

Otimização de um banco de dados geográficos utilizando PostGIS

João Luís dos Santos¹
João Francisco Gonçalves Antunes²
Júlio César Dalla Mora Esquerdo²
Alexandre Camargo Coutinho²
Lucas Porto Maziero³

¹ Fundação de Ciência, Aplicações e Tecnologias Espaciais - FUNCATE
Av. Dr. João Guilhermino 429 - 12210-131 - São José dos Campos, SP, Brasil
joao.santos@funcate.org.br

² Embrapa Informática Agropecuária
Av. André Tosello, 209 - Caixa Postal 6041 - 13083-886 – Campinas, SP, Brasil
{joao.antunes, julio.esquerdo, alex.coutinho}@embrapa.br

³ Instituto de Computação – Universidade Estadual de Campinas (UNICAMP)
Av. Albert Einstein, 1251 - 13083-852 – Campinas, SP, Brasil
lucas.maziero@ic.unicamp.br

Abstract. The pervasive usage of geoprocessing tools between researchers, students and users' communities leads to resources integration and allowed online tools to accomplish demands of people's daily life. With the progress of Information and Communications Technology (ICT), new computational resources are constantly becoming available and the effort of research institutes goes towards to release online interactive systems to the public that allow geospatial data visualization and manipulation. However, behind all provided resources there are a lot of process dealing with data preparation, storage and recovery. In this way, this work was carried out to deal with these tasks, implementing logical models of geographic database in order to store land use and land cover maps produced for the Brazilian Legal Amazon. The dataset used to handle the experiments was obtained from TerraClass initiative, a project created to qualify the deforested areas for that region. The main objective of this study was to identify which thematic classes intersect with a set of municipalities in the state of Pará, returning results in an acceptable time window for a web application. To accomplish that, a PostGIS spatial database was established and geoprocessing techniques and spatial operators were used to perform the tasks. To speed-up the queries execution time, a new model was proposed, reorganizing data in a new object structure which results in an 80 times faster performance of data queries. Thereafter, indexes were implemented and the cost of query execution was optimized more than 50%, addressing an important issue regarding a multi-user environment scenario.

Palavras-chave: geoprocessing, spatial operators, geoprocessamento, operadores espaciais, TerraClass.

1. Introdução

As técnicas e ferramentas de geoprocessamento ocupam um espaço importante na sociedade contemporânea e seu uso está disseminado nas comunidades de pesquisadores, estudantes e usuários. Ao mesmo tempo, a integração de recursos computacionais tem permitido a utilização pervasiva de ferramentas para a visualização de mapeamentos, a sobreposição de camadas temáticas ou mesmo a obtenção de registros instantâneos georreferenciados, relacionados às atividades do cotidiano das pessoas. Com o avanço das Tecnologias da Informação e Comunicação (TIC), novos recursos computacionais tornam-se disponíveis constantemente e o esforço das instituições de pesquisa ocorre no sentido de tornar acessível ao público mecanismos de consulta e sistemas interativos, que permitam a visualização e manipulação de dados geoespaciais. Alesheikh et al. (2002) discutem a distribuição de dados geoespaciais e destacam como as funções de coleta, armazenamento, recuperação, análise e visualização desses dados são remodeladas em sistemas de informação desenvolvidos para a plataforma Web.

Por detrás de cada recurso disponibilizado na Internet existe um processo de preparação, armazenamento e recuperação de dados até sua disponibilização aos usuários. No caso de dados geoespaciais, a coleta ocorre de diversas formas, seja através de sensores a bordo de satélites orbitais, seja através da obtenção de registros fotográficos da superfície terrestre ou mesmo pelo trabalho de campo de pesquisadores e estudantes (Jensen, 2009). Posteriormente, por meio de Sistemas de Informação Geográfica (SIG) são aplicadas técnicas matemáticas e computacionais para manipulação dos atributos descritivos e das geometrias dos diferentes tipos de dados geográficos. Para realizar essas ações, os SIG apresentam, entre outros mecanismos, algoritmos de análise espacial através de operações topológicas (Egenhofer e Franzosa, 1991) e mecanismos para o processamento de imagens e manipulação dos dados geográficos através de álgebra de mapas (Mather, 2004) e (Tomlin, 1990). Essas características, entretanto, posicionam os SIG como softwares *standalone*, onde os recursos necessários para o processamento dos dados e a geração dos resultados ocorrem de forma isolada, permanecendo disponíveis apenas para seu usuário-produtor.

A partir do paradigma de ampla conectividade da Internet, a utilização dos SIG se intensificou e permitiu sua disseminação ao atingir milhões de usuários na sociedade, nas indústrias, no governo e nos negócios para educação e pesquisa (Fu e Sun, 2010). Nesse cenário, os SIG têm se transformado em um aplicativo para Internet bastante vantajoso, possibilitando às pessoas se beneficiarem da Web e dos diversos recursos e serviços que ela propicia (Longley et al., 2005). Entretanto, para a distribuição de dados em um ambiente multiusuário como a internet, existe a demanda por mecanismos que possibilitem controlar a requisição, recuperação, processamento e disponibilização desses dados aos seus usuários (Ceri et al., 2003). Um Sistema Gerenciador de Banco de Dados (SGDB) atende a esses propósitos, além de permitir a consulta e modificação de dados previamente armazenados.

Em um SGBD, essas tarefas podem ser realizadas através de uma interface de programação de aplicações, *Application Programming Interface* (API) e através de *drivers* que executam comandos em *Structured Query Language* (SQL), uma linguagem estruturada de consulta que permite a manipulação de banco de dados relacionais (Ramakrishnan e Gehrke, 2000). Da mesma forma, um Sistema Gerenciador de Banco de Dados Geográficos (SGBDG) possui mecanismos que permitem a manipulação de dados geoespaciais, geralmente incorporados como extensões responsáveis pelo tratamento da componente espacial (Güting, 1994). Cada SGBDG possui uma arquitetura própria, estabelecendo assim diferentes mecanismos responsáveis pela manipulação dos dados geográficos.

Nesse contexto, o objetivo do trabalho foi a otimização de consultas espaciais utilizadas na identificação das classes temáticas do projeto TerraClass que interceptam determinados municípios no estado do Pará, por meio da implementação de um modelo lógico e físico alternativo em um banco de dados geográficos, da aplicação de técnicas de geoprocessamento e de uma nova estratégia para indexação dos dados.

1.1 Estudo de Caso

Instituída em 1953 pela Lei Nº 1.806 para fins de planejamento econômico da região amazônica, a Amazônia Legal Brasileira abrange cerca de 60% do território do país. Diversos pesquisadores estão envolvidos em projetos que estudam o processo de desflorestamento na região da Amazônia Legal. Entre eles, o Projeto de Monitoramento do Desmatamento na Amazônia Legal por Satélite (PRODES) tem por objetivo realizar o monitoramento do desmatamento da Amazônia Legal, produzindo taxas anuais de desmatamento da região desde 1988 (Câmara et al., 2006). Essas taxas evoluíram de aproximadamente 17 mil km² na década de 1990 para aproximadamente 27 mil km² nos anos 2000. No período entre os anos de 2004 e 2012, as taxas anuais de desmatamento diminuíram, sendo registrada em 2012 a menor taxa histórica de aproximadamente 4 mil km². Nesse processo, os estados do Mato Grosso e Pará

responderam por aproximadamente 67% do desmatamento monitorado no período compreendido entre os anos de 1988 e 2015 (INPE, 2016a).

A partir das áreas de desflorestamento da Amazônia Legal que foram mapeadas pelo PRODES, iniciou-se em 2008 outra iniciativa denominada TerraClass, que tem como objetivo qualificar o desflorestamento da Amazônia Legal apresentando resultados de mapeamento do uso e cobertura da terra para todas as áreas desflorestadas (Almeida, 2016). Cinco mapeamentos do TerraClass foram realizados desde então (2004, 2008, 2010, 2012 e 2014), gerando como resultado para cada ano, mapas e dados numéricos que estão disponíveis ao público em arquivos digitais através de sítio eletrônico (INPE, 2016b).

2. Metodologia de Trabalho

Para a realização dos experimentos foi utilizado o PostGIS 2.1 (Obe e Hsu, 2015), uma extensão espacial do SGBD objeto-relacional PostgreSQL 9.3 de código aberto sob a licença Berkeley Software Distribution (BSD). No âmbito do ambiente computacional, foi configurado um sistema virtual Ubuntu de 64 bits através do software Oracle VirtualBox. Nesse ambiente, foram disponibilizados 10GB de memória RAM e processador Intel® Core™ i7-4770 CPU @ 3.40GHz de 4 núcleos físicos.

O processo de alimentação de um SGBDG PostGIS ocorre a partir de um banco criado no PostgreSQL. Os dados, originalmente no formato vetorial *shapefile*, podem ser carregados para o banco através da ferramenta *shp2pgsql* (PostGIS, 2016a). Os principais parâmetros a serem utilizados na ferramenta são:

- o sistema de coordenadas utilizado pelos dados geográficos (parâmetro -s);
- a indicação de criação de índice espacial (parâmetro -I);
- a codificação de caracteres das colunas de atributos (parâmetro -W);
- o nome do *shapefile* a ser carregado no SGBDG PostGIS;
- o esquema e a tabela onde serão armazenados dados geográficos.

Ao final do processo, para cada arquivo carregado no PostGIS é criada uma tabela que contém, entre outras, uma coluna do tipo espacial. Nessa etapa de criação do banco de dados para a realização dos experimentos, foram carregados no PostGIS os dados referentes ao mapeamento do TerraClass do ano de 2008 (INPE, 2016b), além do recorte municipal do estado do Pará do ano de 2010 (IBGE, 2016).

O PostGIS oferece uma grande quantidade de recursos para o processamento da componente espacial. Esses recursos viabilizam a criação de um ambiente integrado para o armazenamento e a manipulação tanto de atributos convencionais como espaciais. O *Open Geodata Interpretability Specification* (OpenGIS), uma marca registrada do *Open Geospatial Consortium* (OGC), estabelece padrões e regras para o processamento dos dados espaciais. Entre eles pode-se destacar o *Simple Feature*, que estabelece o Modelo de Objeto de Geometria, as Operações Espaciais e os Sistemas de Coordenadas. No Modelo de Objeto de Geometria, *Geometry Object Model* (GOM), são especificados os tipos de dados espaciais representadas em estruturas geométricas vetoriais através de pontos, linhas e polígonos. O PostGIS segue o padrão *Simple Feature* do OpenGIS, estendendo assim os tipos de dados espaciais de acordo a especificação do GOM. A Tabela 1 apresenta uma lista com os tipos de dados suportados pelo PostGIS.

No documento de declaração do *Simple Feature*, o GOM é tratado como uma classe na linguagem de programação orientada a objetos, onde essas representações de pontos, linhas e polígonos são herdadas da especificação de geometria. Com isso, mecanismos de *Query* e *Analysis* atuam, respectivamente, como operações espaciais em consultas de intersecção e tarefas de análise de distância, por exemplo.

Tabela 1. Tipos de dados suportados pelo PostGIS.

Nome	ST_GeometryType	Dimensões suportadas
Point	ST_Point	2
Polygon	ST_Polygon	
PolygonWithHole	ST_Polygon	
Collection	ST_GeometryCollection	
Linestring	ST_LineString	

Fonte: PostGIS (2016b).

As operações espaciais são realizadas por meio de extensões funcionais em SQL. Os operadores podem ser de dois tipos: topológico e de distância. Além disso, são disponibilizadas funções espaciais que retornam geometrias ou dados numéricos pela comparação ou medição de uma geometria. A Tabela 2 resume alguns dos operadores disponíveis no PostGIS.

Tabela 2. Operadores espaciais disponíveis no PostGIS.

Tipos de operadores		Funções espaciais			
Topológicos		Distância	Retornam geometria		Retornam valor numérico
ST_Touches	ST_Disjoint	ST_DWithin	ST_Union	ST_Buffer	ST_Length
ST_Covers	ST_CoveredBy		ST_Difference	ST_Centroid	ST_Area
ST_Equals	ST_Overlaps		ST_Intersection	ST_ConvexHull	ST_Distance
ST_Contains	ST_Crosses				

Fonte: PostGIS (2016b)

No escopo desse trabalho, as funções *ST_Intersects* e *ST_Intersection* foram utilizadas na implementação da consulta espacial, responsável pelo cálculo que determina as áreas classificadas pelo TerraClass que fazem parte de cada município do estado do Pará.

3. Resultados e Discussão

Normalmente, os produtos gerados em projetos de mapeamento de uso e cobertura da terra a partir de imagens de satélite são produzidos por SIG e armazenados no formato *shapefile*. Essa é uma técnica amplamente difundida e os arquivos que compõem um *shapefile* seguem uma especificação definida originalmente pela *Environmental Systems Research Institute* (ESRI). No entanto, para o compartilhamento e a distribuição eficiente através da Web, os dados podem ser carregados em um banco de dados geográficos. Para isso, os SGBDG disponibilizam recursos para o processamento da componente espacial. O diagrama da Figura 1 ilustra a arquitetura básica que um SGBDG deve atender.

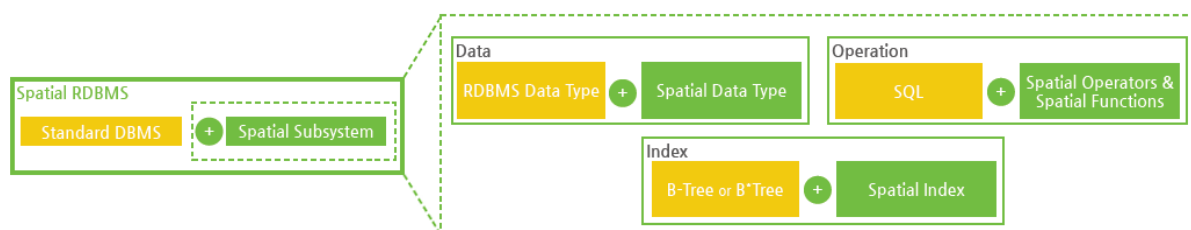


Figura 1. Arquitetura de um SGBDG (Adaptado de Cubrid, 2012).

Nesses casos são utilizados tipos de dados comuns em SQL, como *Integer* (dados numéricos) e *Varchar* (cadeia de caracteres), além de tipos de dados geoespaciais, como pontos, linhas e polígonos. Para o processamento dos índices, um SGBD tradicional utiliza séries *B-tree*, de forma a determinar o tamanho e a redundância dos valores das colunas. Esses

índices são unidimensionais e os tipos de dados espaciais requerem processamentos de dados bidimensionais e tridimensionais. Dessa forma, em um SGBDG são utilizadas séries *R-tree* ou *Quadtree* para o processamento dos índices e assim, dados geoespaciais de até quatro dimensões podem ser indexados.

Em índices *R-tree* ou *Quadtree*, é utilizada a técnica do retângulo envolvente mínimo, *Minimum Bounding Rectangle* (MBR), que determina o menor retângulo que contém a geometria. Dessa forma, os cálculos nas operações espaciais são simplificados, reduzindo uma geometria complexa para uma representação geométrica simplificada, com quatro vértices e quatro arestas. A Figura 2 ilustra a técnica do retângulo envolvente utilizada nos diversos tipos de geometrias existentes.

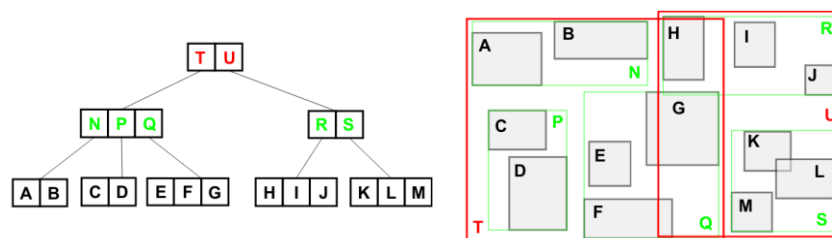


Figura 2. Técnica do retângulo envolvente (Adaptado de Guttman, 1984).

Os dados do TerraClass 2008 do estado do Pará foram carregados no PostGIS seguindo a premissa exposta na Figura 2, gerando uma tabela com 1.270.488 registros que representam os polígonos de cada uma das classes mapeadas pelo projeto. Da mesma forma, a importação dos dados referentes ao recorte municipal do estado do Pará gerou uma tabela com 143 registros. A partir desses dados e seguindo uma abordagem tradicional de utilização de banco de dados geográficos, foi implementada a consulta espacial ilustrada na Figura 3.

```
SELECT      nm_municip, tc_2008, ST_Area(geom) area, geom
FROM        (SELECT
            nm_municip, tc_2008,
            ST_Intersection(tc.geom, mue.geom) geom
            FROM      tc_pa_2008_pol tc, pa_15mue250gc_sir mue
            WHERE     ST_Intersects(tc.geom, mue.geom)
            AND nm_municip LIKE 'PAR%') q_intersec;
```

Figura 3. Declaração SQL implementada para a consulta espacial.

Os municípios do estado do Pará (tabela *pa_15mue250gc_sir*) que começam com cláusula “PAR” e que são interceptados pelos polígonos do TerraClass (tabela *tc_pa_2008_pol*), são identificados através função *ST_Intersects*, especificada na cláusula *where* da declaração SQL da Figura 3. Sem seguida, a função *ST_Intersection*, declarada na cláusula *select*, retorna a geometria resultante dessa intersecção. A consulta foi executada em aproximadamente 7 minutos, retornando um total de 54.093 registros.

Essa abordagem representa um alto custo de processamento e, conseqüentemente, um tempo excessivo de resposta. Essas condições inviabilizam sua utilização em um ambiente Web, onde vários usuários realizam diversas operações de consulta simultaneamente. Por se tratar de um software de código aberto, uma opção para otimizar o tempo de resposta das consultas espaciais é a revisão dos algoritmos implementados nos operadores espaciais disponíveis no PostGIS. No entanto, esse é um esforço que demanda tempo e o planejamento adequado das tarefas envolvidas devido, principalmente, às características multidisciplinares envolvidas no problema. Dessa forma, para otimizar as consultas no SGBDG PostGIS foram propostas as técnicas de modelagem, implementação e indexação que estão desmembradas nas seções 3.1 e 3.2.

3.1 Redefinição do Modelo de Dados

O processo de criação e alimentação de um banco de dados geográficos possui etapas bem definidas e as técnicas empregadas para sua execução estão consolidadas entre os usuários-produtores e consumidores desses dados. Entretanto, a utilização de técnicas adicionais de modelagem pode ser muito eficiente para a otimização de consultas complexas. De forma a aprimorar o desempenho da consulta apresentada na Figura 3, foi definida a implementação de um modelo de dados alternativo. A Figura 4 ilustra o modelo proposto, estabelecendo um relacionamento de associação simples (Borges et al., 2001) entre as tabelas que representam o mapeamento do TerraClass e os municípios do estado do Pará.

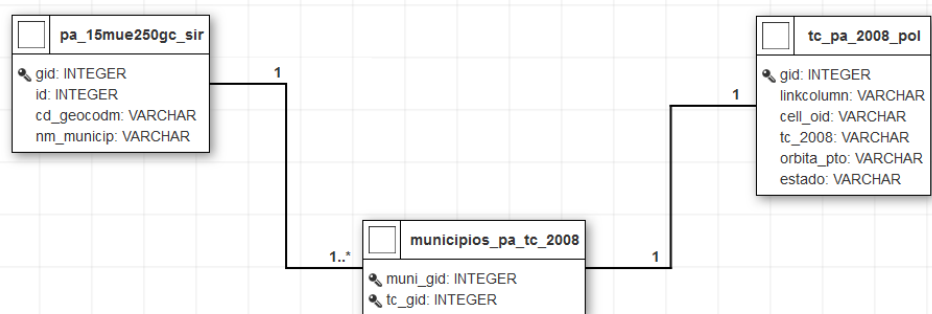


Figura 4. Relacionamento entre as tabelas do TerraClass e municípios do estado do Pará.

Essa é uma técnica extremamente difundida em banco de dados relacionais. Entretanto, a principal modificação em relação à abordagem tradicional é a criação da coluna “geom”, onde o resultado da consulta espacial que faz uso da função *ST_Intersection* é previamente armazenado, capturando dessa forma a semântica dos dados geográficos (Borges et al., 2001). Assim, os cálculos necessários para estabelecer a intersecção entre as classes do TerraClass e os municípios do estado do Pará é realizado antecipadamente, durante o processo de alimentação da nova tabela. Com isso, os processos efetuados pelos operadores espaciais para realização desses cálculos são preservados e, mesmo representando um alto custo para o banco de dados, sua execução ocorre em segundo plano, não impactando na realização das consultas que são efetuadas em tempo real pelos usuários.

Essa solução representou o principal fator para o ganho de desempenho na execução das consultas. O tempo obtido anteriormente de 7 minutos, foi reduzido para aproximadamente 5 segundos, uma otimização superior a 80 vezes em relação ao tempo inicial.

3.2 Aplicação de Índices

O PostgreSQL disponibiliza diferentes tipos de índices, que podem ser utilizados em colunas que não compõem a chave primária da tabela. Índices espaciais do tipo *GiST*, que implementam a série *R-tree* para o seu processamento, são aplicados nas colunas das tabelas que possuem dados geográficos. Por outro lado, colunas com tipo de dados convencionais podem receber índices do tipo *Hash* ou do tipo *B-tree* (PostGIS, 2016a). Esses índices foram criados na coluna do tipo texto que possuem os nomes dos municípios. No caso do índice *Hash*, o PostgreSQL optou pela busca sequencial na tabela (*Sequential Scan*), uma vez que apenas comparações que envolvem o operador de igualdade habilitam a utilização desse tipo de índice. Para o índice *B-tree*, o PostgreSQL considera sua utilização em comparações que envolvem todos os tipos de operadores, inclusive “*Between*” e “*In*”.

Para o seu funcionamento com o operador “*LIKE*”, é necessário que a cláusula envolvida na comparação seja uma constante e esteja ancorada no início da *string* (cadeia de caracteres). A declaração SQL da Figura 3 atende a essa condição, onde foi implementada a cláusula para a busca de todos os municípios com nomes que começam com a sequência “*PAR*”. No

entanto, o PostgreSQL novamente optou por uma busca sequencial na tabela, conforme ilustrado na linha 3 da Figura 5.

	QUERY PLAN text
1	Nested Loop (cost=327.26..98642.67 rows=9021 width=952) (actual time=116.569..1815.951 rows=54093 loops=1)
2	-> Nested Loop (cost=326.84..27798.64 rows=9021 width=938) (actual time=77.527..798.515 rows=54093 loops=1)
3	-> Seq Scan on pa_15mue250gc_sir_mue (cost=0.00..42.79 rows=1 width=16) (actual time=0.045..0.118 rows=2 loops=1)
4	Filter: ((nm_municip)::text ~ 'PAR%':::text)
5	Rows Removed by Filter: 141
6	-> Bitmap Heap Scan on municipios pa_tc_2008 tc_mue (cost=326.84..27665.00 rows=9085 width=930)
7	Recheck Cond: (muni_gid = mue_gid) (actual time=38.933..328.879 rows=27046 loops=2)
8	-> Bitmap Index Scan on municipios pa_tc_2008 pkey (cost=0.00..324.56 rows=9085 width=0)
9	Index Cond: (muni_gid = mue_gid) (actual time=17.982..17.982 rows=27046 loops=2)
10	-> Index Scan using tc_pa_2008_pol_pkey on tc_pa_2008_pol_tc (cost=0.43..7.84 rows=1 width=22)
11	Index Cond: (gid = tc_mue.tc_gid) (actual time=0.010..0.011 rows=1 loops=54093)
12	Total runtime: 1873.649 ms

Figura 5. Exemplo de plano de execução utilizado pelo PostgreSQL.

Para que o otimizador de consultas do PostgreSQL utilize o índice *B-tree* com o operador “LIKE”, é necessário especificar a opção “*varchar_pattern_ops*” (PostgreSQL, 2016a). Nessa condição, ilustrada na Figura 6, um novo índice será criado estabelecendo assim uma estratégia de combinação de índices *B-tree*.

```
CREATE INDEX pa_15mue250gc_sir_nm_municip_idx
ON public.pa_15mue250gc_sir
USING btree (nm_municip COLLATE pg_catalog."default" varchar_pattern_ops);
```

Figura 6. Declaração SQL para criação de índice *B-tree* personalizado.

Nessa estratégia, os índices *Bitmaps* são criados dinamicamente na memória e são preparados a partir de informações das tabelas que atendam às condições dos índices disponíveis (PostgreSQL, 2016a). A Figura 5 ilustra essa condição nas linhas 5 e 7. Na situação anterior, após a redefinição do modelo de dados, o resultado da consulta obtido em 5 segundos gerou um custo total de transação de aproximadamente 1.816. Esse custo representa um número arbitrário, mas que convencionalmente indica a quantidade de páginas de disco que foram retornadas (PostgreSQL, 2016b). O custo total da transação após aplicação da estratégia de combinação de índices ficou próximo de 900, resultando em uma otimização superior a 50% em relação ao valor medido anteriormente.

4. Conclusões

Os experimentos realizados nesse trabalho apresentaram resultados bastante expressivos. A implementação de um novo modelo lógico e físico no banco de dados contribuiu de forma significativa para a otimização das consultas.

Da mesma forma, a definição adequada da estratégia de indexação das tabelas e o processamento prévio de consultas espaciais, permitiram a redução no tempo total para recuperação dos dados, especialmente no âmbito do custo total de transações. Esse pode ser considerado um fator muito representativo em um cenário com ambiente multiusuário, no qual o SGBDG deve ser capaz de gerenciar várias requisições e respostas simultâneas.

Em trabalhos futuros, uma análise recomendada é a avaliação das técnicas de coleta de estatísticas disponíveis no PostgreSQL, um recurso onde são coletados dados sobre as atividades do SGBD que podem auxiliar na definição de novos índices.

Referências Bibliográficas

Alesheikh, A. A.; Helali, H.; Behroz, H. A. Web GIS: technologies and its applications. In: **Symposium on geospatial theory, processing and applications**. 2002.

- Almeida, C. A.; Coutinho, A. C.; Esquerdo, J. C. D. M.; Adami, M.; Venturieri, A.; Diniz, C. G.; Dessay, N.; Durieux, L.; Gomes, A. R. High spatial resolution land use and land cover mapping of the Brazilian Legal Amazon in 2008 using Landsat-5/TM and MODIS data. **Acta Amazonica**, Vol 46 (3) 2016: 291-302.
- Borges, K. A. V.; Davis, C. A.; Laender, A. H. F. OMT-G: na object-oriented data model for geographic applications. **GeoInformatica**, v. 5, n. 3, p. 221-260, 2001.
- Câmara, G.; Valeriano, D. M.; Soares, J. V. **Metodologia para o cálculo da taxa anual do desmatamento na Amazônia Legal**. São José dos Campos: INPE, 2006. 24p. Disponível em: <<http://www.obt.inpe.br/prodes/metodologia.pdf>>. Acesso em: 31.out.2016.
- Cubrid. **Open Source RDBMS - 20 Minutes to Understanding Spatial Database**. 2012. Disponível em: <<http://www.cubrid.org/blog/dev-platform/20-minutes-to-understanding-spatial-database/>>. Acesso em: 20.out.2016.
- Ceri, S.; Bongio, A.; Fraternali, P.; Brambilla, M.; Comai, S.; Matera, M. Morgan Kaufmann series in data management systems: **Designing data-intensive Web applications**. Morgan Kaufmann, 2003.
- Egenhofer, M.; Franzosa, R. Point-Set Topological Spatial Relations. **International Journal of Geographical Information Systems**, v. 5, n.2, p. 161-174, 1991.
- Fu, P.; Sun, J. **Web GIS: principles and applications**. Esri Press, 2010.
- Guttman, A. **R-trees: a dynamic index structure for spatial searching**. ACM, 1984.
- Güting, R. H. An introduction to spatial database systems. **The VLDB Journal—The International Journal on Very Large Data Bases**, v. 3, n. 4, p. 357-399, 1994.
- Instituto Brasileiro de Geografia e Estatística (IBGE). **Bases cartográficas**. 2016. Disponível em: <<http://mapas.ibge.gov.br/bases-e-referenciais/bases-cartograficas/malhas-digitais.html>>. Acesso em: 20.set.2016.
- Instituto nacional de pesquisas espaciais (INPE). **Projeto de monitoramento da floresta amazônica brasileira por satélite**. 2016(a). Disponível em: <<http://www.obt.inpe.br/prodes/index.php>>. Acesso em: 06.out.2016.
- Instituto nacional de pesquisas espaciais (INPE). **Dados TerraClass**. 2016(b). Disponível em: <http://www.inpe.br/cra/projetos_pesquisas/dados_terraclass.php>. Acesso em: 06.out.2016.
- Jensen, J. R. **Sensoriamento remoto do ambiente: uma perspectiva em recursos terrestres**. Tradução J. C. N. Epiphany et. al. 3. ed. São José dos Campos: Editora Parêntese, 2009. 604 p.
- Longley, P. **Geographic information systems and science**. John Wiley & Sons, 2005.
- Mather, P. M. **Computer Processing of Remotely-Sensed Images: An Introduction**. 3rd ed. New York: John Wiley, 2004.
- Obe, R. O.; Hsu, L. S. **PostGIS in action**. Manning Publications Co., 2015.
- PostGIS. **Boundless - introduction to PostGIS**. 2016(b). Disponível em: <<http://workshops.boundlessgeo.com/postgis-intro/>>. Acesso em: 28.out.2016.
- PostGIS. **PostGIS manual**. 2016(a). Disponível em: <<http://postgis.net/docs/>>. Acesso em: 30.set.2016.
- PostgreSQL. **PostgreSQL documentation**. 2016(a). Disponível em: <<https://www.postgresql.org/docs/9.3/static/>>. Acesso em: 30.set.2016.
- PostgreSQL. **PostgreSQL documentation - explain**. 2016(b). Disponível em: <<https://www.postgresql.org/docs/9.3/static/sql-explain.html>>. Acesso em: 18.out.2016.
- Ramakrishnan, R.; Gehrke, J. **Database management systems**. McGraw-Hill, 2000.
- Shimabukuro, Y. E.; Batista, G. T.; Mello, E. M. K.; Moreira, J. C.; Duarte, V. Using shade fraction image segmentation to evaluate deforestation in Landsat Thematic Mapper images of the Amazon region. **International Journal of Remote Sensing**, v. 19, n. 3, p. 535-541, 1998.
- Tomlin, C. D. **Geographic information systems and cartographic modeling**. Prentice Hall, 1990.
- Valeriano, D. M. et al. Monitoring tropical forest from space: the PRODES digital project. **International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences**, v. 35, p. 272-274, 2004.