Banco de Dados

BD Geográficos

Valéria Times

vct@cin.ufpe.br

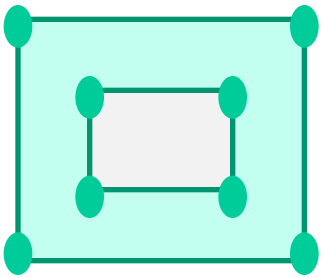
INTRODUÇÃO AO POSTGIS

Introdução ao PostGIS

- PostGIS: Uma extensão Geo para o PostgreSQL
  - Download
    - http://postgis.net/install
  - Manual
    - http://postgis.net/documentation

Introdução ao PostGIS

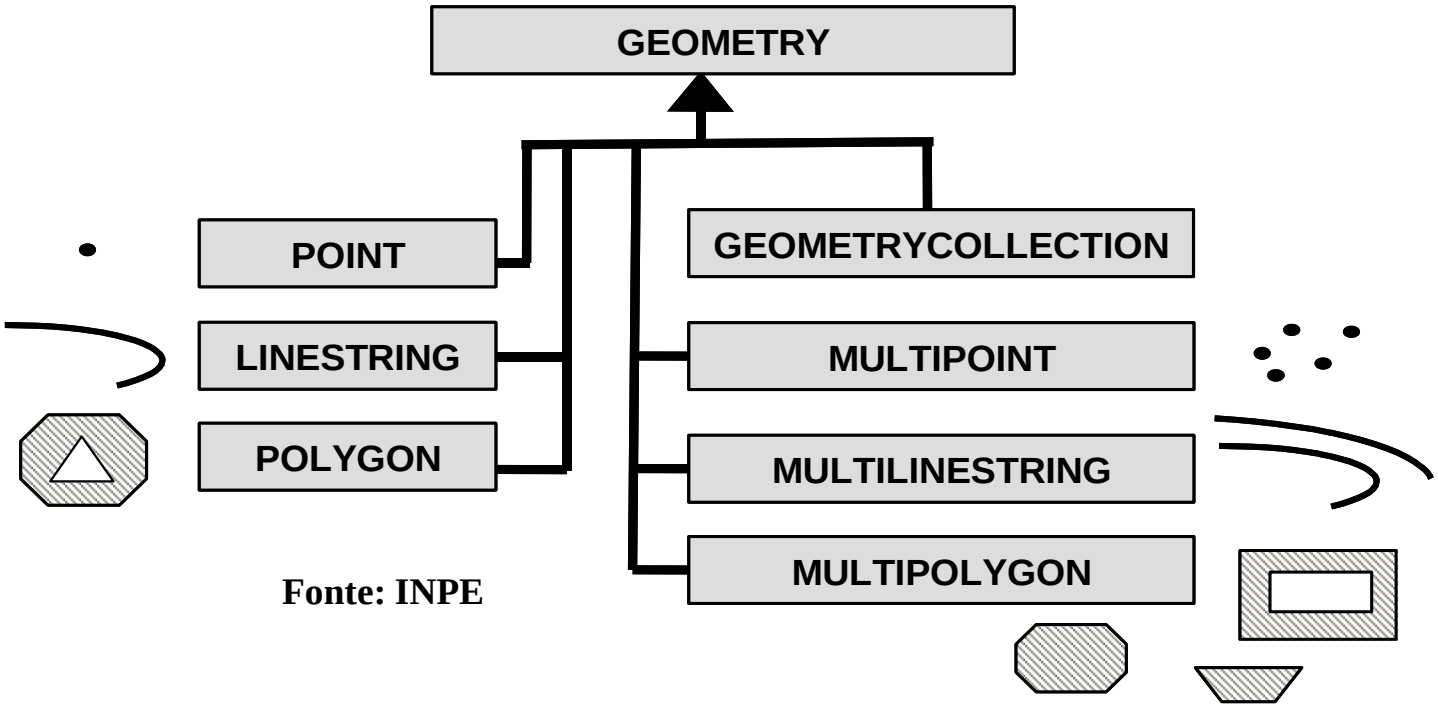
- O que é PostGIS?
  - Um novo tipo de dado para o Postgres
    - Geometry



  - Novas Funções sobre o tipo Geometry
    - ST_Distance(geometry, geometry)
    - ST_Area(geometry)
    - ST_Intersects(geometry, geometry)
    - …
  - Mecanismo de indexação p/ consultas espaciais

Introdução ao PostGIS

- PostGIS segue o padrão OpenGIS
  - Provê suporte para todos objetos e funções da especificação SFS (Simple Features for SQL)



Fonte: INPE

Introdução ao PostGIS

- A especificação SFS/OpenGIS define tipos, funções e metadados para manipular ObjetosGeo

- A principal tabela de metadados do OGC é:
  - SPATIAL_REF_SYS → guarda os IDs e as descrições textuais do sistema de coordenadas usados no BDGeo

Tabela: SPATIAL_REF_SYS			
srid	INTEGER	PK	identificador do SRS
auth_name	VARCHAR(256)		nome da autoridade que especificou o SRS
auth_srid	INTEGER		identificador do SRS definido pela autoridade
srtext	VARCHAR(2048)		representação WKT do SRS
proj4text	VARCHAR(2048)		especificações para transformação de SRS

Introdução ao PostGIS

- SRID (Spatial Referencing System Identifier)
  - Todo Objeto Geográfico deve ter um SRID para ser inserido no BDGeo

	srid [PK] integer	auth_name character varying(256)	auth_srid integer	srtext character varying(2048)	proj4text character varying(2048)
1	2000	EPSG	2000	PROJCS["Anguilla 19	+proj=tmerc +lat_0=
2	2001	EPSG	2001	PROJCS["Antigua 194	+proj=tmerc +lat_0=
3	2002	EPSG	2002	PROJCS["Dominica 19	+proj=tmerc +lat_0=
4	2003	EPSG	2003	PROJCS["Grenada 195	+proj=tmerc +lat_0=
5	2004	EPSG	2004	PROJCS["Montserrat 1	+proj=tmerc +lat_0=
6	2005	EPSG	2005	PROJCS["St. Kitts 19	+proj=tmerc +lat_0=
7	2006	EPSG	2006	PROJCS["St. Lucia 19	+proj=tmerc +lat_0=
8	2007	EPSG	2007	PROJCS["St. Vincent	+proj=tmerc +lat_0=
9	2008	EPSG	2008	PROJCS["NAD27(CGQ77)	+proj=tmerc +lat_0=
10	2009	EPSG	2009	PROJCS["NAD27(CGQ77)	+proj=tmerc +lat_0=
11	2010	EPSG	2010	PROJCS["NAD27(CGQ77)	+proj=tmerc +lat_0=
12	2011	EPSG	2011	PROJCS["NAD27(CGQ77)	+proj=tmerc +lat_0=
13	2012	EPSG	2012	PROJCS["NAD27(CGQ77)	+proj=tmerc +lat_0=
14	2013	EPSG	2013	PROJCS["NAD27(CGQ77)	+proj=tmerc +lat_0=
15	2014	EPSG	2014	PROJCS["NAD27(CGQ77)	+proj=tmerc +lat_0=

Introdução ao PostGIS

- Formatos WKB e WKT do OpenGIS
  - Duas formas padrões para manipular Objetos Geográficos
    - Well-Known Text (WKT) e Well-Known Binary (WKB)
    - Guardam informações sobre tipo e coordenadas do ObjetoGeo
  - Exemplos:
    - POINT(0 0)
    - LINESTRING(0 0,1 1,1 2)
    - POLYGON((0 0,4 0,4 4,0 4,0 0),(1 1, 2 1, 2 2, 1 2,1 1))
    - MULTIPOINT(0 0,1 2)
    - MULTILINESTRING((0 0,1 1,1 2),(2 3,3 2,5 4))
    - MULTIPOLYGON(((0 0,4 0,4 4,0 4,0 0),(1 1,2 1,2 2,1 2,1 1)), ((-1 -1,-1 -2,-2 -2,-2 -1,-1 -1)))
    - GEOMETRYCOLLECTION(POINT(2 3),LINESTRING((2 3,3 4)))

UTILIZANDO O POSTGIS

Usando o PostGIS

- Abrindo uma conexão com PostGIS:
  - Servidor: localhost
  - Porta: 5432
  - Banco: postgres
  - Usuário: postgres
  - Senha: postgres

Usando o PostGIS

- Criando uma Tabela Espacial:
  - CREATE TABLE estacoes_pluviometricas (
    gid INT4,
    location GEOMETRY,
    nome VARCHAR(25));

- ERROR: type "geometry" does not exist
- SOLUÇÃO:
  - CREATE EXTENSION postgis;
  - CREATE EXTENSION postgis_topology;
  - CREATE EXTENSION fuzzystrmatch;

Usando o PostGIS

- Inserindo dados em uma Tabela Espacial:
  - INSERT INTO estacoes_pluviometricas
    VALUES (1, 'POINT(-46.98 -19.57)',
    'DINIZ-ARAXA');
  - INSERT INTO estacoes_pluviometricas
    VALUES (2, 'POINT(-43.59 20.37)',
    'QUEIROZ-OURO-PRETO');

Usando o PostGIS

- Recuperando dados de uma Tabela Espacial:
  - SELECT gid, nome, ST_AsText(location)
    FROM estacoes_pluviometricas;
  - Qual o problema com este método?
    - Deixa-se de preencher alguns metadados da tabela que possui uma coluna espacial!
    - Deixa-se de associar um SRID à geometria!

Usando o PostGIS

- Forma correta de criar uma Tabela Espacial:
  - Primeiro Passo:
    - CREATE TABLE estacoes_pluviometricas (
      gid INT4,
      nome VARCHAR(25));
  - Segundo Passo:
    - SELECT AddGeometryColumn('estacoes_pluviometricas',
      'location', 4291, 'POINT', 2) ;

Usando o PostGIS

- Forma correta de inserir a geometria de um dado espacial:

  - INSERT INTO estacoes_pluviometricas
    VALUES (1, 'DINIZ-ARAXA',
    ST_GeomFromText ('POINT(-46.98 -19.57)', 4291)) ;

Geometria expressa em WKT

SRID (Spatial Reference ID)

Usando o PostGIS

- Fazendo a carga de arquivos Shape
  - Arquivo Shape:
    - .shp = contém a parte geométrica
    - .dbf =contém a parte alfa-numérica (string, number, date)
    - .shx = contém dados de índice
  - Tabelas PostgreSQL+PostGIS:
    - Colunas podem conter geometrias
    - Colunas podem conter atributos convencionais
  - Em geral, um arquivo Shape corresponde a uma tabela PostgreSQL+PostGIS

Usando o PostGIS

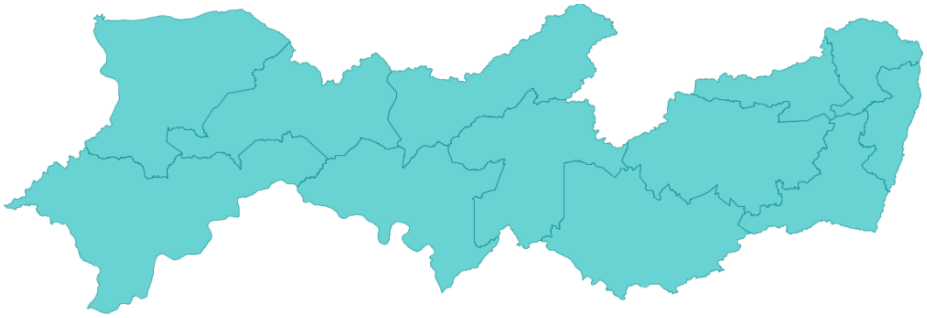
- Exemplos da carga de arquivos Shape:
  - http://www.mapacultural.pe.gov.br/inicial/shapefile.htm
  - Download:
    - Estado de Pernambuco com os municípios
    - Regiões de Desenvolvimento
    - Hidrografia
    - Bens Imateriais

Usando o PostGIS

- Municípios

- Hidrografia

- Regiões



- Bens Imateriais

Usando o PostGIS

- Inserindo ObjetosGeo nas tabelas (2 formas)

1) Usando SQL

```
BEGIN;
INSERT INTO roads (road_id, roads_geom, road_name)
  VALUES (1,ST_GeomFromText('LINESTRING(191232 243118,191108 243242)',2000),'Jeff Rd');
INSERT INTO roads (road_id, roads_geom, road_name)
  VALUES (2,ST_GeomFromText('LINESTRING(189141 244158,189265 244817)',2000),'Geordie Rd');
INSERT INTO roads (road_id, roads_geom, road_name)
  VALUES (3,ST_GeomFromText('LINESTRING(192783 228138,192612 229814)',2000),'Paul St');
INSERT INTO roads (road_id, roads_geom, road_name)
  VALUES (4,ST_GeomFromText('LINESTRING(189412 252431,189631 259122)',2000),'Graeme
Ave');
INSERT INTO roads (road_id, roads_geom, road_name)
  VALUES (5,ST_GeomFromText('LINESTRING(190131 224148,190871 228134)',2000),'Phil Tce');
INSERT INTO roads (road_id, roads_geom, road_name)
  VALUES (6,ST_GeomFromText('LINESTRING(198231 263418,198213 268322)',2000),'Dave Cres');
COMMIT;
```

Usando o PostGIS

- **Inserindo ObjetosGeo nas tabelas (2 formas)**

  2) Usando o Loader shp2pgsql

  - Converte um arquivo shape para pgsql.sql
  - Shp2pgsql [<options>] <shapefile> <tablename> <database name>
    - <shapefile> : nome do shape file s/ extensão (inclui shp, shx, dbf)
    - <tablename> : nome da tabela destino. Por default, a geometria fica na coluna 'geo_value'
    - <database name> : nome do BDGeo destino
    - [<options>] : opções de configuração
      - » http://postgis.net/docs/manual-2.0/using_postgis_dbmanagement.html#idp33173584

Usando o PostGIS

- **Inserindo ObjetosGeo nas tabelas (2 formas)**

  2) Usando o Loader shp2pgsql

  - Exemplo com arquivo intermediário:
    - Abrir um terminal (cmd) e executar:
    - set PATH=%PATH%;C:\Program Files\PostgreSQL\9.4\bin
    - Shp2pgsql -W LATIN1 -s 4326 C:\Temp\pernambuco\pernambuco municipios > municipios.sql
    - dir
    - psql -h localhost -p 5433 -d postgres -U postgres -f pernambuco.sql
      - -d: nome do BD
      - -f: nome do arquivo
      - -U: nome do usuário
      - -h: nome do host

- *Shp2pgsql -W LATIN1 -s 4326 C:\Temp\pernambuco\pernambuco municipios > municipios.sql*
- *Shp2pgsql -W LATIN1 -s 4326 C:\Temp\hidrografia\hidrografia hidrografia > hidrografia.sql*
- *Shp2pgsql -W LATIN1 -s 4326 C:\Temp\bens_imateriais\bens_imateriais bens_imateriais > bens_imateriais.sql*
- *Shp2pgsql -W LATIN1 -s 4326 C:\Temp\regioes_desenvolvimento\regioes_desenvolvimento regioes > regioes.sql*

- psql -h localhost -p 5433 -d postgres -U postgres -f municipios.sql
- psql -h localhost -p 5433 -d postgres -U postgres -f hidrografia.sql
- psql -h localhost -p 5433 -d postgres -U postgres -f bens_imateriais.sql
- psql -h localhost -p 5433 -d postgres -U postgres -f regioes.sql

Usando o PostGIS

- **Consultas Espaciais**
  - Forma básica
    - *SELECT gid, ST_AsText(geom) AS geom, nome FROM pernambuco;*
  - Operadores úteis
    - &&: Informa se o MBR de uma geometria intersecta o MBR de outra
    - ~= : Testa se duas geometrias são geometricamente idênticas
    - = : Testa se os MBR de duas geometrias são idênticos
    - Exemplo:
      - *SELECT GID, NOME*

        FROM PERNAMBUCO WHERE

        GEOM = ST_GeomFromText ('LINESTRING(-41 -8, 41 -40)' , 4326);

Usando o PostGIS

- Exemplos de Consultas Espaciais
  - *SELECT GID, NOME*

    FROM PERNAMBUCO WHERE

    GEOM ~= ST_GeomFromText ('LINESTRING(-41 -8, 41 -40)' , 4326);

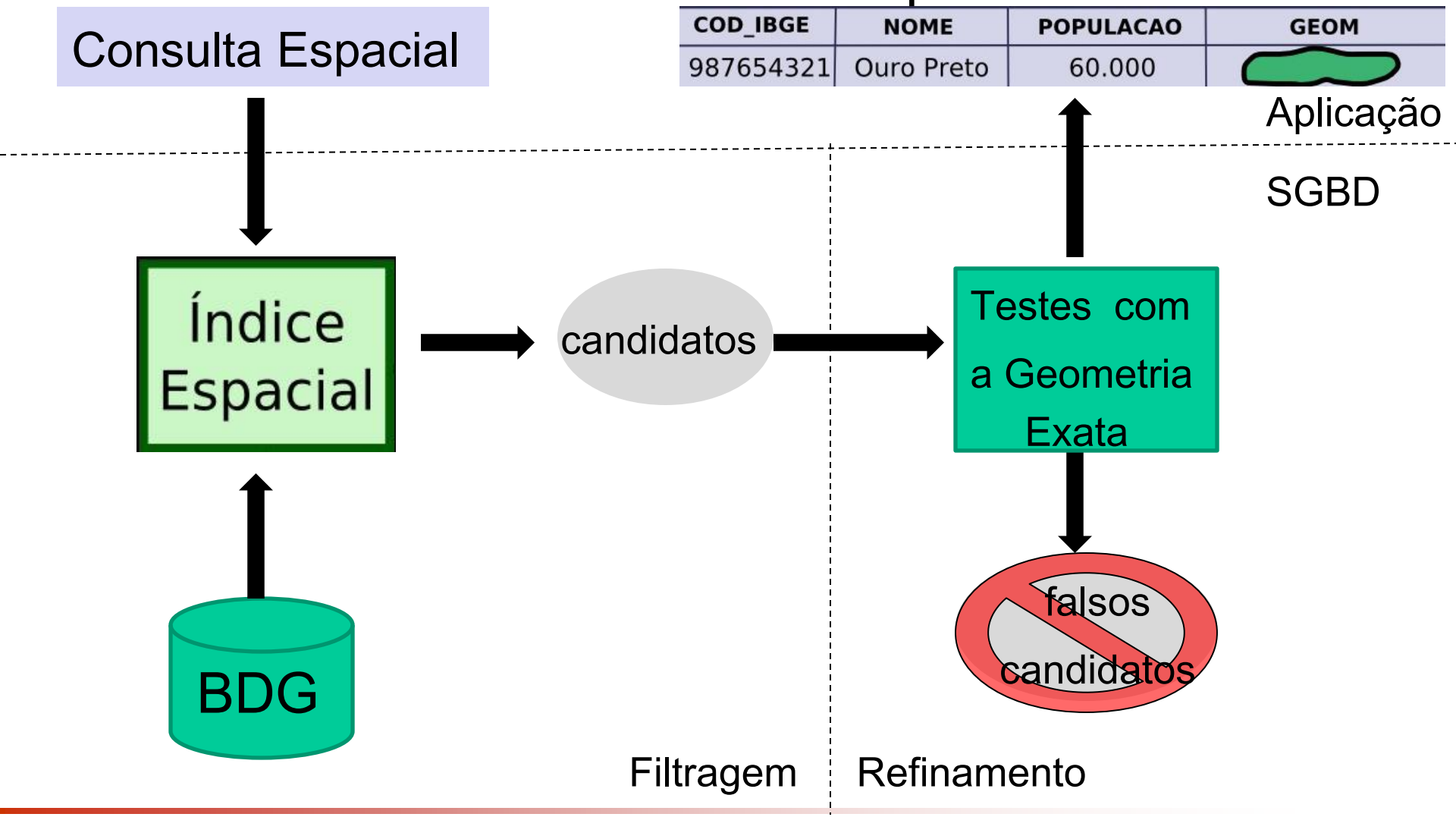
  - SELECT GID, NOME

    FROM PERNAMBUCO WHERE

    GEOM && ST_GeomFromText ('LINESTRING(-41 -8, 41 -40)' , 4326);

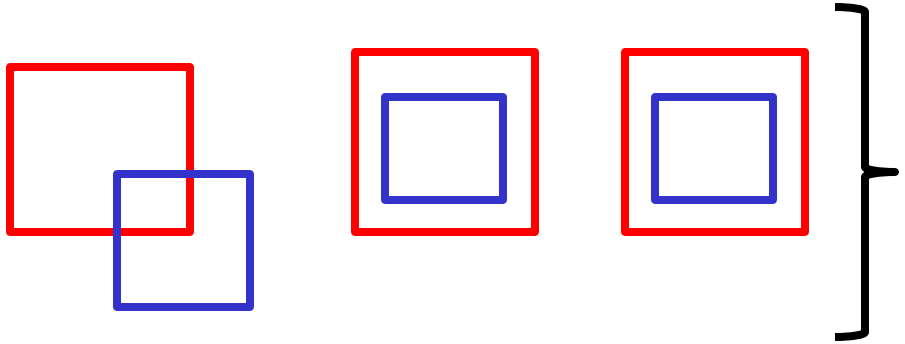
Usando o PostGIS

- Processamento de Consultas Espaciais

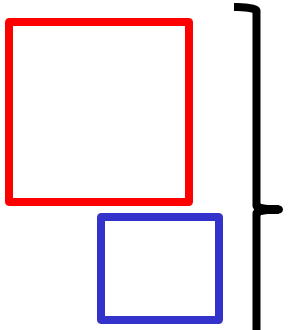


Usando o PostGIS

- Operador de índice espacial é: &&
  - Minimum bounding box intersects



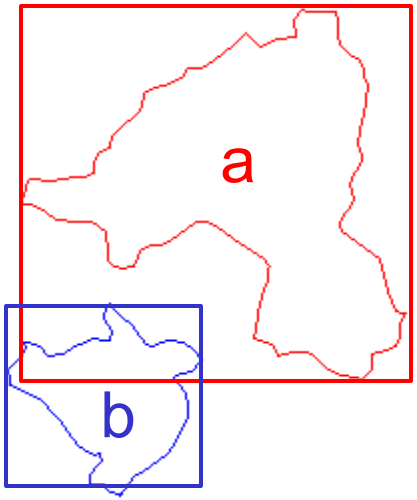
a && b = true



a && b = false

Usando o PostGIS

- Minimum Bounding Box (MBB) não é suficiente



a && b = TRUE

ST_Intersects(a , b) = FALSE

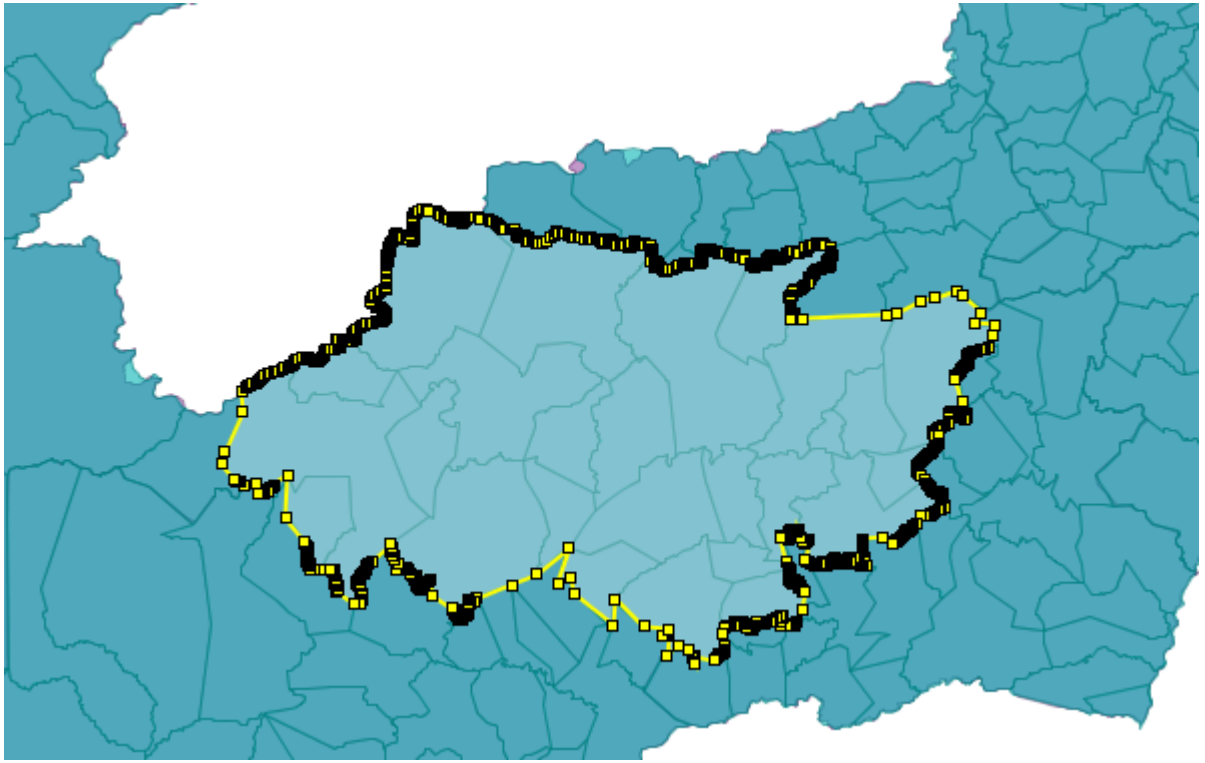
- Processsamento em duas etapas:
  1. Se usa o MBB para diminuir o número de candidatos
  2. Se usa os operadores topológicos para realizar testes mais finos e então obter a resposta final

Usando o PostGIS

- Observações
  - A partir da versão 1.3.X, os operadores espaciais já fazem uso do índice espacial sem a necessidade de explicitar o índice &&
  - Se não for usado o índice, basta utilizar os métodos prefixados com: '_'
    - _ST_TOUCHES

Usando o PostGIS

- **Consultas simples a Objetos geográficos**
  - Quais os municípios que estão na região MATA SUL?
    - Necessitamos fazer um Overlay! Como?



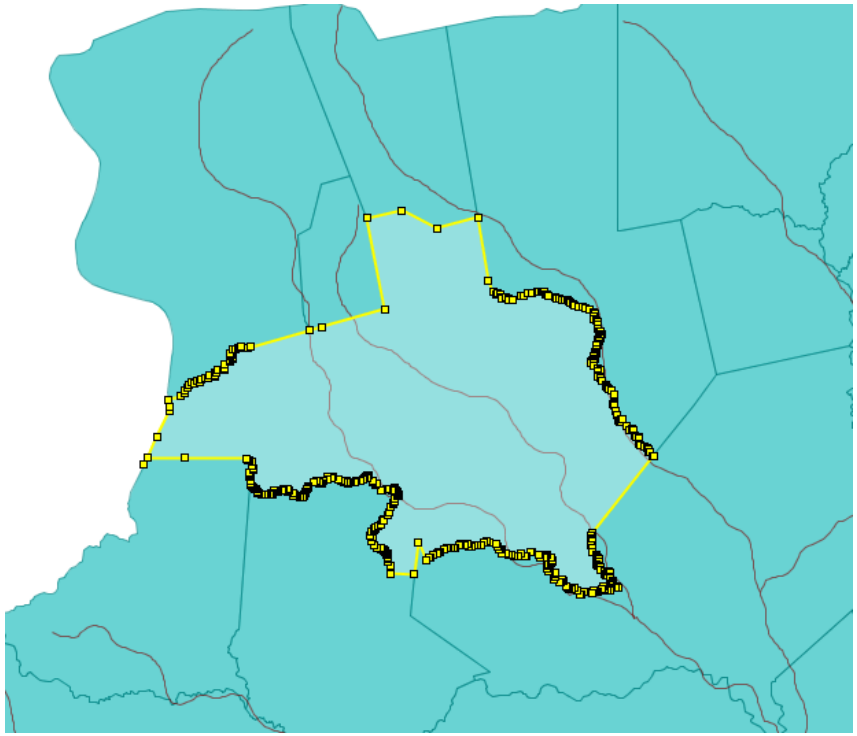
Usando o PostGIS

- Consultas simples a Objetos geográficos
  - Quais os municípios que estão na região MATA SUL?

```
SELECT m.nome,
            ST_Overlaps (m.geom, r.geom)
FROM     municipios m,  regioes  r
WHERE  ST_Overlaps (m.geom, r.geom)
AND       r.regdesenv = 'MATA SUL'
```

Usando o PostGIS

- **Operações Topológicas em SQL**
  - Quais são os recursos hídricos do município de Ouricuri?
    - Obter os relacionamentos espaciais entre o município de Ouricuri e seus recursos hídricos.



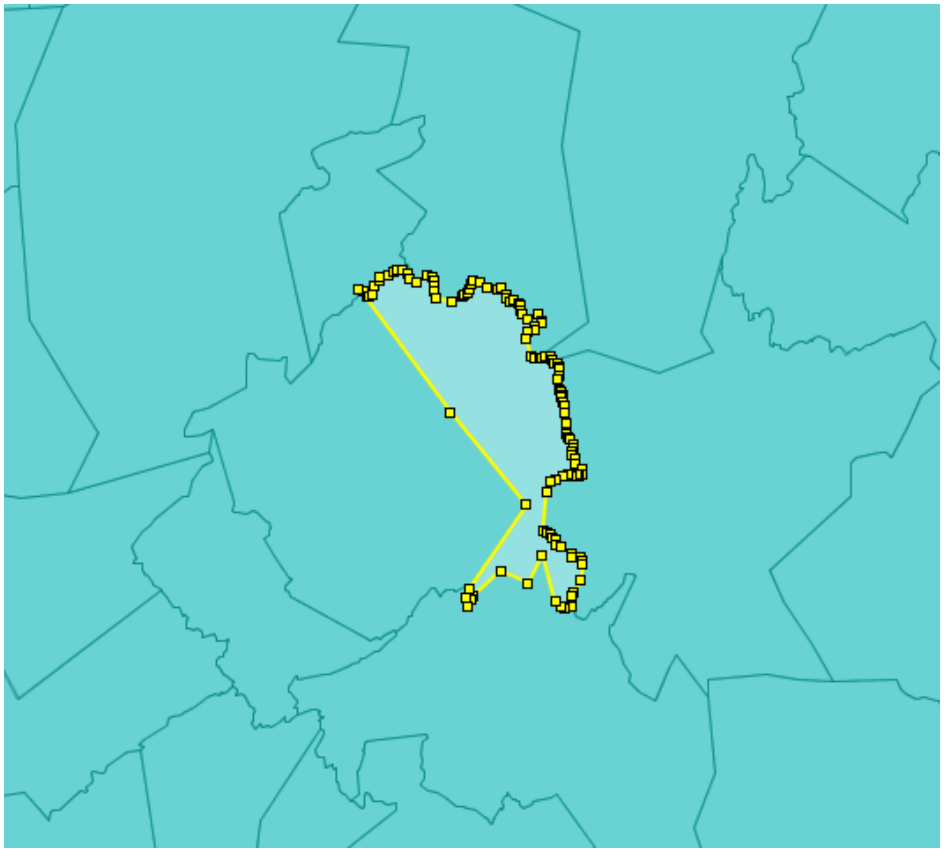
Usando o PostGIS

- Operações Topológicas em SQL
  - Quais são os recursos hídricos do município de Ouricuri?

```
SELECT  m.nome,
        h.nome,
        ST_AsText(h.geom)
FROM    municipios m,  hidrografia h
WHERE   ST_Intersects (m.geom,  h.geom)
AND     m.nome = 'Ouricuri' ;
```

Usando o PostGIS

- **Operações Topológicas em SQL**
  - Quais os municípios vizinhos a Primavera?



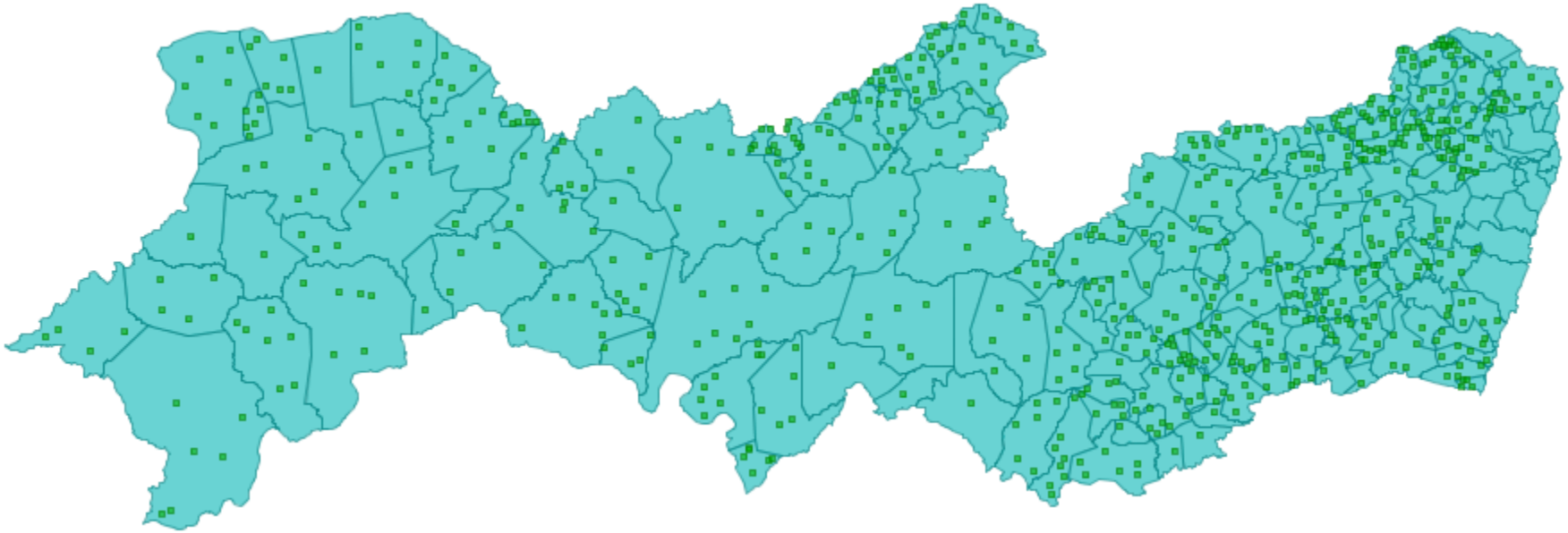
Usando o PostGIS

- **Operações Topológicas em SQL**
  - Quais os municípios vizinhos a Primavera?

```
SELECT  m1.nome, m2.nome
FROM    municipios m1, municipios m2
WHERE   ST_Touches (m1.geom, m2.geom)
AND     m1.nome =  'Primavera'
```

Usando o PostGIS

- Operações Topológicas em SQL
  - Quais os municípios do Estado de Pernambuco que não possuem algum bem imaterial?



Usando o PostGIS

- Operações Topológicas em SQL
  - Quais os municípios do Estado de Pernambuco que não possuem algum bem imaterial?

```
SELECT distinct m.nome
FROM     municipios  m

EXCEPT

SELECT  distinct m.nome
FROM   municipios  m,  bens_imateriais  b
WHERE  ST_Contains(m.geom, b.geom) ;
```

Usando o PostGIS

- Conferindo a integridade dos dados
  - PostGIS pressupõe algumas regras de integridade em relação às geometrias:
    - Geometrias devem ser de acordo com a OGC Simple Feature Specification for SQL
    - Os anéis dos polígonos não devem se sobrepor ou terem auto-intersecções
    - Um MultiPolygon não deve ter polígonos sobrepondo-se
    - Ao contrário de algumas outras extensões espaciais, a orientação dos anéis não é importante

Usando o PostGIS

- Validando geometrias antes de inseri-las no BDGeo
  - ST_ISVALID()
    - Valida as coordenadas de uma geometria
  - Exemplo:
    - SELECT ST_ISVALID ('LINESTRING(0 0, 1 1)'), → t
    - SELECT ST_ISVALID ('LINESTRING(0 0, 0 0)'); → f
  - Opção default é não validar a entrada das geometrias
    - Para validar deve-se adicionar uma restrição à tabela
    - ALTER TABLE parks ADD CONSTRAINT geo_valid_chk
      CHECK (ST_ISVALID (park_geom)) ;

Usando o PostGIS

- Conferindo a integridade dos dados
  - Remover os municípios do Estado de Pernambuco que possuam uma geometria inválida.

*SELECT **

FROM regioes

*WHERE **NOT ST_Isvalid (geom)***

Usando o PostGIS

- Provendo suporte à projeções cartográficas
  - PostGIS possui uma tabela de metadados com todos os sistemas de referência espacial providos:
    - Tabela: spatial_ref_sys
  - ST_Transform (geometria, novo–srid)
    - Retorna uma nova geometria com as coordenadas transformadas para um novo SRID
    - O novo SRID deve estar presente na tabela spatial_ref_sys

Usando o PostGIS

- Projeções Cartográficas em SQL
  - Qual é a área do município de Ibimirim em hectares?

```
SELECT ST_Area( ST_Transform(geom, 29183))/10000
AS hectares
FROM municipios m
WHERE m.nome = 'Ibimirim' ;
```

  - Qual é o maior município do Estado de Pernambuco em termos de área?

```
SELECT m.nome,
        ST_Area(ST_Transform (geom, 29183)) / 10000 AS hectares
FROM municipios m
ORDER BY hectares DESC  LIMIT 1;
```

Usando o PostGIS

- Projeções Cartográficas em SQL
  - Qual é o menor município do estado de Pernambuco?

```
SELECT m.nome
FROM municipios m
WHERE ST_Area(m.geom) =
        (SELECT MIN(ST_Area(m2.geom)) FROM municipios m2)
```

Usando o PostGIS

- Projeções Cartográficas em SQL
  - Qual é o comprimento total dos recursos hídricos do município de Ouricuri?

```
SELECT SUM (ST_Length(ST_Transform( r.rh, 29183))/1000) AS km
FROM  (SELECT ST_Intersection(m.geom,  h.geom) AS rh
        FROM   municipios m,  hidrografia h
        WHERE ST_Intersects(m.geom, h.geom)
        AND m.nome = 'Ouricuri' )  AS r ;
```

Usando o PostGIS

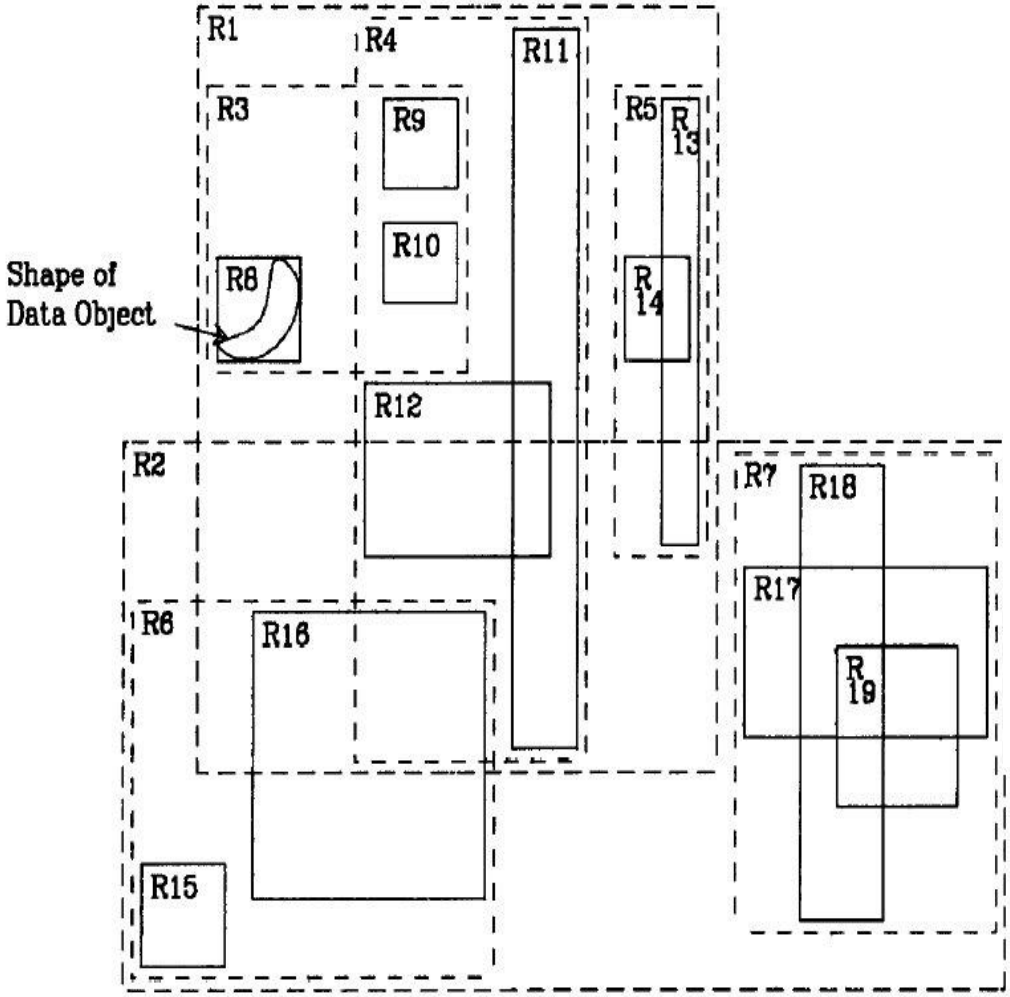
- Usando Índices Geográficos
  - GiST (Generalized Search Tree)
    - Consultas convencionais em tabelas geográficas não usufruem do mecanismo GiST
  - Sintaxe para criação do índice:

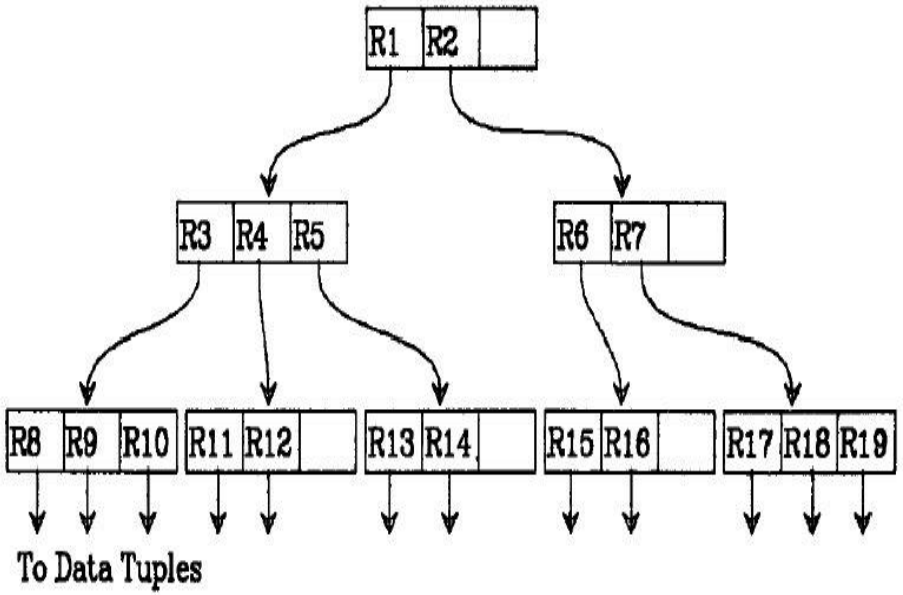
    CREATE INDEX nome_índice ON nome_tabela

    USING GIST (coluna);
  - Índices são utilizados pelo PostgreSQL quando ele reconhece algum operador na consulta: < ,= , ST_WITHIN,...

Usando o PostGIS

- GIST (variação da R-Tree)





Usando o PostGIS

- Usando Índices Geográficos
  - Exemplo:

```
CREATE INDEX indice_bens ON bens_imateriais
    USING GIST (geom);
```

  - É possível usufruir do GiST na consulta: Selecione os tipos de bens materiais que estejam a menos de 20 metros do ponto (-35.225 -7.969)?

```
SELECT distinct(tipo)
   FROM  bens_imateriais
   WHERE ST_distance(geom, ST_GeometryFromText('POINT
        (-35.225 -7.969)', 4326))  < 20 ;
```

Usando o PostGIS

- Principais funções de relacionamento espacial
  - ST_Distance(geometry, geometry)
  - ST_Equals(geometry, geometry)
  - ST_Disjoint(geometry, geometry)
  - ST_Intersects(geometry, geometry)
  - ST_Touches(geometry, geometry)
  - ST_Crosses(geometry, geometry)
  - ST_Within(geometry, geometry)
  - ST_Overlaps(geometry, geometry)
  - ST_Contains(geometry, geometry)

Usando o PostGIS

- Principais funções de processamento geométrico
  - ST_Centroid(geometry)
  - ST_Area(geometry)
  - ST_Length(geometry)
  - ST_PointOnSurface(geometry)
  - ST_Boundary(geometry)
  - ST_Buffer(geometry, double, [integer])
  - ST_Intersection(geometry, geometry)
  - ST_Difference(geometry, geometry)
  - ST_GeomUnion(geometry, geometry)

Usando o PostGIS

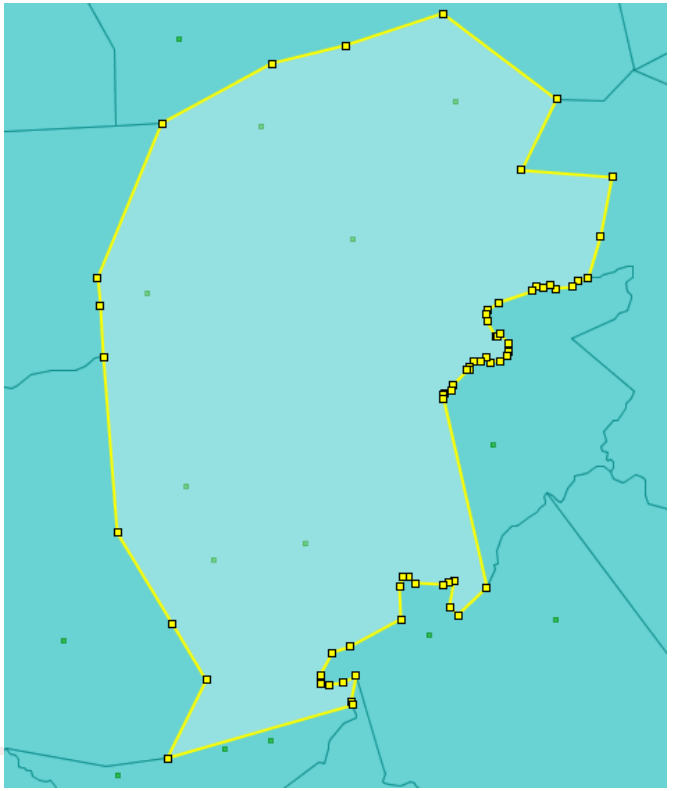
- Exemplos de consultas espaciais
  - Crie uma tabela com todas as partes dos rios de Parnamirim?

```
CREATE TABLE rios_parnamirim2 as
 SELECT h.nome,
        ST_intersection (h.geom, m.geom) AS intersection_geom,
        ST_length ( h.geom ) AS  rio_orig_length
 FROM  hidrografia h,  municipios m
 WHERE ST_intersects ( h.geom,  m.geom )
 AND   m.nome = 'Parnamirim' ;
```

Usando o PostGIS

- Exemplos de consultas espaciais
  - Qual é a quantidade de bens imateriais contidos no município de Gravatá?

```
SELECT count(b.gid) as Qtde_BENS
    FROM municipios m, bens_imateriais b
    WHERE ST_contains(m.geom , b.geom)
    AND m.nome = 'Gravatá'
```



Usando o PostGIS

- Exemplos de consultas espaciais
  - Qual é a quantidade de bens imateriais contidos em cada município?

```
SELECT m.nome, count(b.gid) as Qtde_BENS
  FROM municipios m, bens_imateriais b
  WHERE ST_contains(m.geom , b.geom)
  GROUP BY m.nome
  ORDER BY  Qtde_BENS desc;
```

cin.ufpe.br

Centro de Informática

U · F · P · E

Universidade Federal de Pernambuco