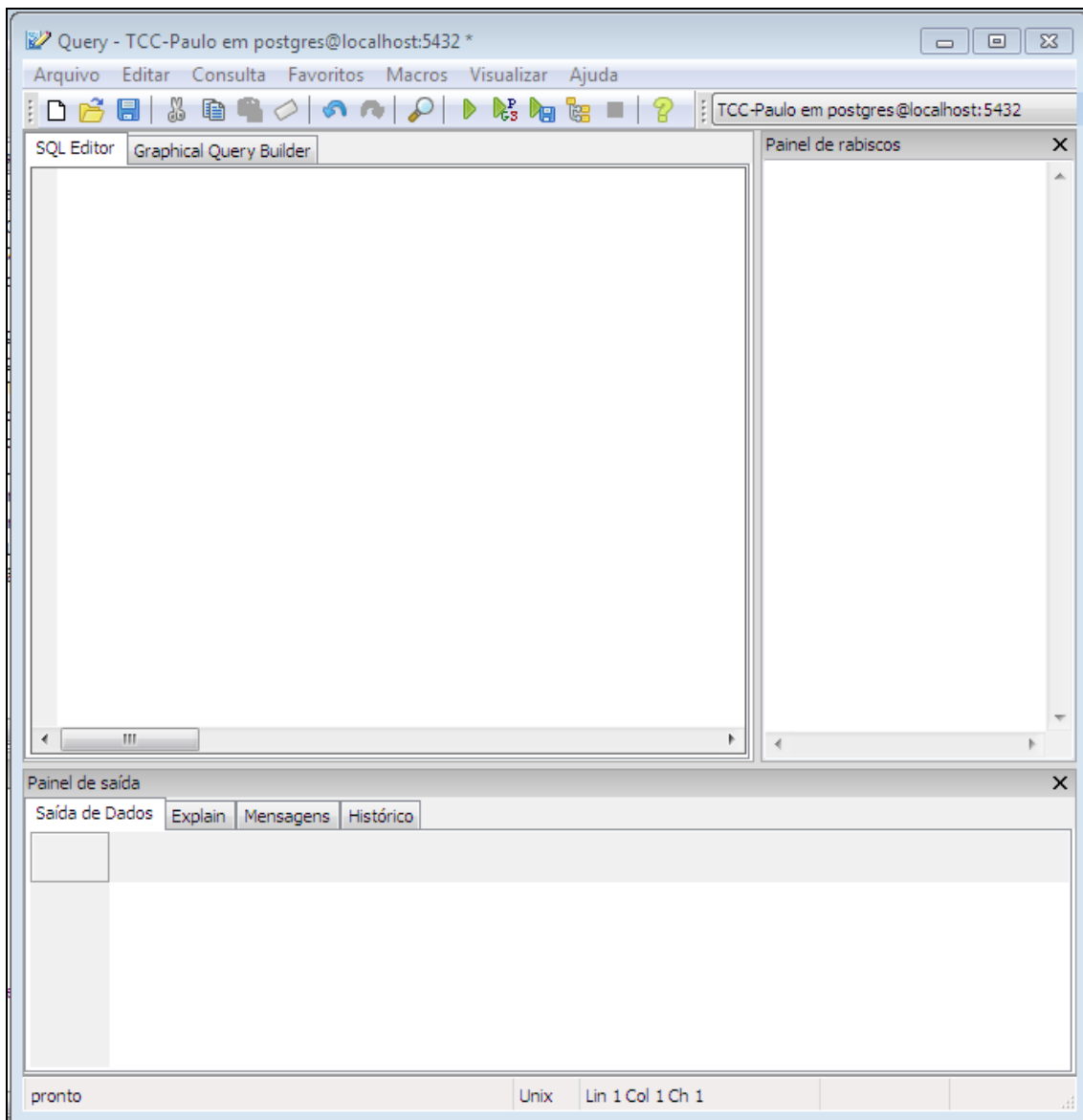


Figura 19 - SQL Editor
FONTE: Aatoria Própria.

O PgAdmin foi desenvolvido por uma comunidade de especialistas PostgreSQL Around the World e está disponível em mais de uma dúzia de línguas. É um Software Livre liberado sob a Licença PostgreSQL. Os arquivos que são carregados dentro do PgAdmin são arquivos do tipo *Shapefile*.

Shapefile é um formato de arquivos contendo informações geo-espacial em formato vetorial para serem usados nos sistemas de informações geográficas. Desenvolvido e mantido pela ESRI (*Environmental Systems Research Institute*) o *shapefile* é um formato de arquivo proprietário, porém mantém uma interoperabilidade com diversos outros softwares, fazendo com que ganhe grande popularidade (ESRI, 1998).

Um shapefile é composto por três arquivos, sendo ele um arquivo principal, um arquivo de indexação e uma tabela *dBASE*. O Arquivo principal (extensão .shp) é onde são guardadas as feições geométricas, o arquivo de índices (extensão .shx) contém em cada registro uma referência dos dados do arquivo principal, e o arquivo *dBASE* (extensão .dbf) contém uma tabela com os dados alfanuméricos dos atributos espaciais do *shapefile* (ESRI, 1998).

5.3 Introdução às Consultas realizadas com os Índices

Para iniciar os testes dos índices é necessário carregar os *shapefile* dentro do PgAdmin III, como é ilustrado na Figura 20.

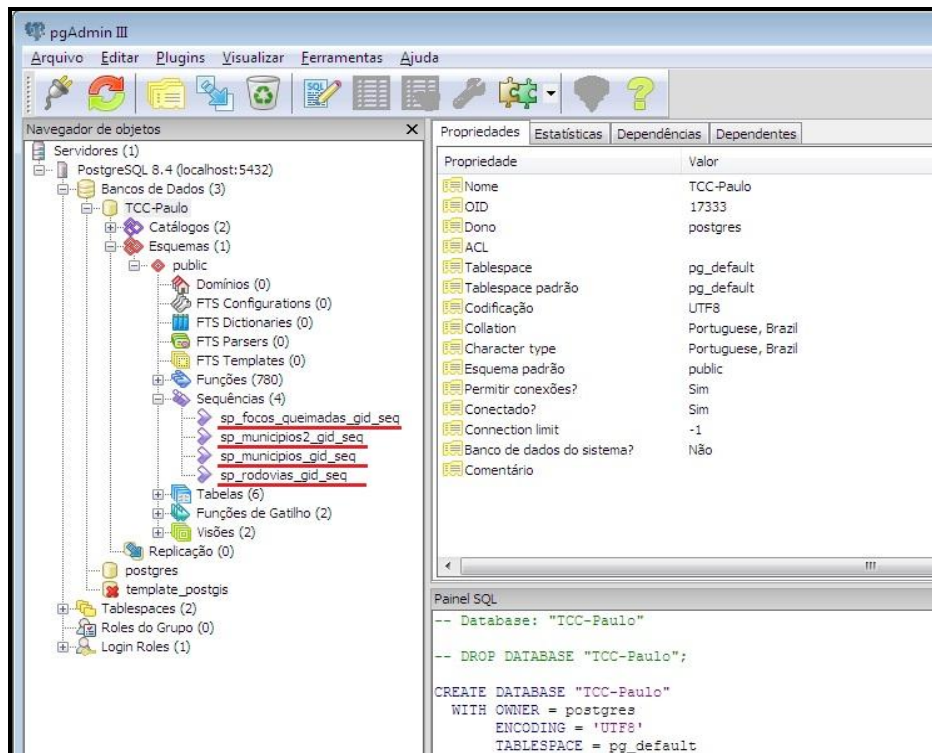


Figura 20 - ShapeFile carregados no PgAdmin
FONTE: Autoria Própria.

Os arquivos grifados em vermelho na Figura 21, são os shape files que serão usados nos testes dos índices, todos possuem dados reais, o próprio PgAdmin já reuniu 780 funções que podem auxiliar nas consultas que o usuário necessite realizar, como demonstrado na Figura 21.

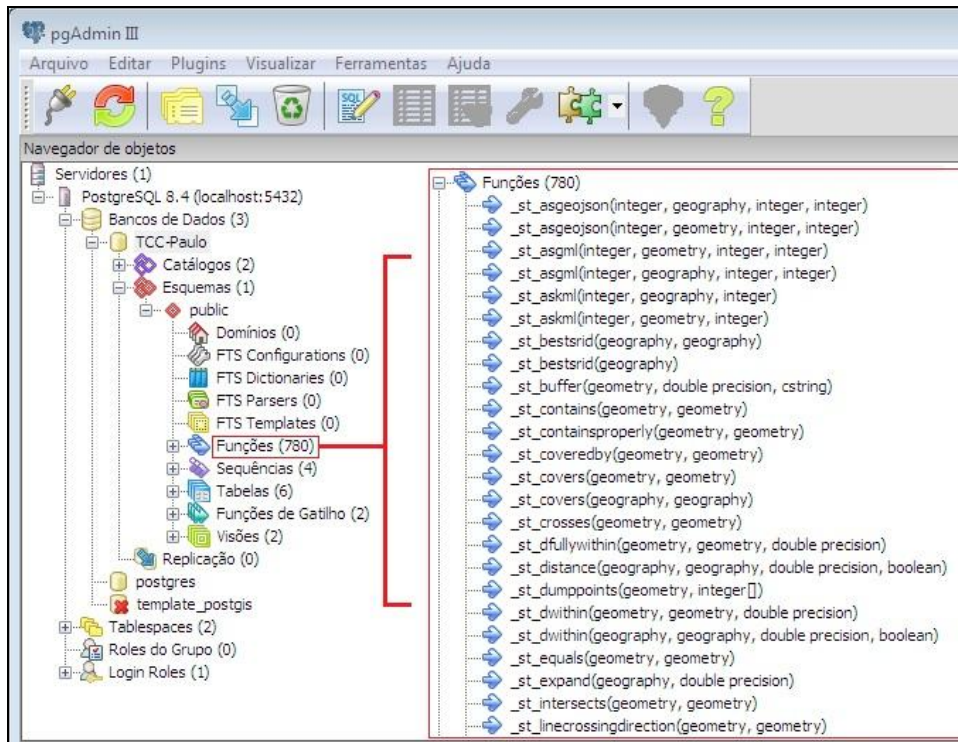


Figura 21 - Funções fornecidas pelo PgAdmin
Fonte: Autoria Própria.

Leve em consideração a tabela SP_MUNICIPIOS criada dentro do banco que esta sendo usado para testar os índices, uma simples consulta (Figura 22) de seleção dentro da tabela *Sp_Municipios* normalmente traria o resultado selecionado.

```
SELECT municipio FROM sp_municipios
WHERE municipio = 'garca';
```

Figura 22 - Consulta na tabela sp_municipios
Fonte: Autoria Própria.

É coerente dizer que essa consulta trará uma linha de resposta contendo todos os dados do município de Garça, isto é fato, porém sem que o sistema esteja preparado para tal consulta ele terá de percorrer todas as linhas da tabela em busca do município consultado. Se a tabela possuir um numero de registros pequeno, o resultado será retornado de uma forma rápida, o que não acontece neste caso, a tabela de municípios trás todos os registros das cidades do estado de São Paulo (Figura 23), sendo assim o sistema gastará um tempo maior processando a consulta. Utilizando os índices este fato pode ser evitado.

SQL Editor Graphical Query Builder

```
SELECT * FROM sp_municipios
```

Panel de saída

Saída de Dados Explain Mensagens Histórico

gid integer	id integer	sprarea numeric	sprperim numeric	sprotulo character varying(25)	sprnome character varying(25)	municipio character varying(64)	codigo numeric	ei numeric	area_km2 numeric	populacao numeric	dens_popul numeric	latitude character v	longitude character v	obje char	objei char	the_geom geometry
1	1	41517081	109229.8046	3500105	3500105		0.000000	0.00000000	0.00000000	0.00000000	0.00000000			0	0	0106000020
2	2	21242867	74272.82809	3500204	3500204		0.000000	0.00000000	0.00000000	0.00000000	0.00000000			1	1	0106000020
3	3	18535260	74870.90630	3509452	3509452	CAMPINA DO MONTE ALEGRE	3509452.(SP	184.000000	4547.00000	24.69097000	23 35 31	o 48 28 37	107 10	0106000020		
4	4	35532630	121350.0390	3516606	3516606	GALLIA	3516606.(SP	356.000000	7503.00000	21.08943000	22 17 27	o 49 33 10	189 100	0106000020		
5	5	12327564	60914.45700	3501707	3501707	AMERICO BRASILENSE	3501707.(SP	123.000000	23993.00000	194.34658999	21 43 26	o 48 6 7.	19 101	0106000020		
6	6	55713068	143802.4843	3516705	3516705	GARCA	3516705.(SP	556.000000	40481.00000	72.83915000	22 12 39	o 49 39 21	190 102	0106000020		
7	7	17901804	63243.07420	3516804	3516804	GASTAO VIDIGAL	3516804.(SP	181.000000	3739.00000	20.67669999	20 47 56	o 50 11 13	191 103	0106000020		
8	8	24450955	88841.13280	3516853	3516853	GAVIAO PEIXOTO	3516853.(SP	244.000000	4453.00000	18.27202000	21 50 20	o 48 29 42	192 104	0106000020		
9	9	49228889	121896.1406	3516903	3516903	GENERAL SALGADO	3516903.(SP	493.000000	10551.00000	21.39099999	20 38 52	o 50 21 39	193 105	0106000020		
10	10	67773830	123254.5546	3517000	3517000	GETULINA	3517000.(SP	675.000000	9223.00000	13.65471999	21 47 56	o 49 55 44	194 106	0106000020		
11	11	27521753	100573.1406	3517109	3517109	GLICERIA	3517109.(SP	274.000000	4174.00000	15.22875000	21 23 2	o 50 12 39	195 107	0106000020		
12	12	26902076	118957.0859	3517208	3517208	GUACARA	3517208.(SP	269.000000	8426.00000	31.28867000	21 37 19	o 49 47 56	196 108	0106000020		
13	13	21764211	68159.19530	3517307	3517307	GUAIMBE	3517307.(SP	217.000000	4903.00000	22.54740999	21 54 36	o 49 53 49	197 109	0106000020		
14	14	80047430	168011.5938	3509502	3509502	CAMPINAS	3509502.(SP	796.000000	908906.0000	1142.2821200	22 54 21	o 47 3 39	108 11	0106000020		
15	15	12635336	191812.7968	3517406	3517406	GUAIRA	3517406.(SP	1259.000000	33114.00000	26.31039000	20 19 4	o 48 18 39	198 110	0106000020		
16	16	32731091	88745.30469	3517505	3517505	GUAPIACU	3517505.(SP	325.000000	12566.00000	38.65829000	20 47 42	o 49 13 12	199 111	0106000020		
17	17	47451635	121218.5078	3500303	3500303		0.000000	0.00000000	0.00000000	0.00000000				2	112	0106000020
18	18	25421609	95465.42190	3501806	3501806	AMERICO DE CAMPOS	3501806.(SP	254.000000	5582.00000	21.98669999	20 17 56	o 49 43 55	20 113	0106000020		
643	643	61227782	199253.6406	3516200	3516200	FRANCA	3516200.(SP	607.000000	267235.0000	440.01643000	20 32 20	o 47 24 3.	185 96	0106000020		
644	644	49203512	35076.30000	3516309	3516309	FRANCISCO MORATO	3516309.(SP	49.0000000	106215.0000	2160.49433000	23 16 55	o 46 44 42	186 97	0106000020		
645	645	13562987	68173.51559	3516408	3516408	FRANCO DA ROCHA	3516408.(SP	134.000000	98310.00000	734.12678000	23 19 19	o 46 43 37	187 98	0106000020		
646	646	13876873	51913.56250	3516507	3516507	GABRIEL MONTEIRO	3516507.(SP	139.000000	2414.00000	17.42607999	21 31 51	o 50 33 18	188 99	0106000020		

Figura 23 - select na tabela sp_municipios
 Fonte: Autoria Própria.

Observe que foram retornadas 646 linhas de resposta, existem bases de dados que possuem até 30 mil registros cadastrados, para se trabalhar neste tipo de banco de dados, os índices são a solução mais indicada, para um bancos que possui um numero maior de dados é recomendado que se use das técnicas de indexação de dados, desta forma as buscas serão mais eficientes.

Os índices quando aplicados dentro de um SGBD, melhoram o desempenho da consulta, é necessário que quando se utilizar dos índices é preciso ter cuidado se usados de forma errada podem trazer um resultado contrario ao esperado (ao invés de rapidez, retardar ainda mais o processo).

Aplicando os índices, o primeiro exemplo demonstrado utiliza da função *ST_INTERSECTS*, função que retorna uma geometria que representa a interseção do conjunto de pontos das geometrias.

Em outras palavras - as partes que a geometria A e B compartilham entre si, se as geometrias não compartilham qualquer espaço (são disjuntos), então uma coleção de geometria vazia é retornada. Esta função é muito útil para descobrir dentro de uma geometria quais as regiões que em algum momento se interceptam.

Tendo o município de São José dos Campos como base, será utilizado a função *ST_INTERSECTS* para descobrir quais os municípios que são vizinhos a essa cidade. O resultado é demonstrado na Figura 24.

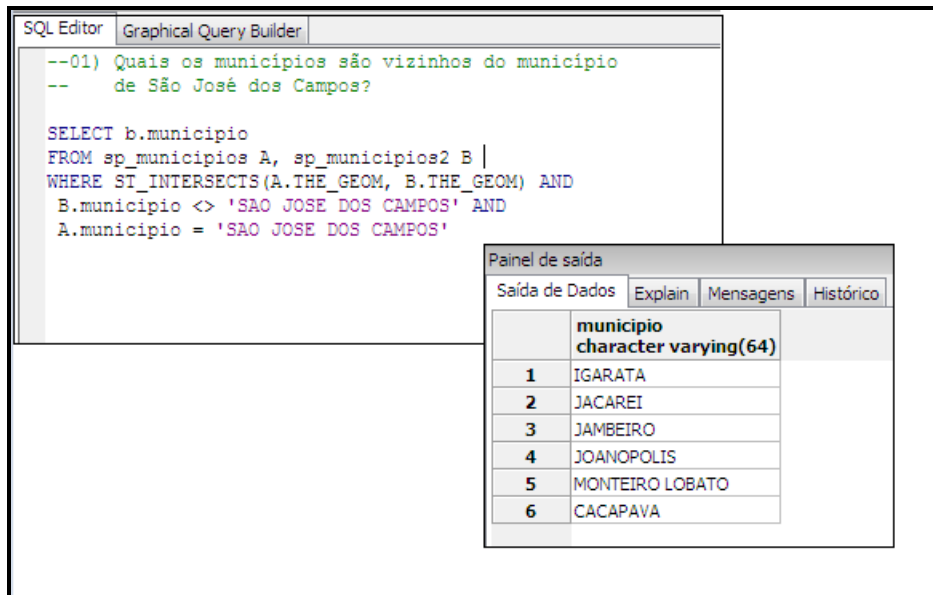


Figura 24 - Consulta aplicada sem utilização de índices.
FONTE: Autoria Própria.

Agora analisando a consulta utilizando dos argumentos do “Explain Analyse” (Figura 25), podemos observar na linha 5, que foi usado uma forma de busca seqüencial na tabela, para que os dados fossem obtidos, e o tempo de resposta foi de 0.00...315.46.

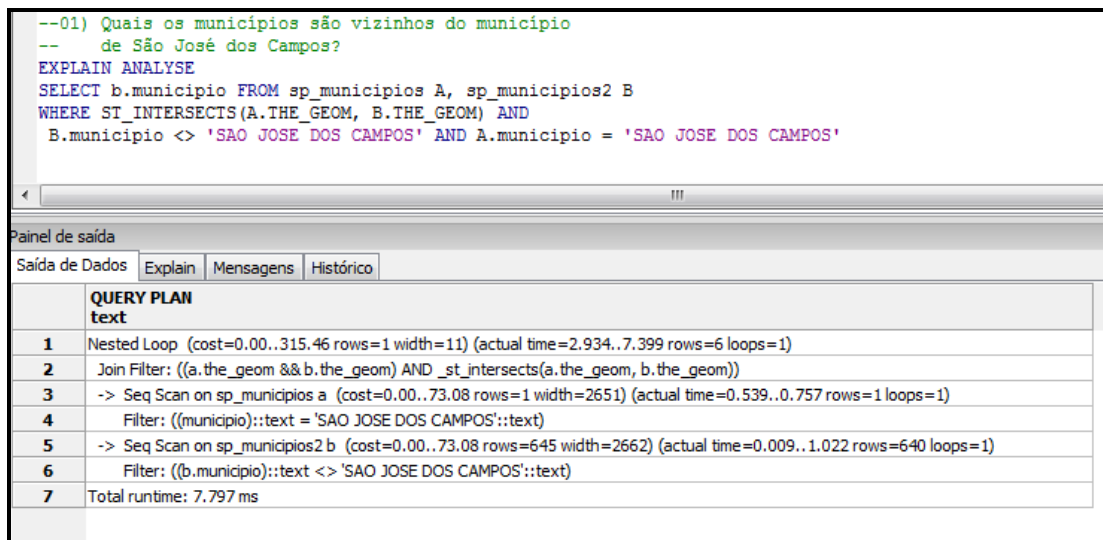


Figura 25 - Resultado da Consulta sem utilizar índices
FONTE: Autoria Própria.

Os campos abordados nesta consulta são do tipo geométrico desta forma foi criado um índice do tipo *GIST*, para melhorar o desempenho da consulta é necessário criar um índice para um dos campos utilizados na cláusula WHERE. Como o campo utilizado é um campo geométrico, o índice a ser criado será do tipo GIST. A figura 26 demonstra a criação do índice.

```
create index idx_vizinhos on sp_municipios using GIST(the_geom)
```

!!!

Painel de saída

Saída de Dados Explain Mensagens Histórico

Consulta executada com sucesso sem resultados em 92 ms.

Figura 26 - Criação do índice GiST.
FONTE: Autoria Própria.

Utilizando novamente do *explain analyse* na Figura 27 podemos ver na linha 3 que o índice criado foi utilizado na consulta, e que o tempo de resposta foi de 0.00...250.66, resultado menor do que o anterior (quando efetuada a consulta no modo sequencial).

```
--01) Quais os municípios são vizinhos do município
-- de São José dos Campos?
EXPLAIN ANALYZE
SELECT b.municipio FROM sp_municipios A, sp_municipios2 B
WHERE ST_INTERSECTS(A.THE_GEOM, B.THE_GEOM) AND
B.municipio <> 'SAO JOSE DOS CAMPOS' AND A.municipio = 'SAO JOSE DOS CAMPOS'
```

!!!

Painel de saída

Saída de Dados Explain Mensagens Histórico

	QUERY PLAN
	text
1	Nested Loop (cost=0.00..250.66 rows=1 width=11) (actual time=2.586..6.748 rows=6 loops=1)
2	Join Filter: ((a.the_geom && b.the_geom) AND st_intersects(a.the_geom, b.the_geom))
3	-> Index Scan using idx_vizinhos on sp_municipios a (cost=0.00..8.27 rows=1 width=2651) (actual time=0.147..0.152 rows=1 loops=1)
4	Index Cond: ((municipio)::text = 'SAO JOSE DOS CAMPOS)::text)
5	-> Seq Scan on sp_municipios2 b (cost=0.00..73.08 rows=645 width=2662) (actual time=0.016..0.958 rows=640 loops=1)
6	Filter: ((b.municipio)::text <> 'SAO JOSE DOS CAMPOS)::text)
7	Total runtime: 7.134 ms

Figura 27 - Resultado da consulta utilizando índices.
FONTE: Autoria Própria.

O exemplo a seguir traz o resultado de qual o centróide do município de cruzeiro em graus, e traz também duas outras funções a *St_X* e *St_Y* (Figura 28). O *St_X* retorna a coordenada do ponto X, ou NULL se não estiver disponível. E o *St_Y* tem a mesma função do *St_Y* porém retorna as coordenadas do ponto Y.

```

--3) QUAL É O CENTROIDE DO MUNICÍPIO DE CRUZEIRO
-- EM GRAUS

SELECT ST_X(ST_CENTROID(THE_GEOM)), ST_Y(ST_CENTROID(THE_GEOM))
FROM SP_MUNICIPIOS WHERE municipio = 'CRUZEIRO'

```

Painel de saída		
Saída de Dados		
	st_x double precision	st_y double precision
1	-45.0053183998485	-22.5514132782176

Figura 28 - Consulta aplicada sem índice B-tree.
 FONTE: Autoria Própria.

Como já visto, as funções que foram utilizadas para realizar esta consulta (St_X e St_Y e St_Centroid), são funções que são aplicadas a pontos. Desta forma o índice B-tree pode ser aplicado para melhorar o tempo de processamento desta questão. Primeiramente vamos aplicar o “*Explain Analise*”, na consulta anterior, o resultado é expresso na Figura 29.

```

--3) QUAL É O CENTROIDE DO MUNICÍPIO DE CRUZEIRO
-- EM GRAUS

Explain analyse
SELECT ST_X(ST_CENTROID(THE_GEOM)), ST_Y(ST_CENTROID(THE_GEOM))
FROM SP_MUNICIPIOS WHERE municipio <> 'CRUZEIRO'

```

Painel de saída	
Saída de Dados	
	QUERY PLAN text
1	Seq Scan on sp_municipios (cost=0.00..79.53 rows=645 width=2651) (actual time=0.210..56.673 rows=640 loops=1)
2	Filter: ((municipio)::text <> 'CRUZEIRO)::text)
3	Total runtime: 57.006 ms

Figura 29 - Análise da Consulta sem utilização de Índices.
 FONTE: Autoria Própria.

O tempo de execução da consulta foi de 0.00..79.53, e a forma de busca que foi aplicada na consulta foi à forma seqüencial, como essa consulta retorna coordenadas de um ponto foi criado um índice do tipo *B-tree*, sua criação é demonstrada na **Figura 30**.

```

CREATE INDEX idx_ponto ON sp_municipios(municipio)

```

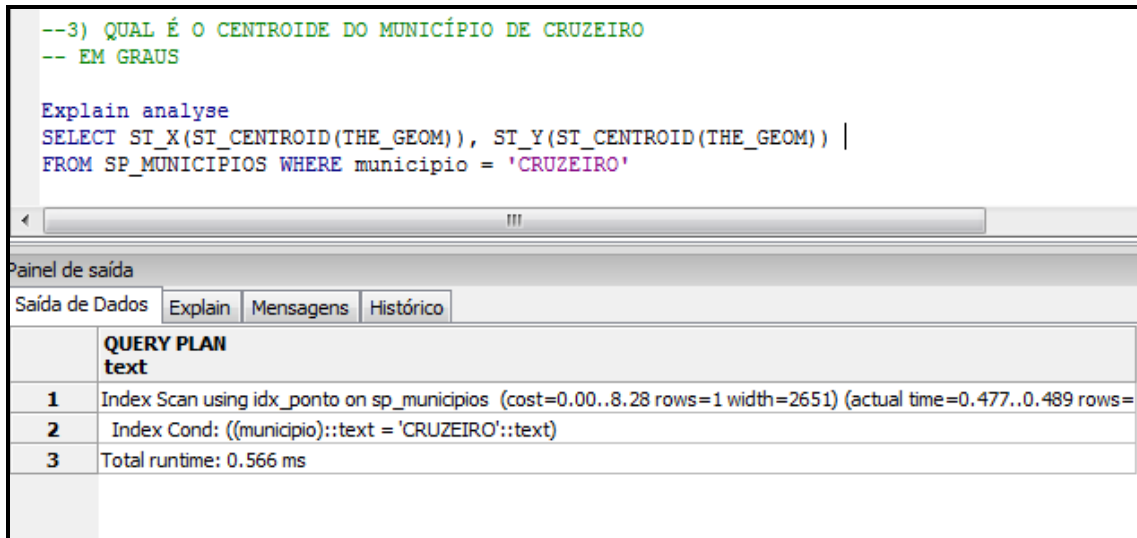
Painel de saída

Saída de Dados

Consulta executada com sucesso sem resultados em 53 ms.

Figura 30 - Criação do índice B-tree
 FONTE: Autoria Própria.

O próximo passo após a criação do índice é testes novamente a consulta utilizando o *Explain Analyse* novamente, o resultado é exibido na Figura 31.



The screenshot shows a database query window with the following content:

```
--3) QUAL É O CENTROIDE DO MUNICÍPIO DE CRUZEIRO  
-- EM GRAUS  
  
Explain analyse  
SELECT ST_X(ST_CENTROID(THE_GEOM)), ST_Y(ST_CENTROID(THE_GEOM)) |  
FROM SP_MUNICIPIOS WHERE municipio = 'CRUZEIRO'
```

Below the query, there is a 'Painel de saída' (Output Panel) with tabs for 'Saída de Dados', 'Explain', 'Mensagens', and 'Histórico'. The 'Explain' tab is selected, showing the following query plan:

	QUERY PLAN text
1	Index Scan using idx_ponto on sp_municipios (cost=0.00..8.28 rows=1 width=2651) (actual time=0.477..0.489 rows=
2	Index Cond: ((municipio)::text = 'CRUZEIRO'::text)
3	Total runtime: 0.566 ms

Figura 31 - Utilização do índice B-Tree
Fonte: Autoria Própria.

Observe que o tempo de execução foi muito inferior comparado à execução realizada sem a criação do índice, o tempo baixou de 0.00..79.53 para 0.00..8.28.

Abaixo foi criada uma tabela de apresentação dos resultados obtidos pelas consultas realizadas.

6.4 Resultados obtidos

Com base nos testes realizados, é possível observar que com a utilização dos índices as consultas são efetuadas mais rápidas, no caso das consulta exibidas a diferença de tempo pode ser considerada baixa, porém é necessário lembrar que existem consultas espaciais que podem demorar horas para serem executadas, e nestes casos os índices reduzem drasticamente o tempo de compilação.

Nas pesquisas realizadas anteriormente, no primeiro teste a sua compilação sem a utilização de índices teve um tempo de compilação de 0.00...315.46, o que mudou quando o índice foi aplicado, baixando esse tempo para 0.00...250.66.

Já no segundo teste o primeiro tempo obtido ao se compilar a consulta sem utilizar índices foi de 0.00..79.53, já quando o índice b-tree foi utilizado o tempo reduziu para 0.00..8.28.

Com esses valores foi criado um gráfico de comparação para demonstrar a melhora no desempenho das consultas quando se utiliza dos índices. Com ele fica nítido que quando utilizado os índices auxiliam na execução das consultas, trazendo mais rapidez para o desenvolvedor.

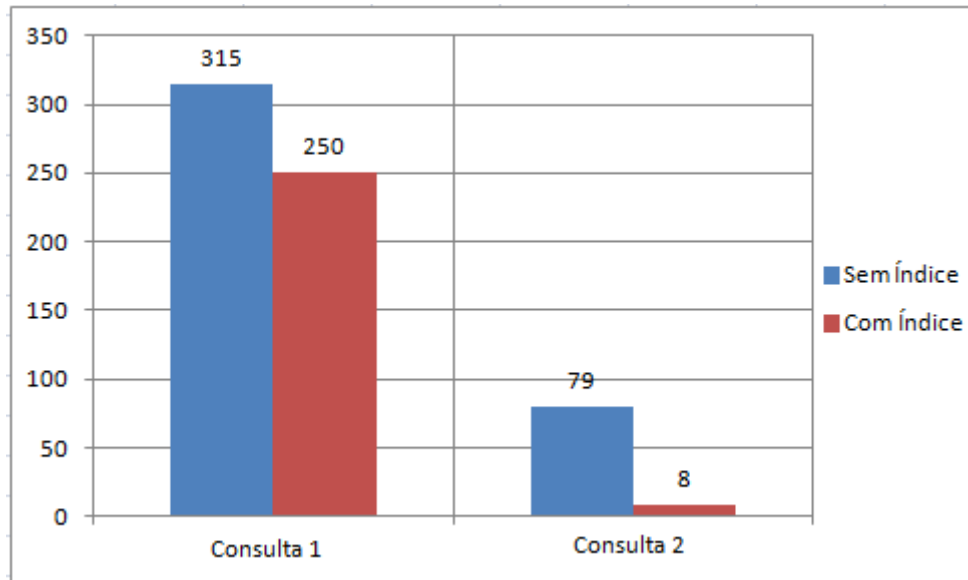


Figura 32 - Resultados Obtidos
Fonte: Autoria própria.

6 CONCLUSÃO

O principal objetivo deste trabalho foi compreender o funcionamento dos índices espaciais, e a sua aplicação específica para cada tipo de dado espacial, as consultas espaciais muitas das vezes fazem com que o desenvolvedor tenha o incômodo de esperar por todo o processo de busca pelos resultados. Porém após estudar os índices pude concluir que quando utilizada essa ferramenta de forma coerente, ela se torna poderosa, trazendo ganho de tempo para o desenvolvedor, e poupando a máquina de executar processos longos e demorados, enfim é um ganho de produtividade excelente.

Foi possível compreender a função de cada índice quando aplicado para cada tipo de dado, observando por exemplo, que quando utilizado o índice GIST para dados os quais não são geométricos, este indexador torna-se inviável, não podendo auxiliar na melhora do processo. Os índices B-tree e R-tree, ambos árvores de buscas, quando aplicados a pesquisas de intervalo ou de igualdade, os quais são dados do tipo ponto, trazem uma melhora significativa ao processo.

Pode-se concluir que quando aplicados os índices para dados os quais não são indicados, por exemplo, criar um índice que seja do tipo Gist, para um tipo de dado que não seja geométrico ou criar um índice B-tree para dados que não sejam do tipo ponto fazem com que ao se executar a consulta, o índice não sejam utilizados.

REFERÊNCIAS BIBLIOGRÁFICAS

ALESSIO, A. Banco de Dados Espacial. Disponível em:

<<http://www.slideshare.net/augustocolombelli/banco-de-dados-espaciais-2501605>>.

Acesso dia: 27/05/2011.

BEAUJARDIERE, J. OpenGIS WEB Map Service Implementation Specification. Ano:

2006. Version 1.3.0.

CASANOVA, M, A. CÂMARA, G. DAVIS Jr. VINHAS, L. QUEIROZ, G, R. Banco de Dados Geográficos Ano: 2005. Editora: Mundo GEO. Pág 181-281.

CÂMARA, Gilberto, QUEIROZ, Gilberto Ribeiro de. Arquitetura de Sistemas de Informação Geográfica. Disponível em:

<<http://www.dpi.inpe.br/gilberto/livro/introd/cap3-arquitetura.pdf>> Acesso dia:

16/04/2011

CÂMARA, Gilberto. Davis, Clodoveu. Introdução. Disponível em:

<<http://www.dpi.inpe.br/gilberto/livro/introd/cap1-introducao.pdf>> Acesso dia:

16/03/2011

CÂMARA, Gilberto. FERREIRA, Karine Reis. QUEIROZ, Gilberto Ribeiro de.

Arquitetura de Banco de Dados Geográfico. Disponível em: <[http://mtc-](http://mtc-m12.sid.inpe.br/col/sid.inpe.br/sergio/2004/10.07.15.53/doc/cap2.pdf)

[m12.sid.inpe.br/col/sid.inpe.br/sergio/2004/10.07.15.53/doc/cap2.pdf](http://mtc-m12.sid.inpe.br/col/sid.inpe.br/sergio/2004/10.07.15.53/doc/cap2.pdf)> Acesso dia:

16/05/2011

CÂMARA, Gilberto. Representação Computacional de Dados Geográfico. Disponível em: <<http://www.dpi.inpe.br/livros/bdados/cap1.pdf>> Acesso dia: 16/03/2011

CAMARA, G. Arquitetura de Sistemas de Informação Geografica. Disponível em:

<www.dpi.inpe.br/gilberto/livro/introd/cap3-arquitetura.pdf> Acesso dia : 07/05/2011

CASANOVA, M. Banco de Dados Geográfico, Processamento de consultas e gerência de transações. Disponível em: < <http://www.dpi.inpe.br/livros/bdados/cap7.pdf>> Acesso dia : 31/05/2011.

CASANOVA, M, A. CÂMARA, G. DAVIS Jr. VINHAS, L. QUEIROZ, G, R. Banco de Dados Geográficos Ano: 2005. Editora: Mundo GEO. Pág 181-281.

CIFERRI, R. Análise da Eficiência de Métodos de Acesso Espaciais em Termos da Distribuição Espacial dos Dados. Disponível em:
<<http://www.dpi.inpe.br/geoinfo/geoinfo2001/papers/145ciferri.pdf>>. Acesso dia: 31/05/2011.

CIFERRI, R. Banco de Dados Espaciais. Disponível em:
<http://gbd.dc.ufscar.br/download/files/courses/NonConventionalDB_2010/RicardoCiferri_slidesBDE.pdf>. Acesso dia: 04/09/2011.

CIFERRI , C. CAROSIA, A. Dados Espaciais e Indexação. Disponível em:
<http://wiki.icmc.usp.br/images/2/2b/SCC5911-04-ApresentacaoRtree.pdf> Acessado em: 11/09/2011.

DAVIS, C. CÂMARA , G. ROCHA C. UNIVERSIDADE FEDERAL DE VIÇOSA - MG. Fundamentos de Geoprocessamento. Disponível em:
<<http://www.ltc.ufes.br/geomaticsee/Modulo%20Geoprocessamento.pdf>>. Acesso dia 20/05/2011

DAVIS, C. QUEIROZ, G. Métodos de acesso para dados espaciais. Disponível em
<http://www.dpi.inpe.br/livros/bdados/cap6.pdf>. Acesso dia: 31/05/2011

DIAS, R. YWASAKI, F. Publicações - Noção de Cartografia, Geoprocessamento e Sistemas de Informação Geográfica. Disponível em:
<www.leb.fmvz.usp.br/.../geoprocessamento/.../nocoas_cartografia.pdf>. Acesso dia: 20/05/2011.

ESRI. ESRI Sshapefile Technical Description. Disponível em:
<<http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>> Acesso dia: 16/04/2010

FARIA, C. Geoprocessamento. Disponível em:
<<http://www.infoescola.com/cartografia/geoprocessamento/>>. Acesso dia 20/05/2011

FU, C. Y. et al. (1997). A study of concurrent operations on R-trees. *Information Sciences*, 98(1-4):263–300.

GONZAGA, LUIZ J. (2007) Dominando o PostgreSQL. Rio de Janeiro: Editora Ciência Moderna Ltda.

HEREDIA, R, L. IOCHPE, C. COMBA, J. Explorando a Multidimensionalidade da *Kd-Tree* para Suporte a Temporalidade em Dados Espaciais Vetoriais do Tipo Ponto. Disponível em: <<http://www.inf.ufrgs.br/~comba/papers/2003/geo-info.pdf>>. Acesso dia 12/09/2011

IRRIGART. Banco de Dados. Disponível em:
http://www.agenciadeaguapcj.org.br/download/RS-04-06_Capitulo-10.pdf/. Acesso dia: 13/04/2010.

JUNIOR, A, F, R. Introdução ao Oracle PARTE 1. Disponível em:
<<http://www.juliobattisti.com.br/tutoriais/almirrivvas/oracle003.asp>>. Acesso dia: 12/10/2011.

LEHMAN, P. YAO, S. Efficient Locking for Concurrent Operations on B-Trees. Disponível em: <citeseerx.ist.psu.edu/.../download?doi=10...>. Acessado dia: 30/10/2011.

LIMA, A. FERREIRA, F. ALVES N. YURI P. Banco de Dados Geográfico. Disponível em: <<http://im.ufba.br/pub/MATA60/SemestreCorrente/bdg.pdf>> Acesso dia: 23/05/2010

LORENÇO, B, M, P. Um Estudo sobre Recursos de Tratamento de Dados Espaciais em SGBDs Geográficos. Disponível em:

<http://www.csr.ufmg.br/geoprocessamento/publicacoes/paula_lourenco.pdf>.

Acessado em 15/09/2011.

MARBLE, D. Geographical information system: ano Overview. In: Pecora 9 Conference, Sioux Falls, S. D. Proceedings... Sioux Falls, S. D. V.1, p. 18-24, 1984.

MANICA, H. DANTAS, M. A. R. TODESCO, J. Processamento de Consultas dependentes de localização em cache Semântico para Dispositivos Móveis. Disponível em: <<http://revistaseletronicas.pucrs.br/ojs/index.php/hifen/article/viewFile/3894/2961>>

Acesso dia: 05/09/2011.

NEBERT, D. Nebert, D. Developing Spatial Data Infrastructures: The SDI Cookbook Version 2.0. Ano: 2004.

OLIVEIRA, T. RODRIGUES, S. LIMA, P. CARDOSO, M. Processamento de Consultas dependentes de localização em cache Semântico para Dispositivos Móveis. Disponível em:

<<http://revistaseletronicas.pucrs.br/ojs/index.php/hifen/article/viewFile/3894/2961>>

Acesso dia: 05/09/2011.

ORACLE. Banco de Dados Oracle 11g para data warehousing e business intelligence. Disponível em: <www.oracle.com/database> Acesso dia: 05/09/2011.

QUEIROZ, G. Web Feature Service . Disponível em:

<www.dpi.inpe.br/~gribeiro/apresentacoes/wfs_2007.pdf> Acesso dia: 06/05/2011.

ORACLE. Banco de Dados Oracle 11g para data warehousing e business intelligence. Disponível em: <www.oracle.com/database> Acesso dia: 12/10/2011.

OLIVEIRA, B. DREWS, I. COSTA, F. ARAUJO, F. LAZZARI, G. GASPARETO, O. Estruturas Avançadas de Dados. Disponível em:

<<http://www.oocities.org/grupotrees/downloads.htm>>. Acesso dia: 05/11/2011.

PGADMIN. Introduction PgAdmin III. Disponível em: < <http://www.pgadmin.org/>>
Acesso dia: 13/10/2011.

POSTGREESQL. PostgreSQL 8.4.8 Documentation. Disponível em
<http://www.mcs.vuw.ac.nz/technical/software/PostgreSQL/gist-examples.html>.
Acessado dia: 30/10/2011.

SANTILLI, S. Manual PostGIS. Disponível em
<<http://pt.scribd.com/doc/38556149/Manual-PostGIS>>. Acessado dia: 31/10/2011.

TINY. generalized search trees for database systems. Disponível em:
<<http://www.tactic-tech.com/blog/generalized-search-trees-for-database-systems/>>.
Acessado dia: 05/11/2011.

WIKIPIDIA. Centróide. Disponível em: <<http://pt.wikipedia.org/wiki/Centroide>>
Acesso dia: 16/10/2011.