Uso de índices

Noemi Pereira Scherer, Higor Celante

Universidade Tecnológica Federal do Paraná (UTFPR)
Campo Mourão – Paraná - Brasil

Banco de Dados 2

{Noemi,Higor} noemischerer13@gmail.com, higor.celante@gmail.com

Abstract. *This work consists of using a database known as Dataset, analyzing and dividing the database, creating tables in the corresponding database, creating a code in which to pass each record to the table in the database. The goal is to create queries, and analyze performance using indexes and not using indexes.*

Resumo. *Esse trabalho consiste em utilizar uma base de dados conhecida como Dataset, analisar e dividir a base, criar tabelas no banco correspondente a base de dados, criar um código no qual passe cada registro para a tabela no banco de dados. O objetivo é criar consultas, e analisar o desempenho utilizando índices e não utilizando índices.*

1. Introdução

Índice é uma referência associada a uma chave, que é utilizada otimizar, permitindo uma localização mais rápida de um registro quando efetuada uma consulta. Um índice é uma estrutura (ou arquivo) auxiliar associado a uma tabela (ou coleção de dados). Sua função é acelerar o tempo de acesso às linhas de uma tabela, criando ponteiros para os dados armazenados em colunas específicas. O banco de dados usa o índice de maneira semelhante ao índice remissivo de um livro, verifica um determinado assunto no índice e depois localiza a sua posição em uma determinada página [1].

2. Objetivo

O objetivo desse trabalho é através de um dataset qualquer, com pelo menos 20GB de tamanho, analisar o desempenho de consultas dos dados em um banco de dados utilizando e não utilizando índices, para verificar se o uso de índices, quando bem aplicado, realmente melhora o desempenho da busca no banco.

3. Materiais e Métodos

- Banco de dados POSTGRESQL;
- Dataset com tamanho de 23 GB, e 70 milhões de dados: Link de Download;
- Conversor de dados: Link para conversor de dados;
- Dez consultas complexas;

- Índices: Sintaxe dos índices criados;
- Tabela relacional do dataset: Acesso a tabela relacional.

4.Dataset

O dataset é obtido no site spatialhadoop, chamado de Edges. Os dados são arestas do conjunto de Tiger do ano 2015. Os conjuntos de dados TIGER são todos extraídos dos arquivos TIGER do Census Bureau dos EUA. Os dados são convertidos de *Shapefiles* para CSV para serem facilmente processados. Cada linha no arquivo CSV contém uma forma representada no formato WKT (Well-Known Text) seguido por outras informações meta para este registro.

Basicamente, os dados são características de arestas de um conjunto Tiger. Ele possui 32 campos, no qual é possível dividir em 32 tabela e realizar associação entre elas. Para o significado de cada campo, pode-se encontrar em um documento obtido no site EUA census Bureau, o documento extraído encontra-se no diretório do GitHub, junto ao *script* de conversor de dados. Os dados para arestas são extraídos de diversos lugares dos EUA, como mostra a Figura 1.

Todos os campos são relacionados com arestas extraídas de lugares, no qual os dados são armazenados em uma figura geométrica com várias faces, os significado de cada campo são:

1. WKT: São um sistema de coordenadas. Essas coordenadas define a forma da figura geométrica em que se encontra todos os dados das arestas;
2. STATEFP: É um estado pelo ID. No sistema Tiger, eles representando cada estado com um id diferente;
3. COUNTYFP: Código FIP;
4. TLID: ID da borda que nunca é alterada;
5. TFIDL: ID da borda permanente do lado esquerdo da face;
6. TFIDR: ID da borda permanente do lado direito da face;
7. MTFCC: Código principal da face;
8. FULLNAME: Nome completo do lugar onde está sendo extraída as arestas;
9. SMID: Metadados especiais;
10. LFROMADD: Número do local que está sendo extraída as arestas a partir de um endereço já existe. O número é localizado no lado esquerdo da borda;
11. LTOADD: Endereço do local que fica no lado esquerdo da borda;
12. RFROMADD: Número do local que está sendo extraída as arestas a partir de um endereço já existe. O número é localizado no lado direito da borda;
13. RTOADD: Endereço do local que fica no lado direito da borda;
14. ZIPL: Código zip do lado esquerdo da borda;

15. ZIPR: Código zip do lado direito da borda;
16. FEATCAT: Se possui características gerais;
17. HYDROFLG: Se possui algum indicador de recursos hidráulicos;
18. RAILFLG: Se possui algum indicador de recurso ferroviário;
19. ROADFL: Se possui indicador de recursos de estrada;
20. OLFFLG: Se possui algum indicador de características linear, ou seja, figuras com uma dimensão;
21. PASSFLG: Flag para dados espaciais;
22. DIVROAD: Flags de estradas divididas;
23. EXTTYP: Tipos de extensões;
24. TTYP: Tipos de pacotes;
25. DECKEDROAD: Se possui indicador de estrada pavimentada;
26. ARTPATH: Indicadores de pacotes artificiais;
27. PERSIST: Flag para persistências geográficas;
28. GCSEFLG: Flag para corredores geográficos;
29. OFFSETL: Flag para o lado esquerdo da borda;
30. OFFSETR: Sinalizador de deslocamento para face do lado direito de determinada aresta;
31. TNIDF: Nó que não muda que fica no começo;
32. TNID: Nó que não muda que fica na borda.

Figura 1: Lugares que foram extraído as arestas.

5. Tabela relacional

Cada campo do dataset edges é criado uma tabela e ralações entre elas. A tabela Fullname está relacionada com quase todos as outras tabelas, isso é feito para que a

consulta possa ser melhor realizada. A tabela tfidl representa o ID da borda permanente do lado esquerdo da face, por isso ela está relacionada com todas as tabelas que representam borda do lado esquerdo, como, lfromadd, ltoadd, zipl. Essa mesma ideia é aplicada a tabela tfidr, que relaciona com todas as outras tabelas nos quais idicam o lado direito da face, como, rfromadd, rtoadd e zipr.

A tabela tlid se relaciona com todas as tabelas que possuem bordas que não são alteradas, tnidf, tnidt, tfidl, tfidr. Assim como a tabela featcat, que se relaciona com todas as tebelas que representam características gerais, hydroflg, ralflg, roadflg, deckedroad e artpath.

A tabela olfflg se relaciona com as tabelas que possuem características lineraes, como, divroad, passflg e persist.

Uma noção breve pode ser vista na Figura 2. Para visualizar melhor, acesse Tabela Relacional.

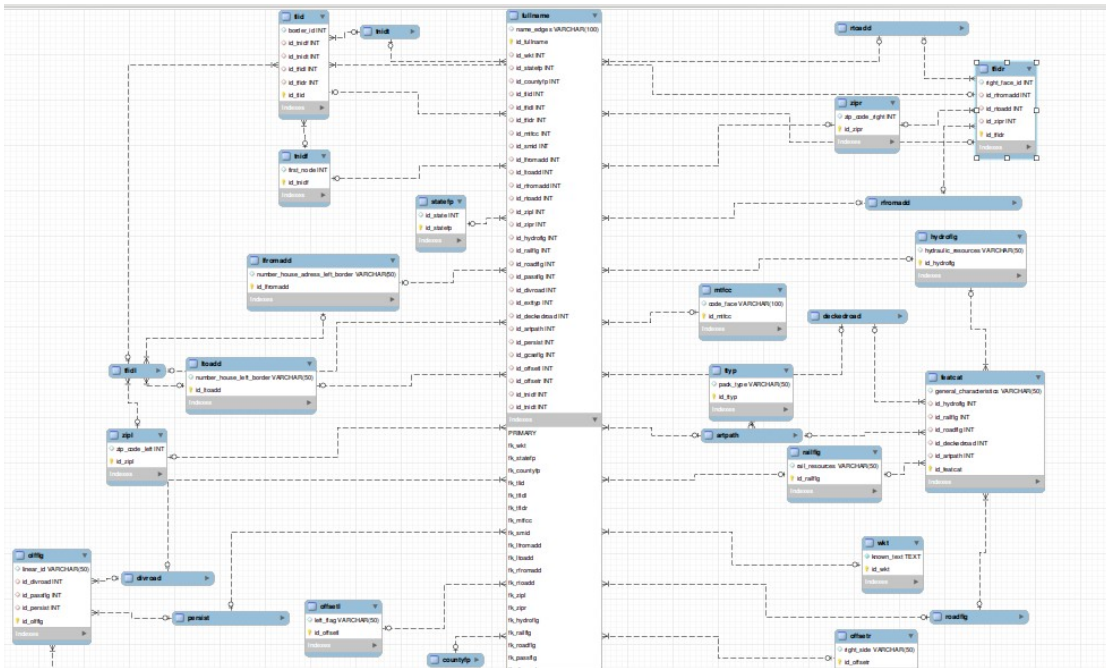

Figura 2: Tabela relacional dos dados Edges.

6.Procedimentos e Resultados

É utilizado o código em Java para inserir os dados no banco. A inserção foi dividida em duas partes: primeiro é inserido todos os registros, exceto da tabela fullname, pois essa inserção teve que ser modificada por possuir caracteres com acentos. Devido a demora da inserção, de 70 milhões de dados, foram inseridos um pouco mais de 300.000 registros por tabela. Já a tabela fullname, como foi a última a ser inserida, foram armazenados mais de 10 milhões de dados no banco, o que deixou a consulta mais demorada.

Para testar o desempenho das consultas sem o uso de índices, são aplicadas dez consultadas ao banco, analisado o tempo total de processamento e quantidade de registros retornados. Os resultados obtidos se encontram na Tabela 1.

- Consulta 1: Listar os nomes, coordenadas, o id do estado, código FIP e o ID da borda permanente.

```
SELECT name_edges, id_wkt, id_statefp, id_countyfp, id_tlid
FROM fullname;
```

- Consulta 2: Listar os códios FIP, nó permante do começo, nó permante para uma borda, id da face esquerda da borda permanente e id da face direita da borda permanente.

```
SELECT border_id, id_tnidf, id_tnidt, id_tfidl, id_tfidl
FROM tlid;
```

- Consulta 3: Ordenar os resultados através do nome.

```
SELECT name_edges
FROM fullname
GROUP BY name_edges;
```

- Consulta 4: Listar os nomes e as coordenadas.

```
SELECT f.name_edges, f.id_fullname, w.known_text
FROM fullname as f
INNER JOIN wkt as w
ON (f.id_fullname = w.id_wkt);
```

- Consulta 5: Juntar todos os nomes com os pacotes

```
SELECT id_fullname, name_edges FROM fullname
UNION
SELECT id_ttyp, pack_type FROM ttyp;
```

- Consulta 6: Unir e listar as características gerais e as coordenadas

```
SELECT general_characteristics
FROM fullname as f, railflg as r, featcat as fe
WHERE f.id_fullname = r.id_railflg and f.id_railflg = fe.id_featcat
UNION
SELECT w.known_text
FROM fullname as f, wkt as w
```

WHERE f.id_fullname = w.id_wkt;

- Consulta 7: Juntar tos os ids de pacotes artificiais com características gerais.

  SELECT w.known_text

  FROM wkt **as** W, fullname **as** f

  WHERE f.id_fullname = w.id_wkt;

- Consulta 8: Selecionar os ids dos recursos hidráulico, recursos ferroviários, recursos de estradas, id das estradas pavimentadas e o id do pacote artificial.

  SELECT id_hydroflg, id_railflg, id_roadflg, id_deckedroad, id_artpath

  FROM fullname;

- Consulta 9: Selecionar os ids dos primeiros nó.

  SELECT id_tnidf

  FROM fullname ;

- Consulta 10: Selecionar os ids do nome completo, os nomes completo e juntar com o id das coordenadas mais as coordenadas.

  SELECT id_fullname, name_edges FROM fullname

  UNION

  SELECT id_wkt, known_text FROM wkt;

	Tempo total (ms)	Tempo total aproximado (minutos)	Quantidade de registros retornados
Consulta 1	296261 ms	4,94 min.	6082403 linhas, 30412015 registros
Consulta 2	11086 ms	Menos de um minuto	226500 linhas, 113200 registros
Consulta 3	70762 ms	1,18 min.	306009 registros
Consulta 4	72457 ms	1,21 min.	226500 linhas, 670500 registros
Consulta 5	234201 ms	3,9 min.	6191418 linhas, 12382836 registros
Consulta 6	79478 ms	1,32 min.	217566 registros
Consulta 7	66101 ms	1,1 min.	226500 registros

Consulta 8	243898 ms	4,06 min.	6082403 linhas, 30 412015 registros
Consulta 9	91167 ms	1,52 min.	6082403 registros
Consulta 10	304997 ms	5,08 min.	6308903 linhas, 12617806 registros

Tabela 1: Resultados das consultas no banco de dados sem utilizar índices.

São aplicados os índices em cada tabela para tentar melhorar o desempenho da consulta, os índices criados pode ser acessados em índices e os resultados obtidos encontram-se na Tabela 2.

	Tempo total (ms)	Tempo total aproximado (minutos)	Quantidade de registros retornados
Consulta 1	296069 ms	4 min.	6082403 linhas, 30412015 registros
Consulta 2	11005 ms	Aproximadamente 6 segundos	226500 linhas, 113200 registros
Consulta 3	71996 ms	1,19 min.	306009 registros
Consulta 4	72203 ms	1,20 min.	226500 linhas, 670500 registros
Consulta 5	234457 ms	3,9 min.	6191418 linhas, 12382836 registros
Consulta 6	70442 ms	1,17 min.	217566 registros
Consulta 7	66070 ms	1,1 min.	226500 registros
Consulta 8	242750 ms	4,04 min.	6082403 linhas, 30 412015 registros
Consulta 9	91100 ms	1,51 min.	6082403 registros
Consulta 10	299701 ms	4,99 min.	6308903 linhas, 12617806 registros

Tabela 2: Resultados das consultas no banco de dados utilizando índices.

7.Conclusão

O uso de índices nas consultas melhoram o desempenho da busca, quando utilizado o índice correto. Em um único caso isso não ocorreu, mas o motivo foi pela escolha do índice (Consulta) *b-tree* e o uso do GROUP BY, no qual o uso de outro índice ou consulta favoreceria o resultado.

8. Referências

[1] Definição de um índice: https://pt.wikipedia.org/wiki/%C3%8Dndice_(estruturas_de_dados).

[2] Materiais do trabalho: https://github.com/noemis13/Conversor-de-dados.