Bancos de Dados Espaciais

Contents

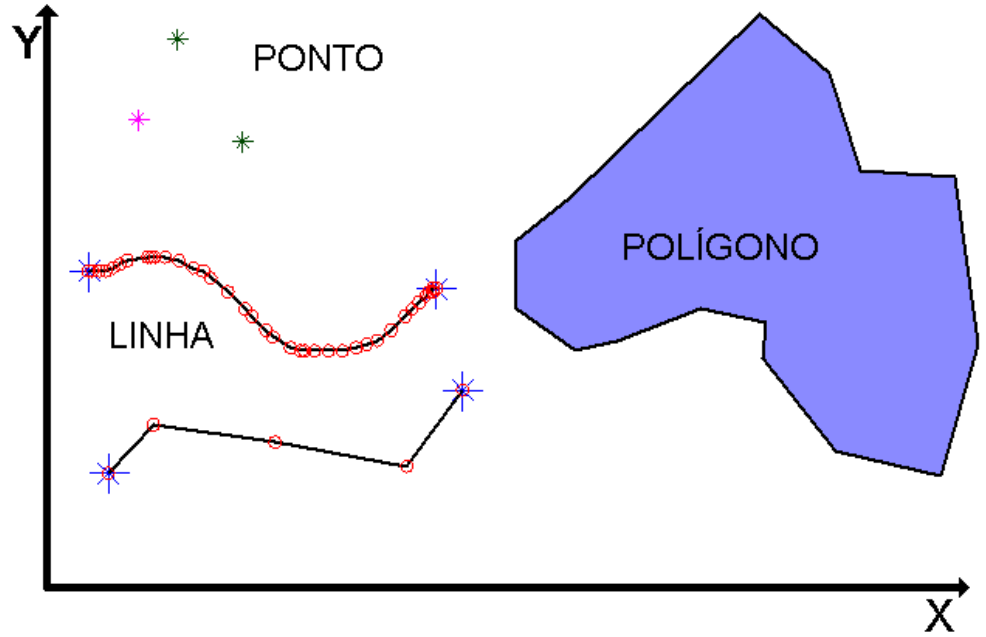
- Tipos de dados espaciais
- *Simple feature access* – SQL option: arquitetura de armazenamento;
- Criação de tabelas com atributos espaciais;
- Importação de dados;
- Índices espaciais;
- *Simple feature access* – SQL option: rotinas suportadas;
- Consultas SQL.

Tipos de dados espaciais
Modelos conceituais

- A classe georreferenciada descreve um conjunto de objetos que possuem representação espacial e estão associados a regiões da superfície da terra, representando a visão de **campos** e de **objetos**.
- Classes geo-campo representam objetos e fenômenos **distribuídos continuamente** no espaço, correspondendo a variáveis como tipo de solo, relevo e geologia.
- Classes geo-objeto representam objetos geográficos particulares, **individualizáveis**, associados a elementos do mundo real.

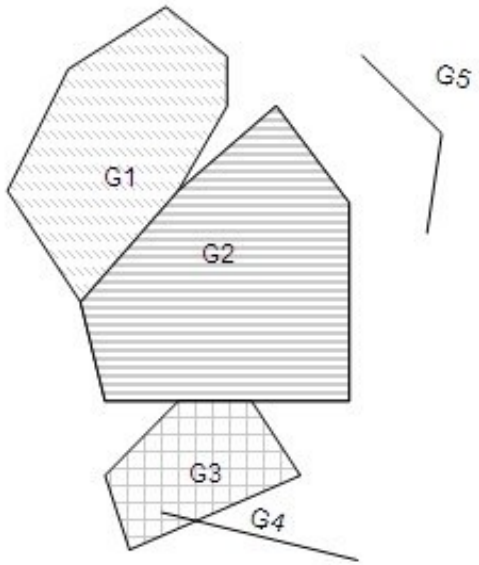
Tipos de dados espaciais
Estrutura de Dados Vetorial

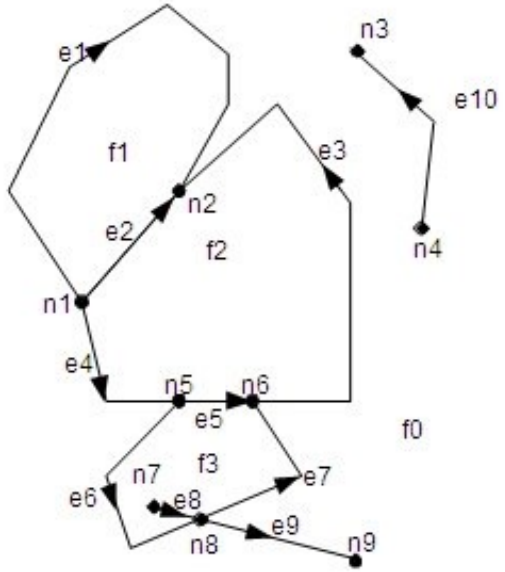
- Estruturas vetoriais são utilizadas para representar as coordenadas das fronteiras de cada entidade geográfica, através de três formas básicas: pontos, linhas, e áreas (ou polígonos), definidas por suas coordenadas cartesianas.
- Um ponto é um par ordenado (x, y) de coordenadas espaciais.
- Uma linha é um conjunto de pontos conectados.
- Uma área (ou polígono) é a região do plano limitada por uma ou mais linhas poligonais conectadas de tal forma que o último ponto de uma linha seja idêntico ao primeiro da próxima.

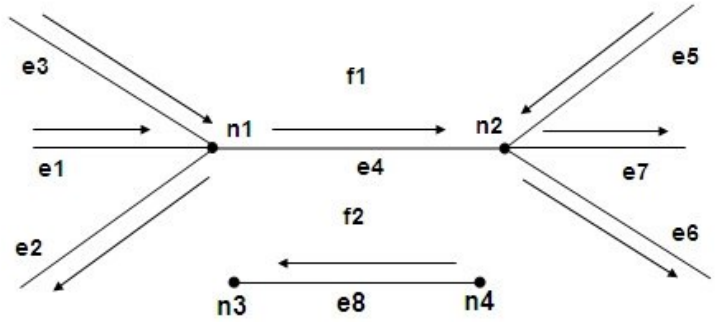


Tipos de dados espaciais
Vetores e Topologia

- A topologia é a parte da matemática na qual se investigam as propriedades das configurações que permanecem invariantes nas transformações de rotação, translação e escala: relações como adjacência (*"vizinho de"*), pertinência (*"contido em"*), intersecção e cruzamento.
- ***Polígonos sem topologia:*** guardam-se as coordenadas de cada objeto isoladamente, e assim duplicam-se as fronteiras em comum com outros objetos;
- ***Topologia arco-nó-polígono:*** armazena-se cada fronteira comum uma única vez, indicando a que objetos elas estão associadas.
- ***Topologia arco-nó:*** Um nó pode ser definido como o ponto de intersecção entre duas ou mais linhas, correspondente ao ponto inicial ou final de cada linha. Nenhuma linha poderá estar desconectada das demais para que a topologia da rede possa ficar totalmente definida.
- ***Dados 2,5 D:*** a cada localização no espaço é associado um valor numérico de atributo. Neste caso, como os valores de localização estão no plano e o valor adicional descreve uma superfície sobre este plano.



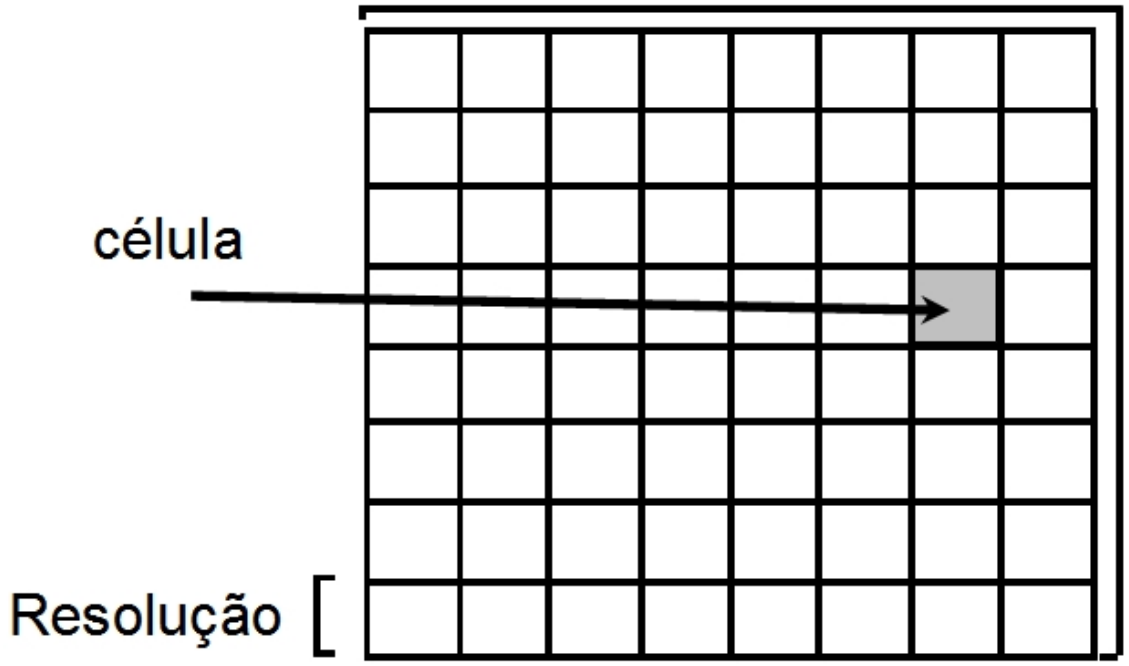




EDGE_ID	START_NODE_ID	END_NODE_ID	NEXT_LEFT_EDGE_ID	PREV_LEFT_EDGE_ID	NEXT_RIGHT_EDGE_ID	PREV_RIGHT_EDGE_ID	LEFT_FACE_ID	RIGHT_FACE_ID	GEOMETRY
E4	N1	N2	-E5	E3	E2	-E6	F1	F2	(...)
E8	N4	N3	-E8	-E8	E8	E8	F2	F2	(...)

Tipos de dados espaciais
Representação matricial

- Estruturas matriciais usam uma **grade regular** sobre a qual se representa, célula a célula, o elemento que está sem do representado.
- A cada célula, atribui-se um código referente ao atributo estudado, de tal forma que o computador saiba a que elemento ou objeto pertence determinada célula.
- O espaço é representado como uma matriz *P(m, n)* composto de *m* colunas e *n* linhas, onde cada célula possui um número de linha, um número de coluna e um valor correspondente ao atributo estudado;
- Cada célula é individualmente acessada pelas suas coordenadas.



Simple feature access – SQL option
Arquitetura de Armazenamento

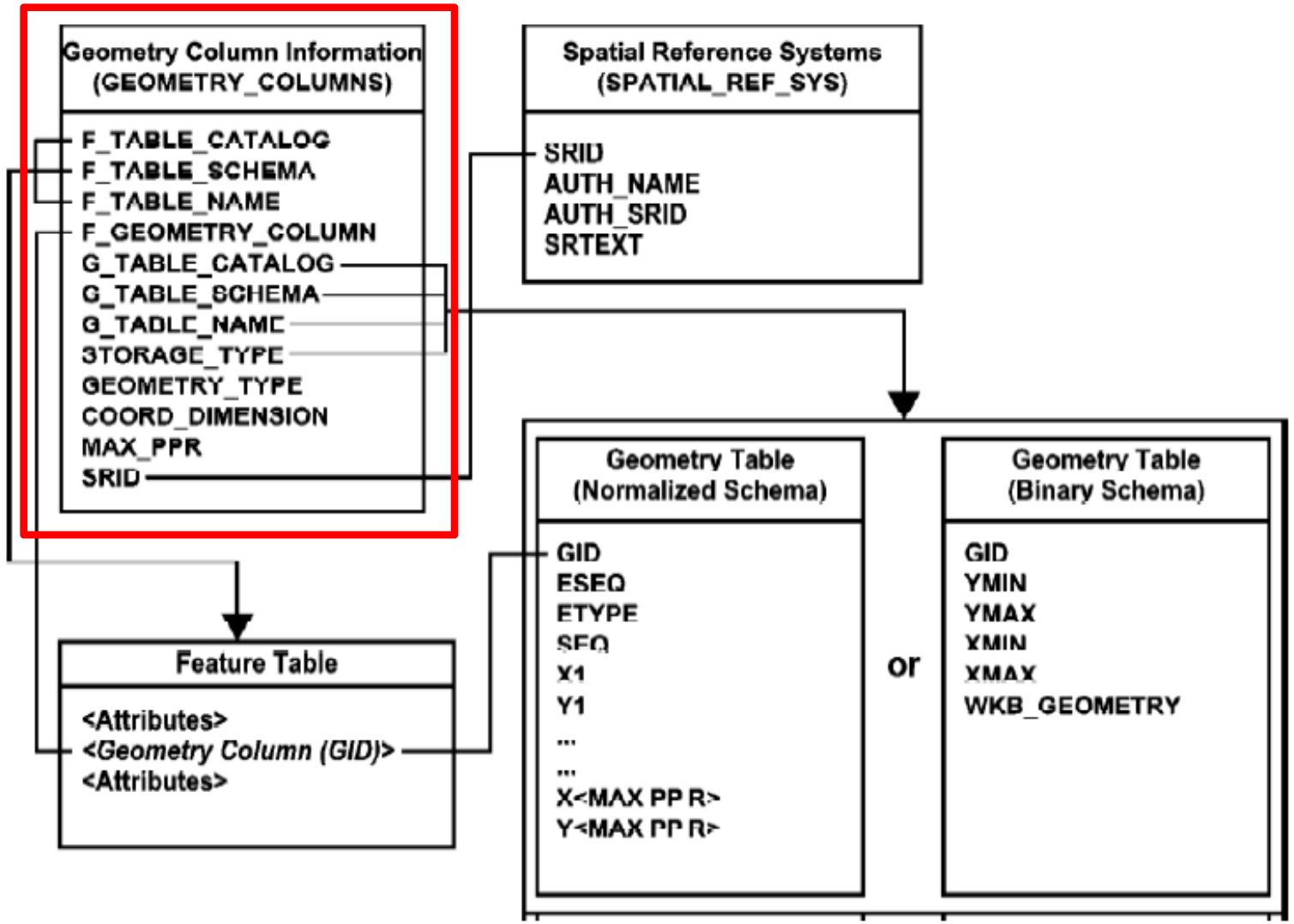
Este padrão especifica um esquema SQL para apoiar o armazenamento, o acesso, a consulta e a a atualização de feições geoespaciais com geometrias simples, empregando o *SQL Call Level Interface* (SQL – CLI | ISO/IEC 9075-3:2003).

O SQL-CLI é uma divisão do SQL que especifica uma interface para o SQL que pode ser usada por um programa de aplicação. A implementação mais conhecida é o ODBC.

Existem 3 estratégias de implementação:

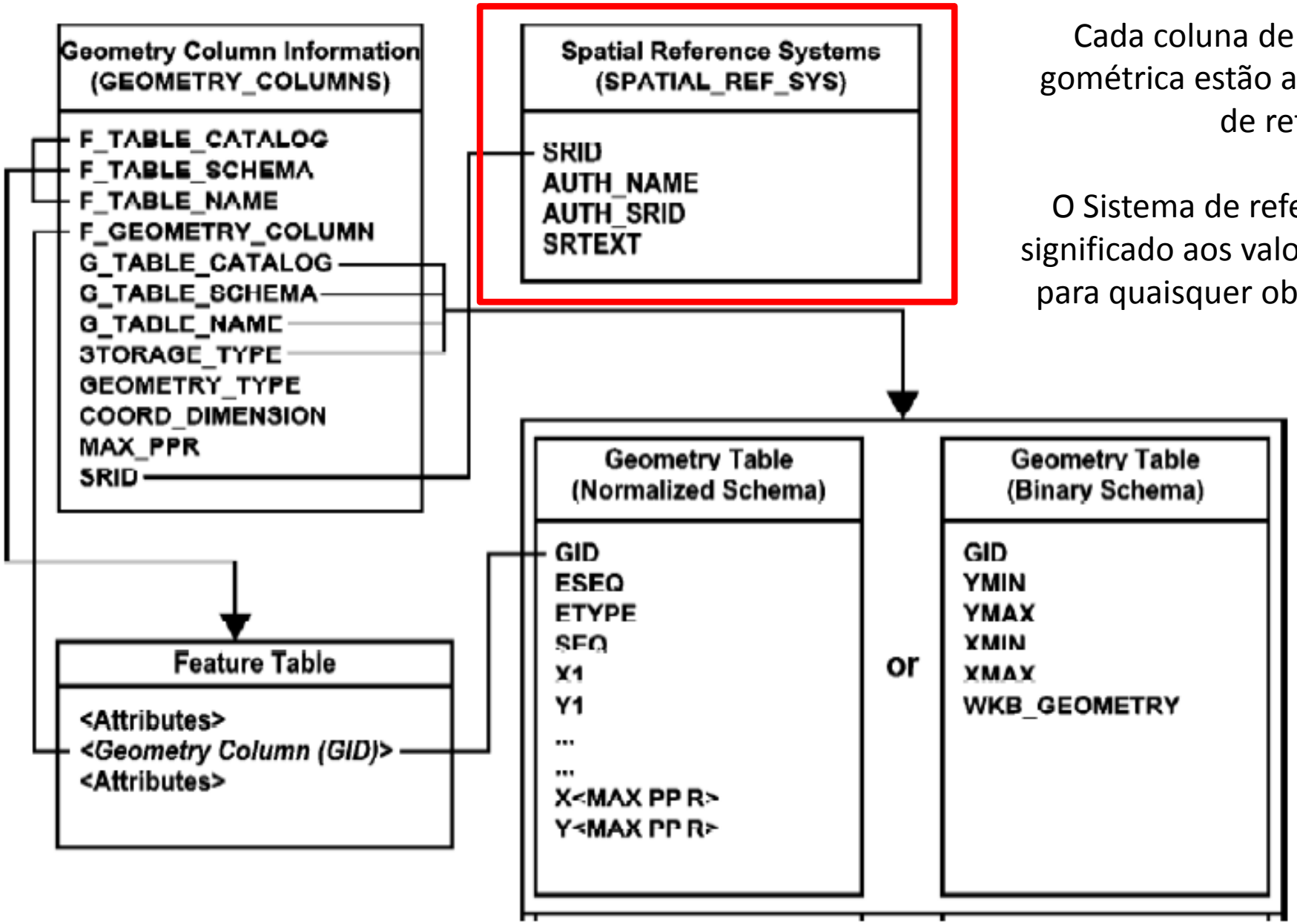
- Tabelas de feições baseadas em tipos pré-definidos da SQL: Usando tipos numéricos da SQL para armazenamento das geometrias e acesso via SQL;
- Tabelas de feições baseadas em tipos pré-definidos da SQL: Usando tipos binários da SQL para armazenar geometrias e acesso via SQL/CLI;
- Tabelas de feições com Tipos Geométricos suportando acesso textual e binário às geometrias através da SQL.

Tabelas de feições baseadas em tipos pré-definidos da SQL



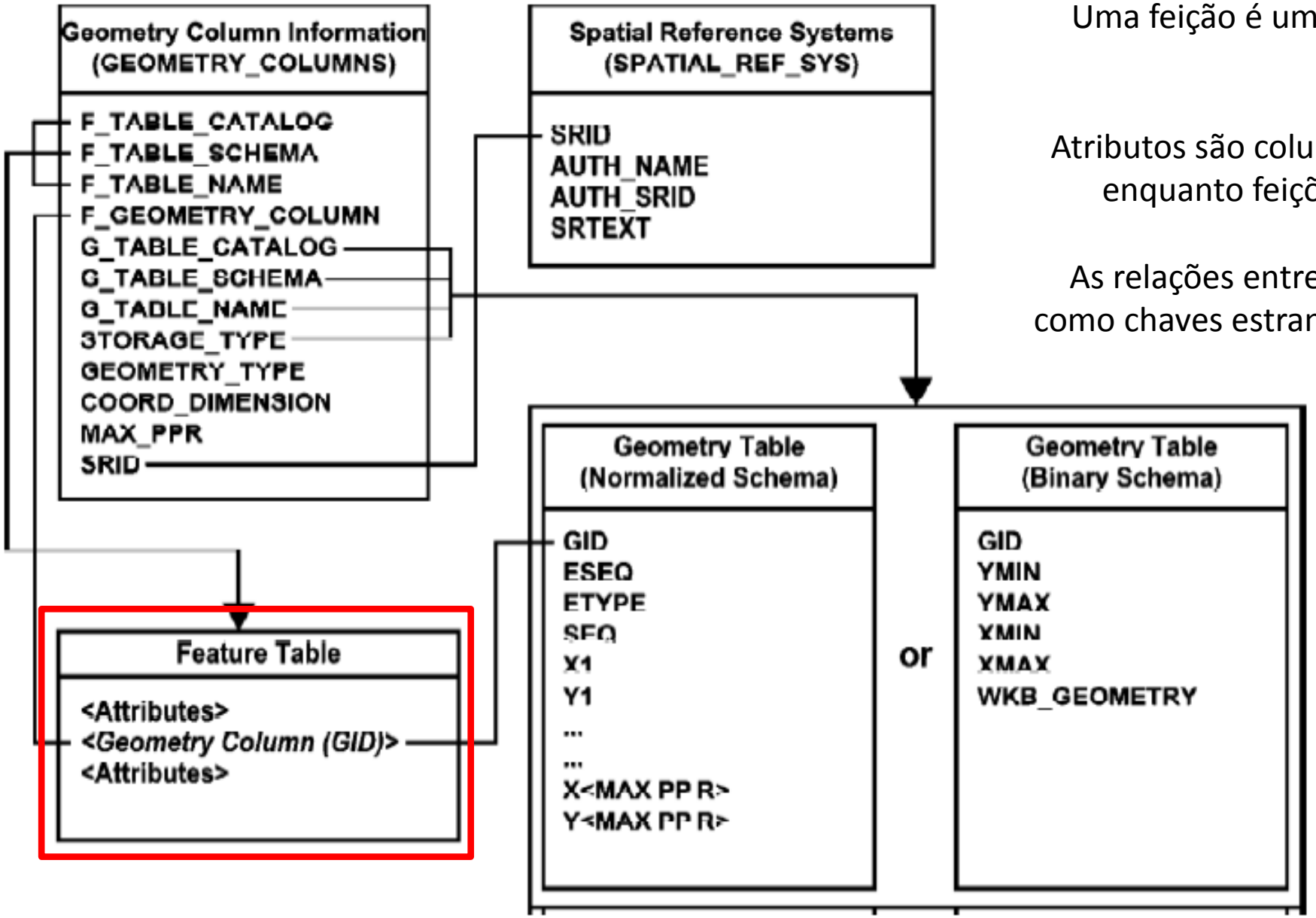
Tabelas de feições e colunas de geometria são identificadas por meio da tabela GEOMETRY_COLUMNS.

Cada coluna de geometria no banco de dados possui uma entrada na tabela GEOMETRY_COLUMNS.



Cada coluna de geometria e cada entidade gométrica estão associadas a um único Sistema de referência espacial.

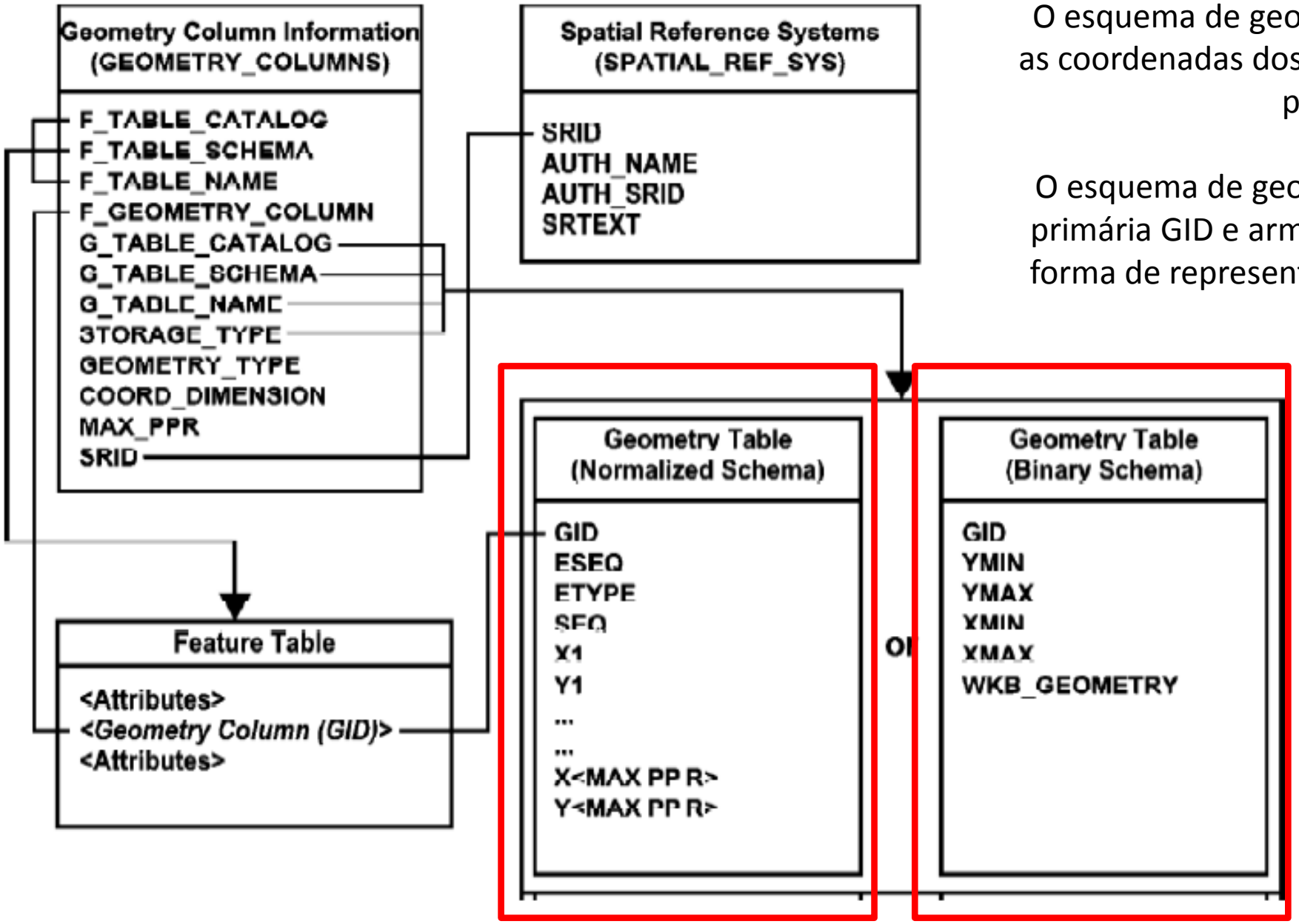
O Sistema de referência especial é que atribui significado aos valores numéricos de coordenadas para quaisquer objetos armazenados na coluna.



Uma feição é uma abstração de um objeto do mundo real.

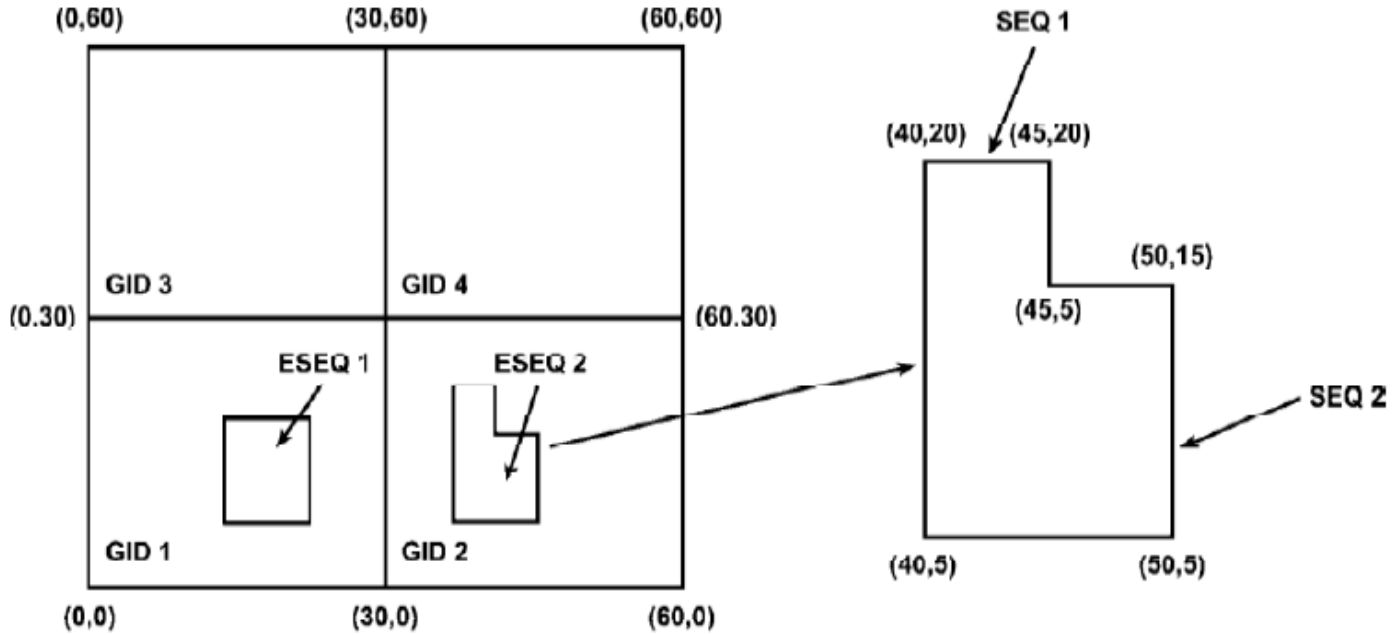
Atributos são colunas em uma tabela de feições, enquanto feições são linhas dessa tabela.

As relações entre feições podem ser definidas como chaves estrangeiras entre tabelas de feições.



O esquema de geometria normalizada armazena as coordenadas dos objetos como tipos numéricos pré-definidos.

O esquema de geometria binária usa uma chave primária GID e armazena o objeto geométrico na forma de representação binária (*WKBGeometry*).



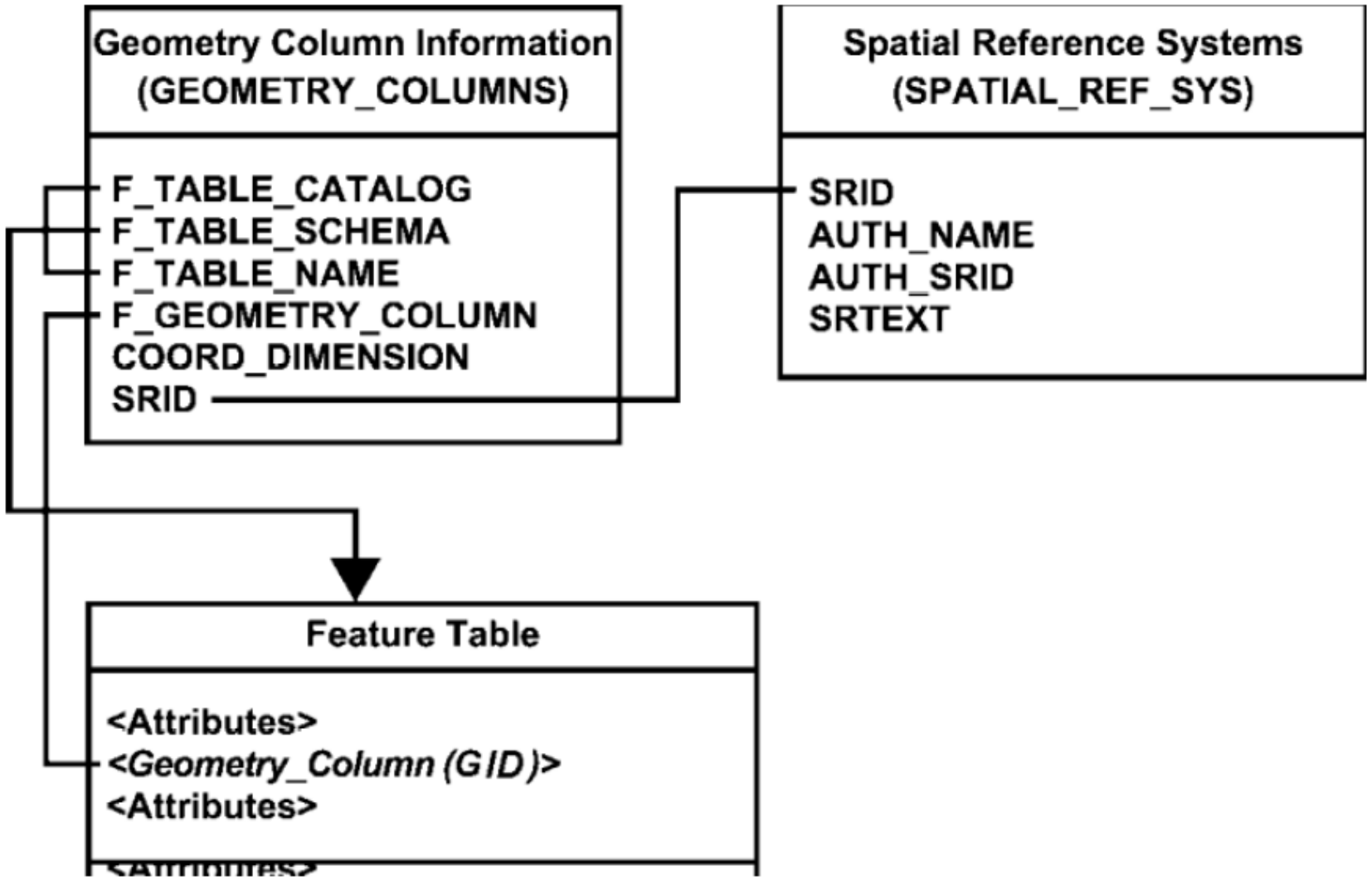
GID 1	ESEQ	ETYPE	SEQ	X0	Y0	X1	Y1	X2	Y2	X3	Y3	X4	Y4
1	1	3	1	0	0	0	30	30	30	30	0	0	0
1	2	3	1	10	10	10	20	20	20	20	10	10	10
2	1	3	1	30	0	30	30	60	30	60	0	30	0
2	2	3	1	40	5	40	20	45	20	45	15	50	15
2	2	3	1	50	15	50	5	40	5	Nil	Nil	Nil	Nil
3	1	3	1	0	30	0	60	30	60	30	30	0	30
4	1	3	1	30	30	30	60	60	60	60	30	30	30

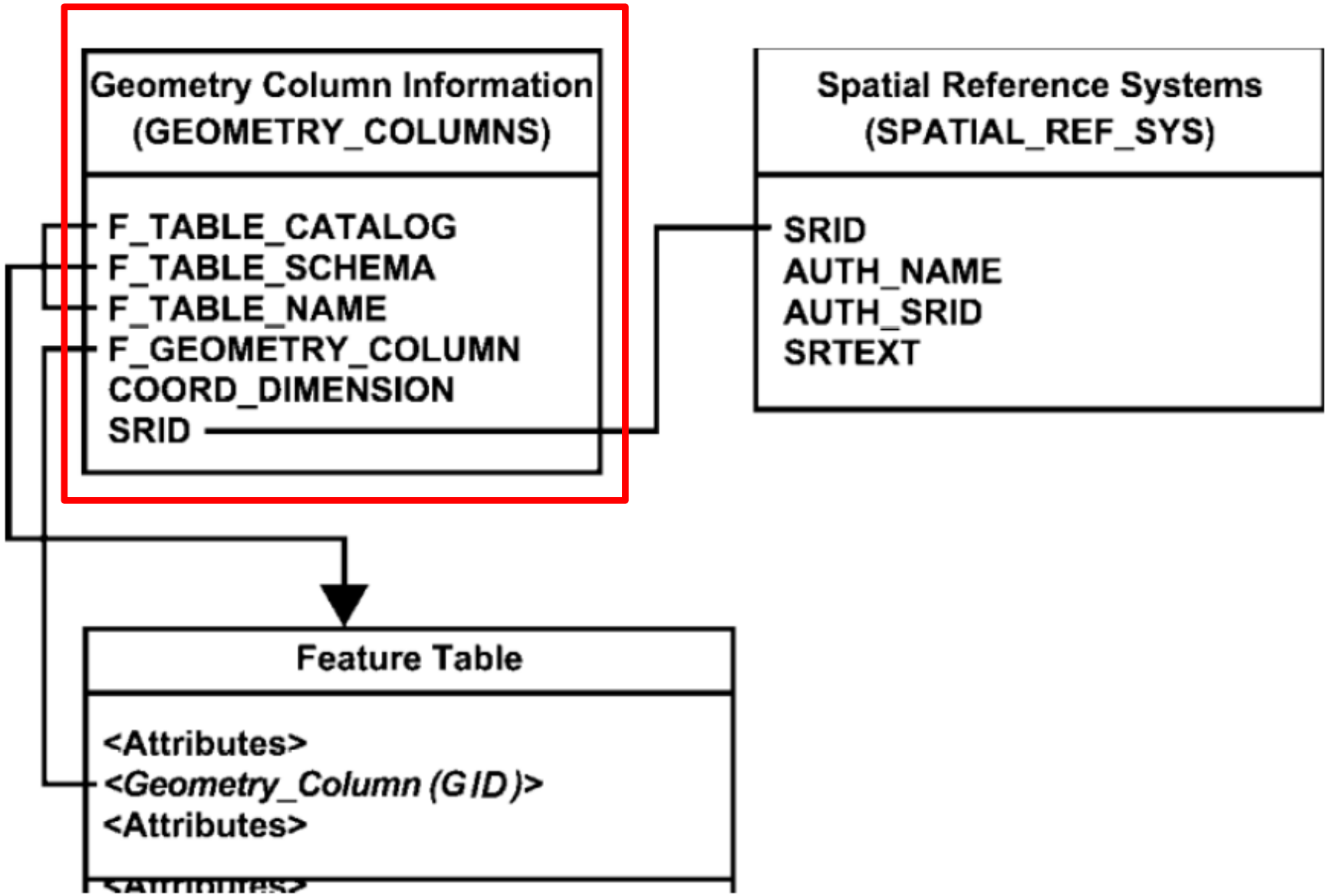
Cada objeto geométrico é identificado por uma chave (GID) e consiste de um ou mais elementos primitivos ordenados por uma sequência de elementos (ESEQ).

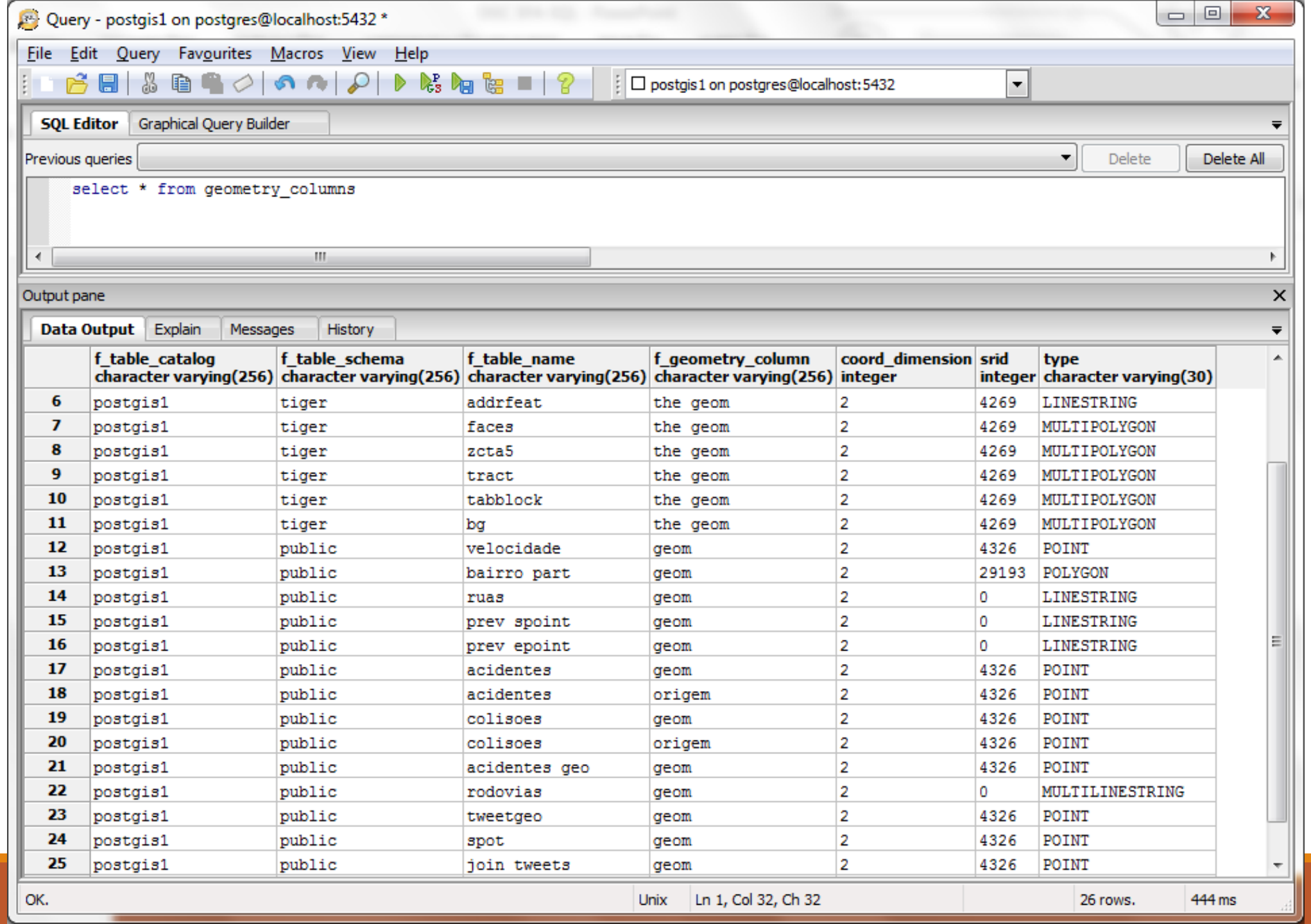
Cada elemento primitivo no objeto geométrico é distribuído por uma ou mais linhas na tabela de geometria, identificado por um tipo de primitiva (ETYPE), e ordenado por um número de sequência (SEQ).

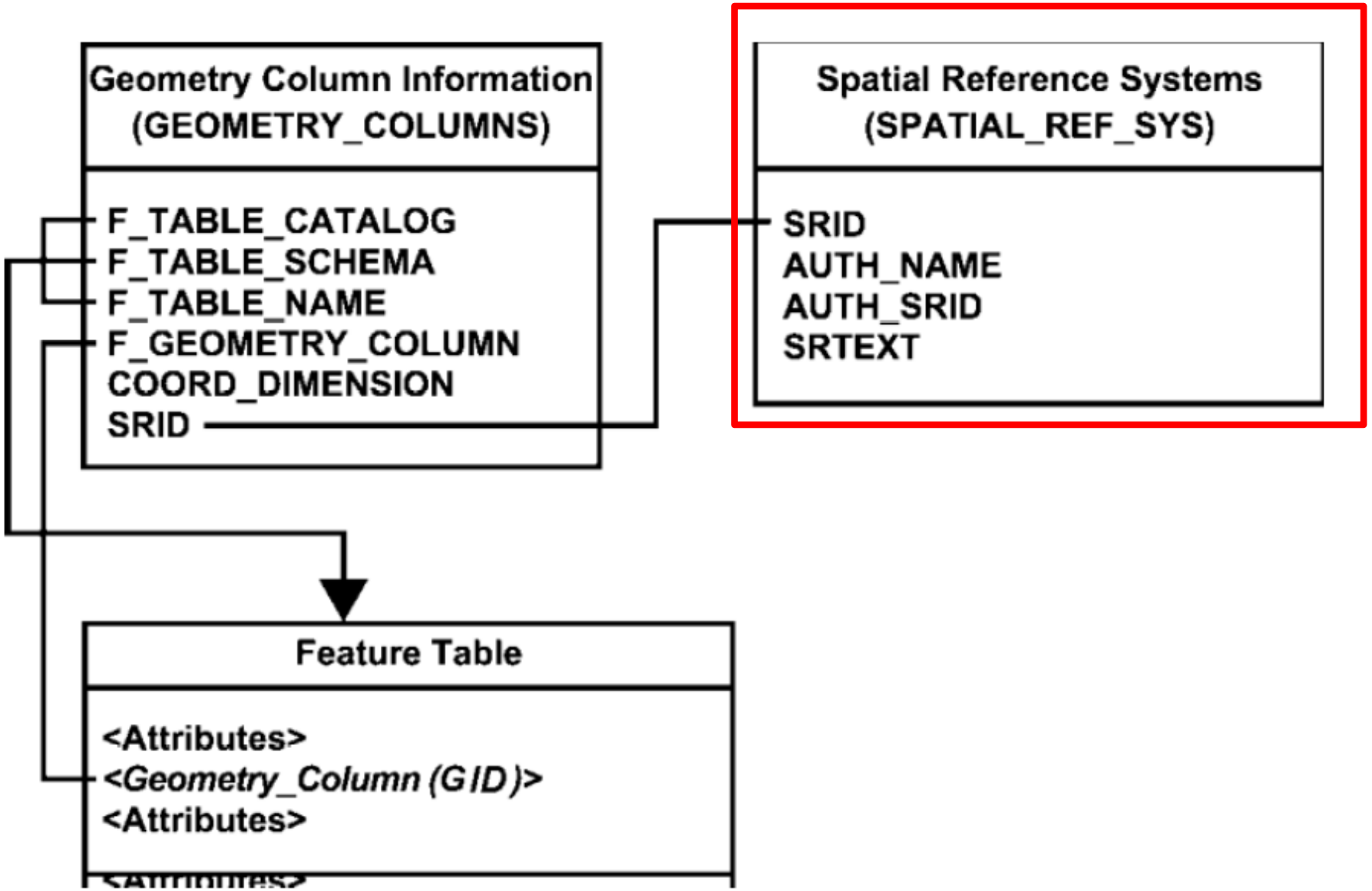
GID	XMIN	YMIN	XMAX	YMAX	Geometry
1	0	0	30	30	< WKBGeometry >
2	30	0	60	30	< WKBGeometry >
3	0	30	30	60	< WKBGeometry >
4	30	30	60	60	< WKBGeometry >

Tabelas de feições com Tipos Geométricos




Geometry Column Information
(GEOMETRY_COLUMNS)
F_TABLE_CATALOG
F_TABLE_SCHEMA
F_TABLE_NAME
F_GEOMETRY_COLUMN
COORD_DIMENSION
SRID
Spatial Reference Systems
(SPATIAL_REF_SYS)
SRID
AUTH_NAME
AUTH_SRID
SRTEXT
Feature Table
<Attributes>
<Geometry_Column (GID)>
<Attributes>


Query - postgis1 on postgres@localhost:5432 *
File Edit Query Favourites Macros View Help
postgis1 on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
select * from geometry_columns
Output pane
Data Output
Explain
Messages
History
f_table_catalog character varying(256)
f_table_schema character varying(256)
f_table_name character varying(256)
f_geometry_column character varying(256)
coord_dimension integer
srid integer
type character varying(30)
6 postgis1 tiger addrfeat the geom 2 4269 LINESTRING
7 postgis1 tiger faces the geom 2 4269 MULTIPOLYGON
8 postgis1 tiger zcta5 the geom 2 4269 MULTIPOLYGON
9 postgis1 tiger tract the geom 2 4269 MULTIPOLYGON
10 postgis1 tiger tabblock the geom 2 4269 MULTIPOLYGON
11 postgis1 tiger bg the geom 2 4269 MULTIPOLYGON
12 postgis1 public velocidade geom 2 4326 POINT
13 postgis1 public bairro part geom 2 29193 POLYGON
14 postgis1 public ruas geom 2 0 LINESTRING
15 postgis1 public prev spoint geom 2 0 LINESTRING
16 postgis1 public prev epoint geom 2 0 LINESTRING
17 postgis1 public acidentes geom 2 4326 POINT
18 postgis1 public acidentes origem 2 4326 POINT
19 postgis1 public colisoes geom 2 4326 POINT
20 postgis1 public colisoes origem 2 4326 POINT
21 postgis1 public acidentes geo geom 2 4326 POINT
22 postgis1 public rodovias geom 2 0 MULTILINESTRING
23 postgis1 public tweetgeo geom 2 4326 POINT
24 postgis1 public spot geom 2 4326 POINT
25 postgis1 public join tweets geom 2 4326 POINT
OK.
Unix
Ln 1, Col 32, Ch 32
26 rows.
444 ms


Geometry Column Information
(GEOMETRY_COLUMNS)
F_TABLE_CATALOG
F_TABLE_SCHEMA
F_TABLE_NAME
F_GEOMETRY_COLUMN
COORD_DIMENSION
SRID
Spatial Reference Systems
(SPATIAL_REF_SYS)
SRID
AUTH_NAME
AUTH_SRID
SRTEXT
Feature Table
<Attributes>
<Geometry_Column (GID)>
<Attributes>

Query - postgis1 on postgres@localhost:5432 *

File Edit Query Favourites Macros View Help

postgis1 on postgres@localhost:5432

SQL Editor | Graphical Query Builder

Previous queries | Delete | Delete All

```
select * from spatial_ref_sys
where srtext ilike '%sirgas%'
```

Output pane

Data Output | Explain | Messages | History

	srid integer	auth_name character varying(256)	auth_srid integer	srtext character varying(2048)	proj4text character varying(2048)
9	31965	EPSG	31965	PROJCS["SIRGAS 2000 / UTM zone 11N",GEOGCS["SIRGAS 2	+proj=utm +zone=11 +
10	31966	EPSG	31966	PROJCS["SIRGAS 2000 / UTM zone 12N",GEOGCS["SIRGAS 2	+proj=utm +zone=12 +
11	31967	EPSG	31967	PROJCS["SIRGAS 2000 / UTM zone 13N",GEOGCS["SIRGAS 2	+proj=utm +zone=13 +
12	31968	EPSG	31968	PROJCS["SIRGAS 2000 / UTM zone 14N",GEOGCS["SIRGAS 2	+proj=utm +zone=14 +
13	31969	EPSG	31969	PROJCS["SIRGAS 2000 / UTM zone 15N",GEOGCS["SIRGAS 2	+proj=utm +zone=15 +
14	31970	EPSG	31970	PROJCS["SIRGAS 2000 / UTM zone 16N",GEOGCS["SIRGAS 2	+proj=utm +zone=16 +
15	31971	EPSG	31971	PROJCS["SIRGAS 2000 / UTM zone 17N",GEOGCS["SIRGAS 2	+proj=utm +zone=17 +
16	31972	EPSG	31972	PROJCS["SIRGAS 2000 / UTM zone 18N",GEOGCS["SIRGAS 2	+proj=utm +zone=18 +
17	31973	EPSG	31973	PROJCS["SIRGAS 2000 / UTM zone 19N",GEOGCS["SIRGAS 2	+proj=utm +zone=19 +
18	31974	EPSG	31974	PROJCS["SIRGAS 2000 / UTM zone 20N",GEOGCS["SIRGAS 2	+proj=utm +zone=20 +
19	31975	EPSG	31975	PROJCS["SIRGAS 2000 / UTM zone 21N",GEOGCS["SIRGAS 2	+proj=utm +zone=21 +
20	31976	EPSG	31976	PROJCS["SIRGAS 2000 / UTM zone 22N",GEOGCS["SIRGAS 2	+proj=utm +zone=22 +

OK. | Unix | Ln 2, Col 30, Ch 60 | 44 rows. | 127 ms

Criação de tabelas com atributos espaciais

Ocorre em duas etapas:

- Criação da tabela não espacial;
- Adição de coluna espacial;

*SELECT AddGeometryColumn(varchar **table_name**, varchar **column_name**, integer **srid**, varchar **type**, integer **dimension**, boolean **use_typmod**=true);*

ALTER TABLE <nome da tabela>

ADD COLUMN <nome da coluna> geometry(<tipo da geometria>,<SRID>);

Importação de dados

Empregando SQL: geometria como Well-Known Text (WKT);

INSERT INTO <nome da tabela> (coluna 1, geom,..., colunaN)

VALUES (<valor 1>,ST_GeomFromText('LINESTRING(x1 y1,x2 y2)',<srid>), ..., <valor N>);

ST_PointFromText(text WKT, integer srid);

SELECT ST_GeomFromEWKT('SRID=4326;LINESTRING(lon1 lat1,lon2 lat2,lon3 lat3)');

SELECT ST_MakePoint(-71.1043443253471, 42.3150676015829);

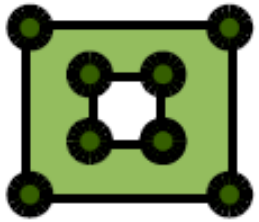
Point

POINT(0 0)

LineString

LINESTRING(0 0, 1 1, 1 2)

Polygon



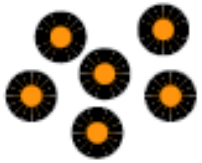
POLYGON**((**0 0, 4 0, 4 4, 0 4, 0 0**)**, **(**1 1, 2 1, 2 2, 1 2,1 1**))**

Anel Externo

Anel Interno

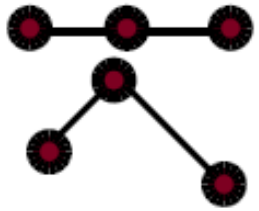
Delimitador Anéis Polígono

MultiPoint



MULTIPOINT(0 0, 1 2, 1 3, 1 4, 2 2, 3 3)

MultiLineString

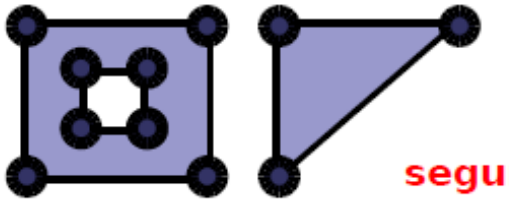


MULTILINESTRING**((**0 0, 1 1, 1 2**)**, **(**2 3, 3 2, 5 4**))**

Primeira Linha

Segunda Linha

MultiPolygon



Primeiro Polígono

MULTIPOLYGON(**(**(0 0, 4 0, 4 4, 0 4, 0 0),
(1 1, 2 1, 2 2, 1 2, 1 1)**)**,
((5 0, 5 4, 6 4, 5 0)**)**)

segundo

terceiro

Importação de dados

Empregando SQL: geometria como Well-Known Binary(WKB);

ST_GeomFromWKB(bytea geom, integer srid);

ST_GeomFromEWKB(bytea EWKB);

Empregando SQL: geometria a partir de campos numéricos (latitude e longitude);

update <nome da tabela>

set <campo da geometria> = st_setsrid(st_makepoint(<longitude>,<latitude>),<srid>)

Importação de dados

Carregando a partir de valores separados por vírgula (CSV);

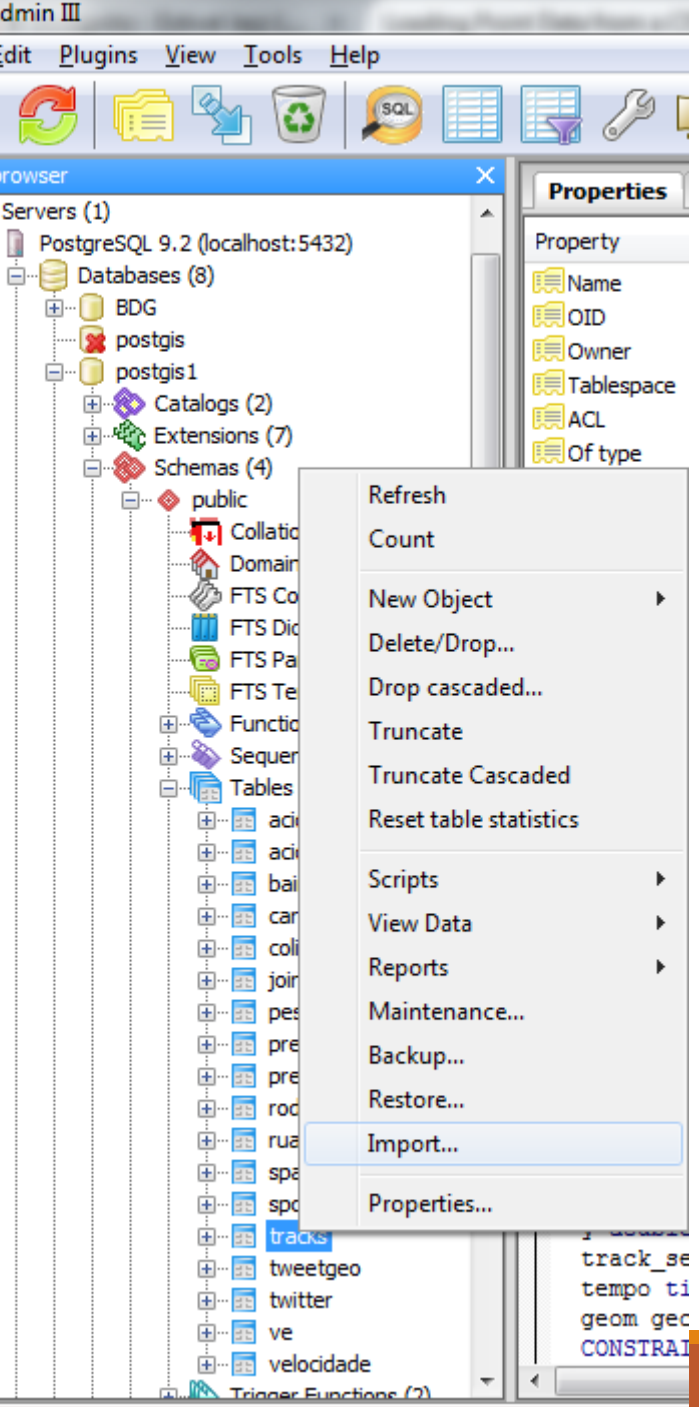
Criar a tabela com a mesma estrutura do arquivo CSV*;*

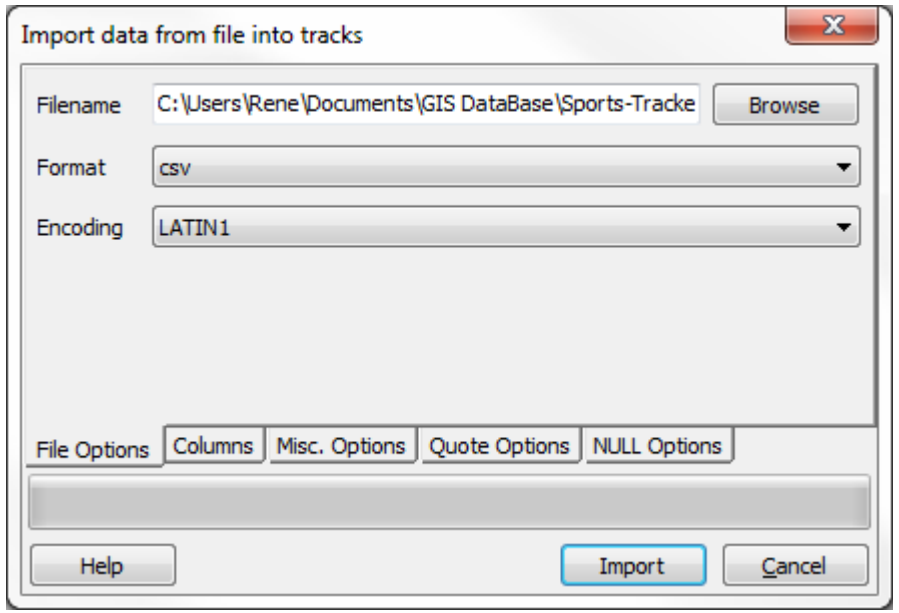
COPY <nome da tabela>(coluna1, …, colunaN)

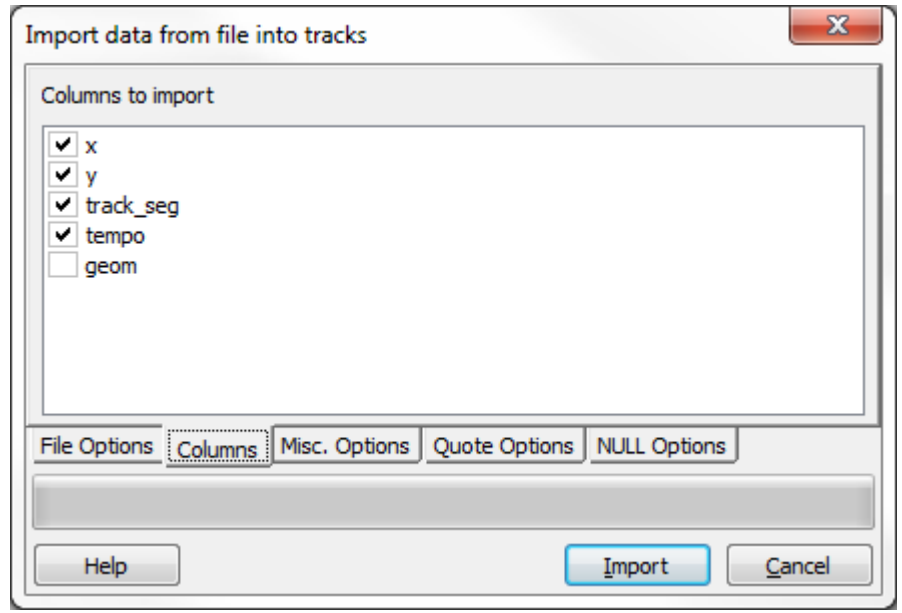
FROM <caminho para o arquivo CSV>

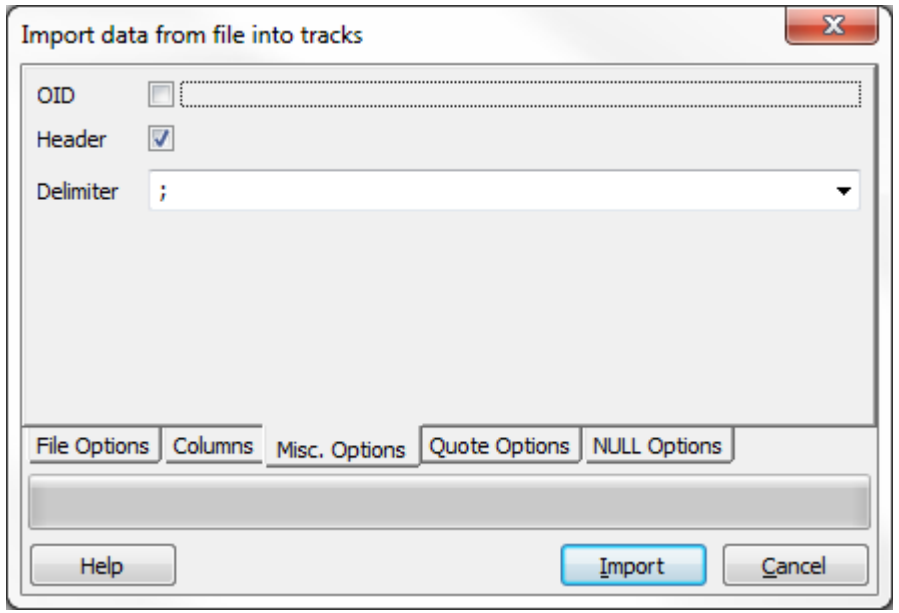
WITH DELIMITER AS ‘,’

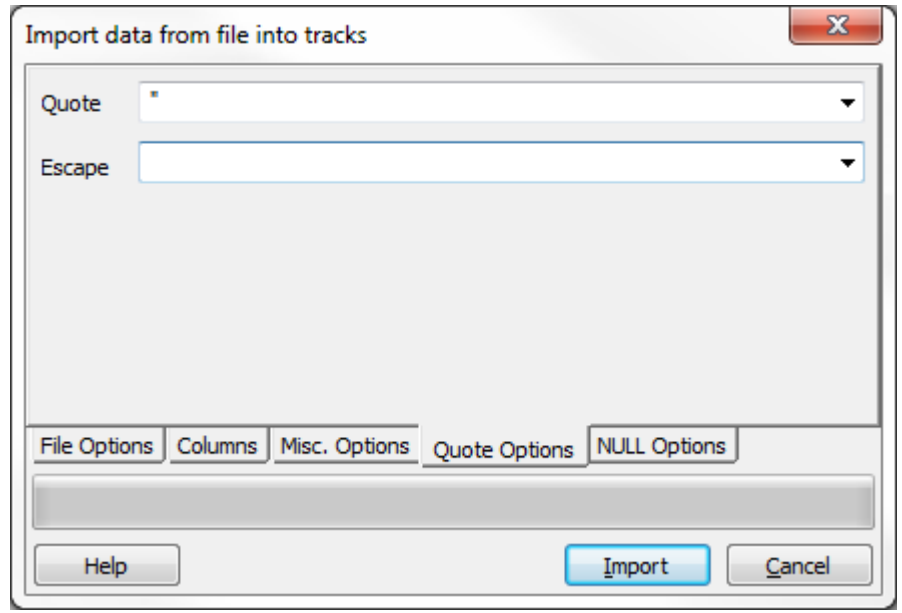
CSV HEADER;


dmin III
Edit Plugins View Tools Help
Browser
Servers (1)
PostgreSQL 9.2 (localhost:5432)
Databases (8)
BDG
postgis
postgis1
Catalogs (2)
Extensions (7)
Schemas (4)
public
Tables
tracks
tweetgeo
twitter
ve
velocidade
Properties
Property
Name
OID
Owner
Tablespace
ACL
Of type
Refresh
Count
New Object
Delete/Drop...
Drop cascaded...
Truncate
Truncate Cascaded
Reset table statistics
Scripts
View Data
Reports
Maintenance...
Backup...
Restore...
Import...
Properties...
ng details on table tracks... Done.


Import data from file into tracks
Filename
C:\Users\Rene\Documents\GIS DataBase\Sports-Tracke
Browse
Format
csv
Encoding
LATIN1
File Options
Columns
Misc. Options
Quote Options
NULL Options
Help
Import
Cancel


Import data from file into tracks
Columns to import
x
y
track_seg
tempo
geom
File Options
Columns
Misc. Options
Quote Options
NULL Options
Help
Import
Cancel


Import data from file into tracks
OID
Header
Delimiter
;
File Options
Columns
Misc. Options
Quote Options
NULL Options
Help
Import
Cancel


Import data from file into tracks
Quote
"
Escape
File Options
Columns
Misc. Options
Quote Options
NULL Options
Help
Import
Cancel

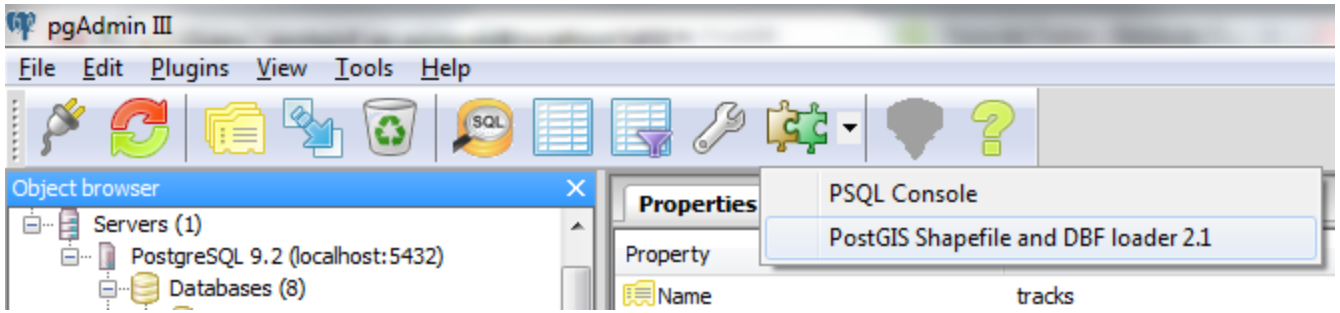
Importação de dados

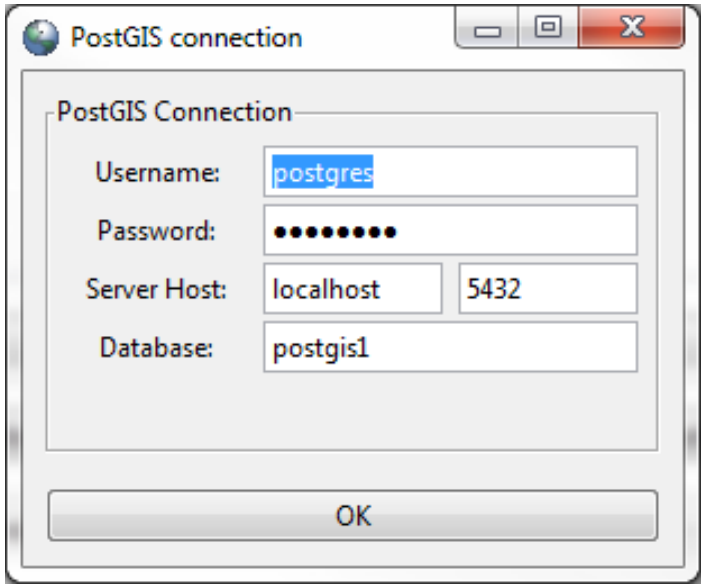
Carregando a partir de *shapefiles*;

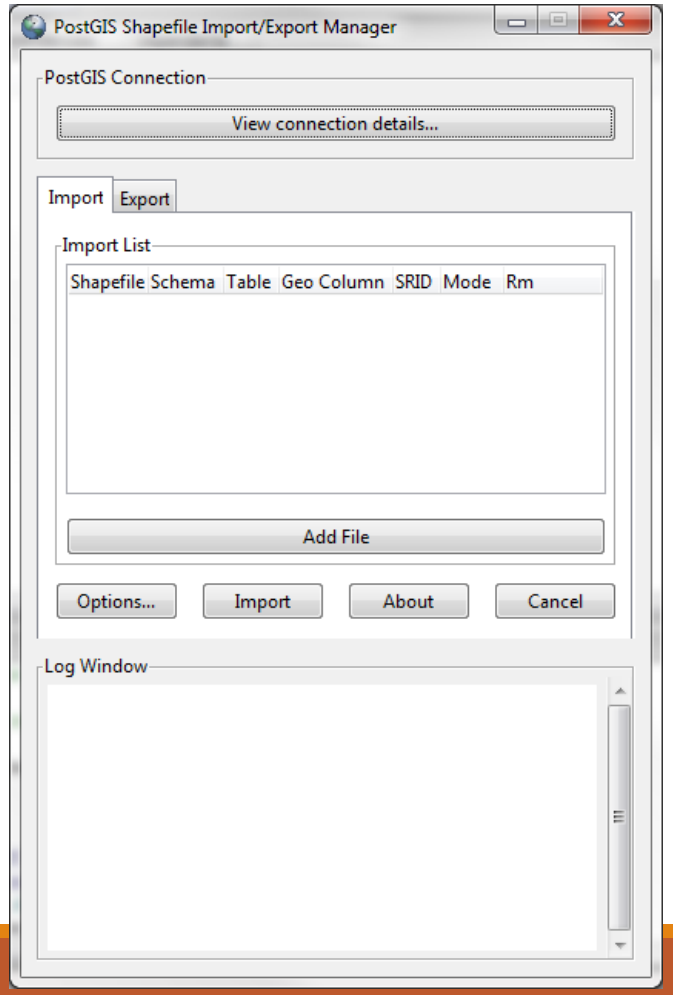
shp2pgsql <arquivo.shp> <nome do esquema>.nome_tabela > tabela.sql;

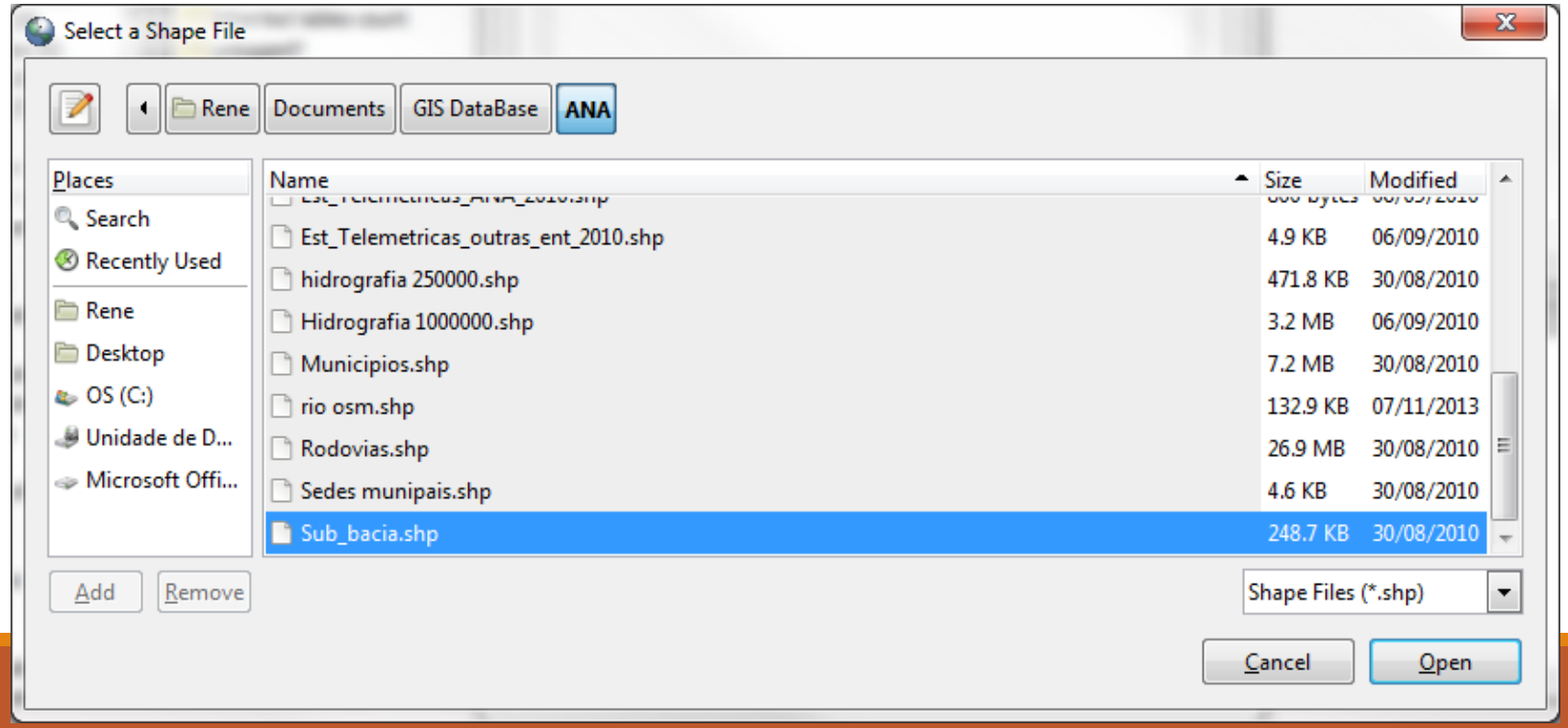
psql -d <nome do banco> -f tabela.sql

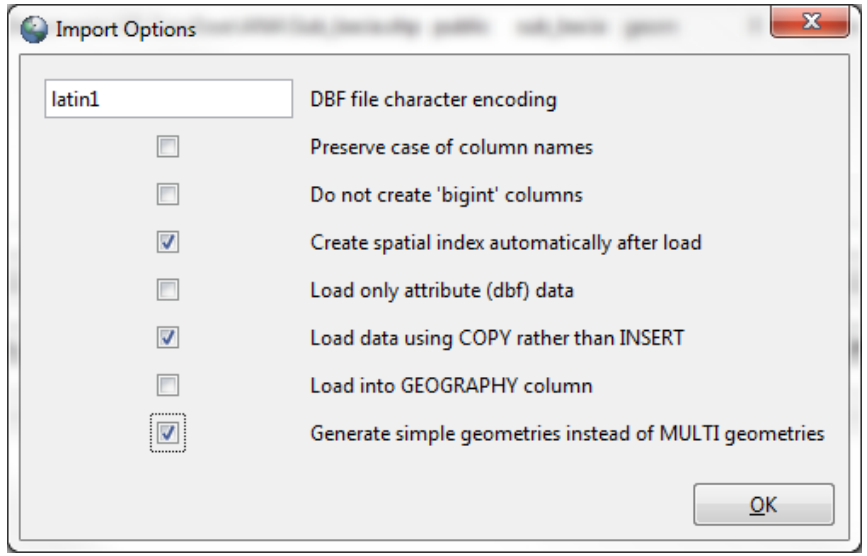
-a	acrescenta dados do *shapefile* ao final da tabela. Múltiplos arquivos;
-D	cria o format de backup do PostgreSQL;
-s<SRID>	cria e preenche as tabelas com o SRID especificado;
-I	cria um índice GiST na coluna da geometria.

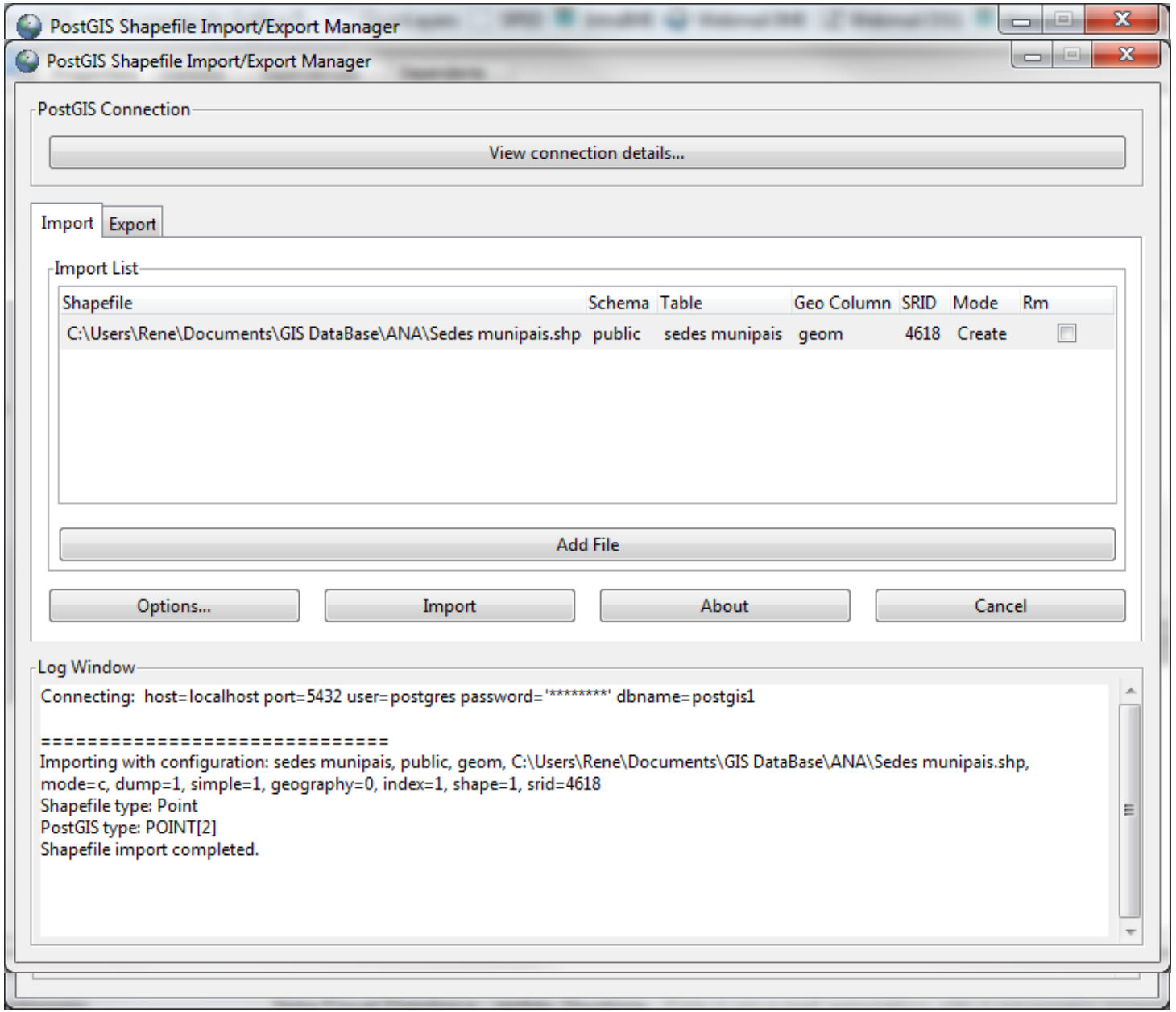

pgAdmin III
File Edit Plugins View Tools Help
Object browser
Servers (1)
PostgreSQL 9.2 (localhost:5432)
Databases (8)
Properties
Property
Name
tracks
PSQL Console
PostGIS Shapefile and DBF loader 2.1


PostGIS connection
PostGIS Connection
Username: postgres
Password: ••••••••
Server Host: localhost 5432
Database: postgis1
OK


PostGIS Shapefile Import/Export Manager
PostGIS Connection
View connection details...
Import Export
Import List
Shapefile Schema Table Geo Column SRID Mode Rm
Add File
Options... Import About Cancel
Log Window


Select a Shape File
Rene Documents GIS DataBase ANA
Places
Search
Recently Used
Rene
Desktop
OS (C:)
Unidade de D...
Microsoft Offi...
Name Size Modified
Est_Telemetricas_outras_ent_2010.shp 4.9 KB 06/09/2010
hidrografia 250000.shp 471.8 KB 30/08/2010
Hidrografia 1000000.shp 3.2 MB 06/09/2010
Municipios.shp 7.2 MB 30/08/2010
rio osm.shp 132.9 KB 07/11/2013
Rodovias.shp 26.9 MB 30/08/2010
Sedes munipais.shp 4.6 KB 30/08/2010
Sub_bacia.shp 248.7 KB 30/08/2010
Add Remove
Shape Files (*.shp)
Cancel Open


Import Options
latin1
DBF file character encoding
Preserve case of column names
Do not create 'bigint' columns
Create spatial index automatically after load
Load only attribute (dbf) data
Load data using COPY rather than INSERT
Load into GEOGRAPHY column
Generate simple geometries instead of MULTI geometries
OK


PostGIS Shapefile Import/Export Manager
PostGIS Shapefile Import/Export Manager
PostGIS Connection
View connection details...
Import
Export
Import List
Shapefile
Schema
Table
Geo Column
SRID
Mode
Rm
C:\Users\Rene\Documents\GIS DataBase\ANA\Sedes munipais.shp
public
sedes munipais
geom
4618
Create
Add File
Options...
Import
About
Cancel
Log Window
Connecting: host=localhost port=5432 user=postgres password='********' dbname=postgis1
==========
Importing with configuration: sedes munipais, public, geom, C:\Users\Rene\Documents\GIS DataBase\ANA\Sedes munipais.shp,
mode=c, dump=1, simple=1, geography=0, index=1, shape=1, srid=4618
Shapefile type: Point
PostGIS type: POINT[2]
Shapefile import completed.

Arquivo *shapefile* carregado no QGIS

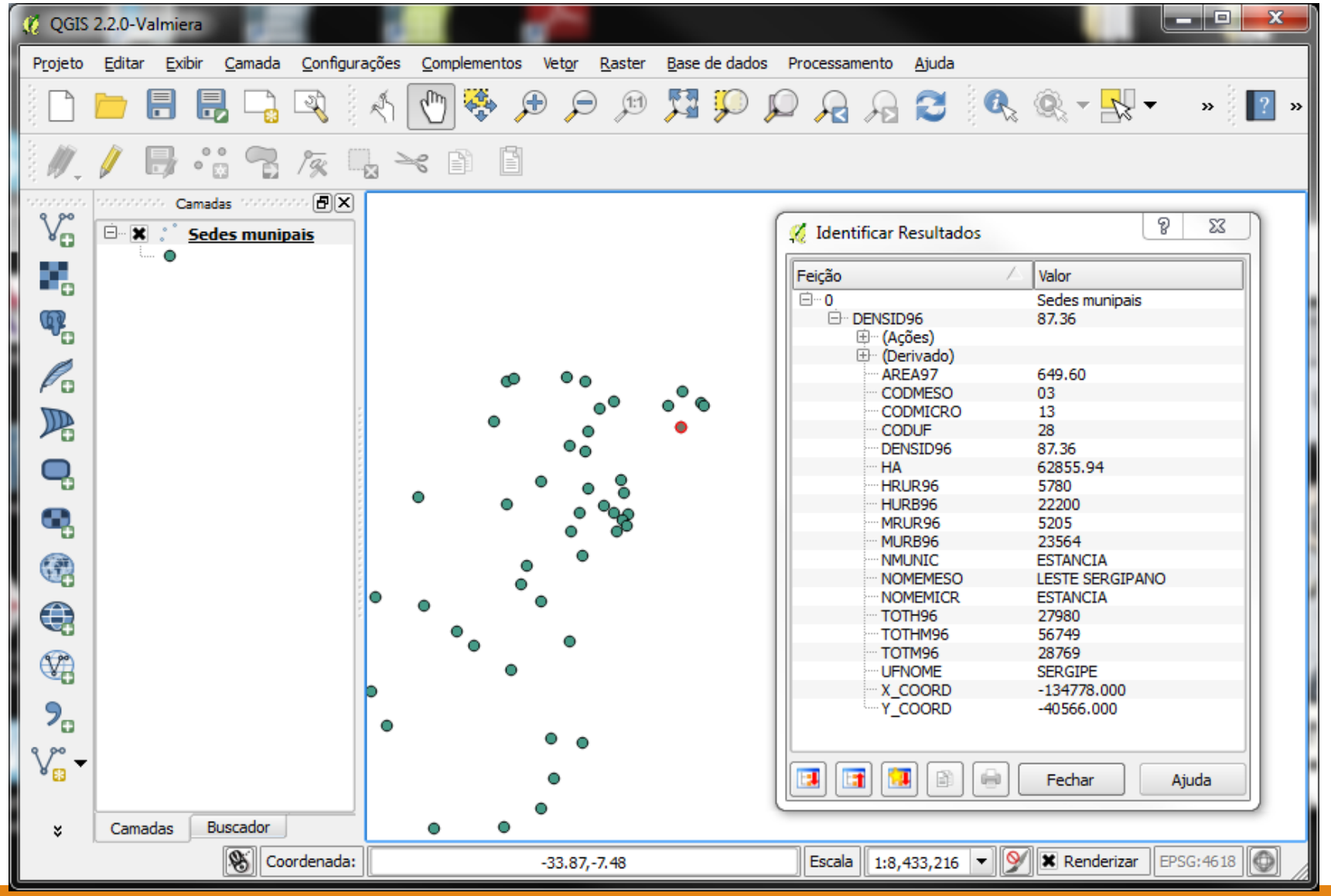
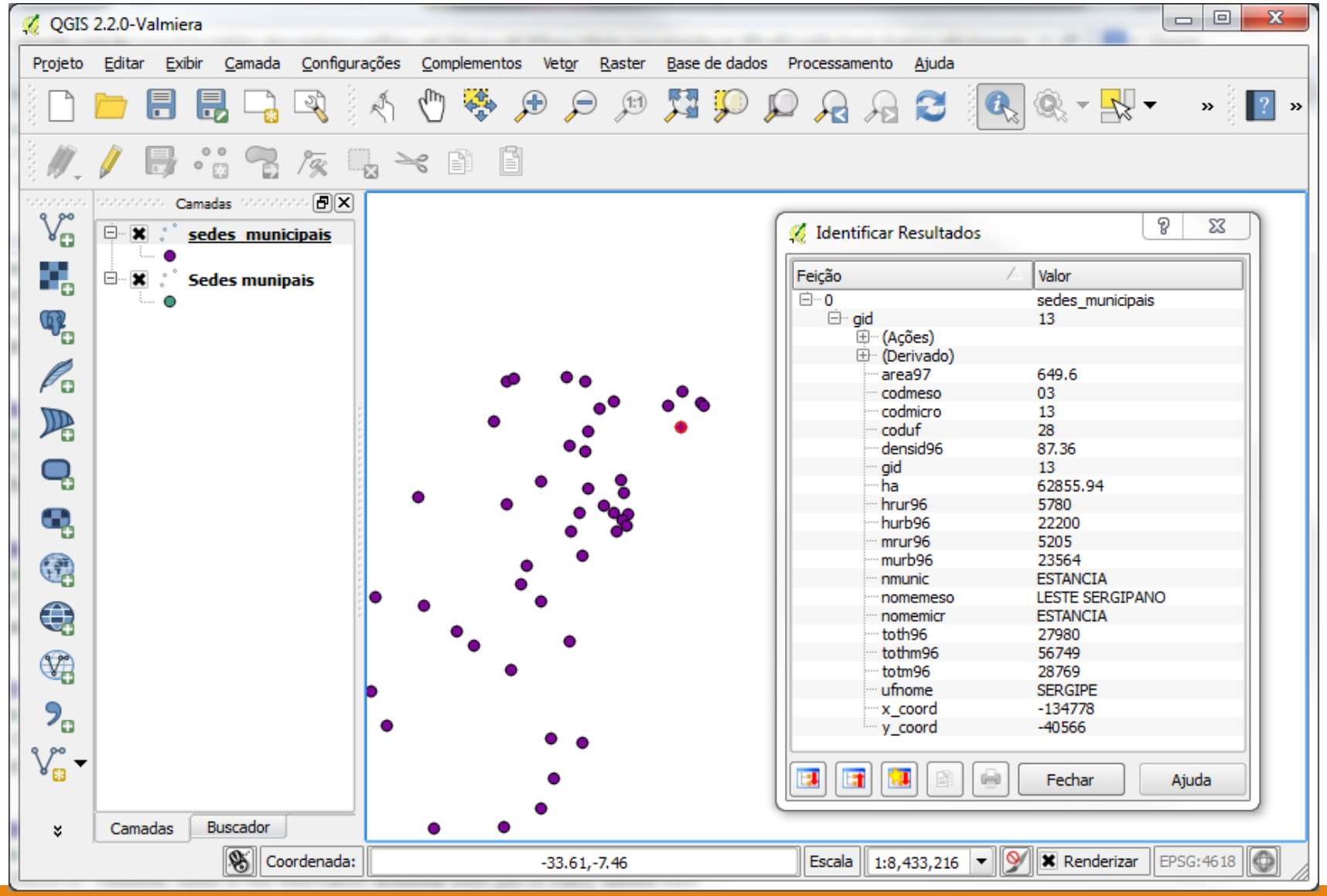


Tabela do PostGIS carregada no QGIS



Índices espaciais

O índices são utilizados pelos SGBD quando se reconhece algum operador na consulta

SELECT * FROM mytable WHERE myname = ‘Paul’;

- **Árvores Binárias Balanceadas:** índices indicados para dados unidimensionais;
- **Árvores R:** Dados são distribuídos por retângulos subdivididos recursivamente;
- **Árvores de Busca Generalizadas (GiST):** estrutura de indexação genérica em que várias estratégias de indexação podem ser implementadas.

CREATE INDEX [indexname]

ON [tablename] USING GIST ([geometryfield])

Simple feature access – SQL option
Rotinas suportadas

Tipo Geometria (unários)

- ST_Dimension;
- ST_GeometryType;
- ST_AsText;
- ST_AsBinary;
- ST_SRID;
- ST_IsEmpty;
- ST_IsSimple;
- ST_Boundary;
- ST_Envelope

Tipo Geometria (binários)

- ST_Equals;
- ST_Disjoint;
- ST_Intersects;
- ST_Touches;
- ST_Crosses;
- ST_Within;
- ST_Contains;
- ST_Overlaps;
- ST_Relate
- ST_Distance

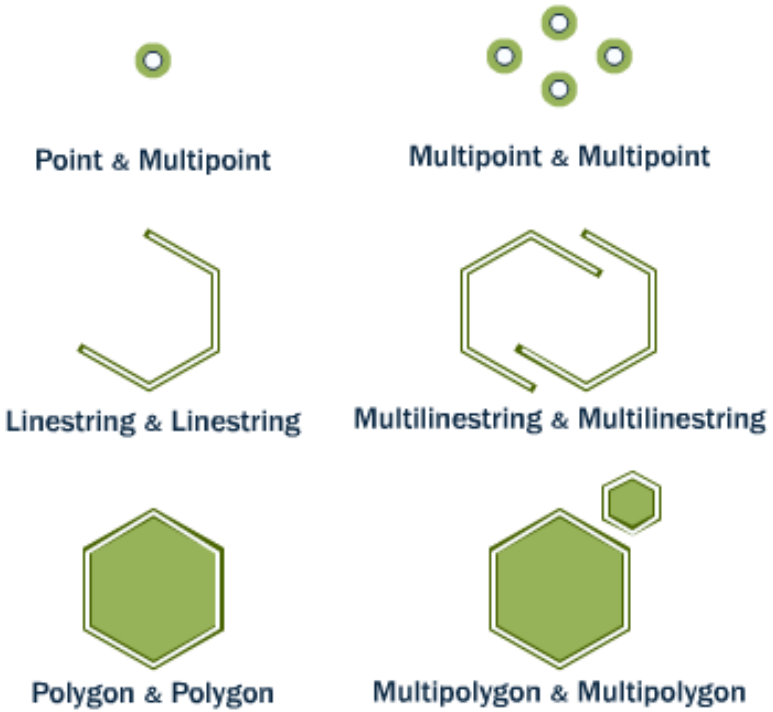
Tipo Geometria (Conjuntos)

- ST_Intersection;
- ST_Difference;
- ST_Union;
- ST_SymDifference;

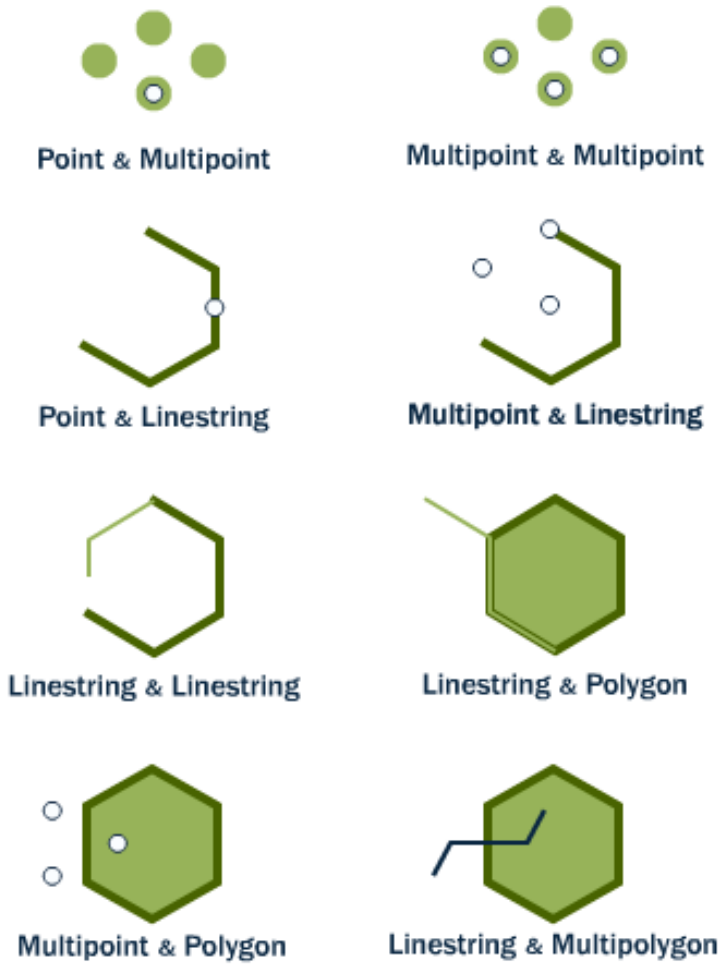
Tipo Geometria (novas geometrias)

- ST_Buffer;
- ST_ConvexHull.

Equals



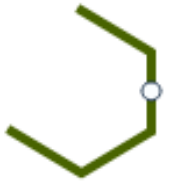
Intersects



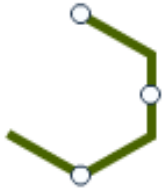
Within/Contains

Point & Multipoint

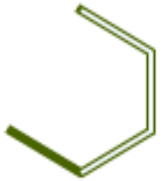
Multipoint & Multipoint



Point & Linestring



Multipoint & Linestring



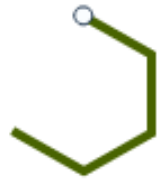
Linestring & Linestring

Linestring & Polygon

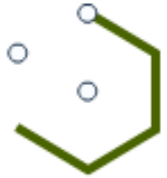
Point & Polygon

Multipoint & Polygon

Touch



Point & Linestring

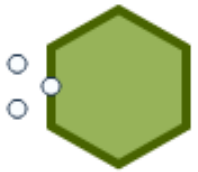


Multipoint & Linestring

Linestring & Linestring

Linestring & Polygon

Point & Polygon

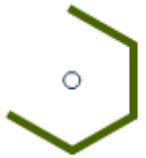


Multipoint & Polygon

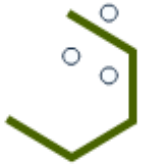
Disjoint

Point & Multipoint

Multipoint & Multipoint

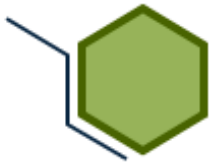


Point & Linestring



Multipoint & Linestring

Linestring & Linestring



Linestring & Polygon

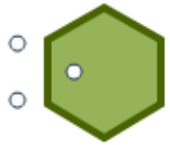
Multipoint & Polygon

Polygon & Polygon

Cross

Multipoint & Linestring

Linestring & Linestring

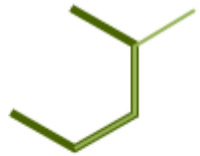


Multipoint & Polygon

Linestring & Multipolygon

Overlap

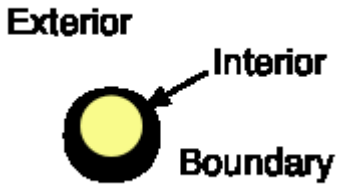
Multipoint & Multipoint

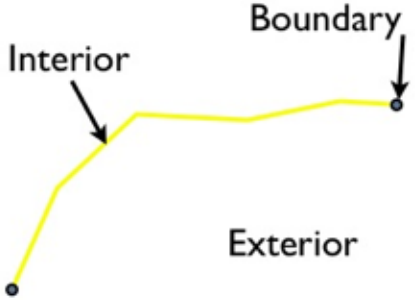


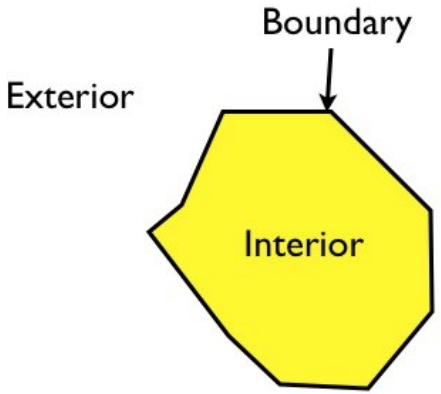
Linestring & Linestring

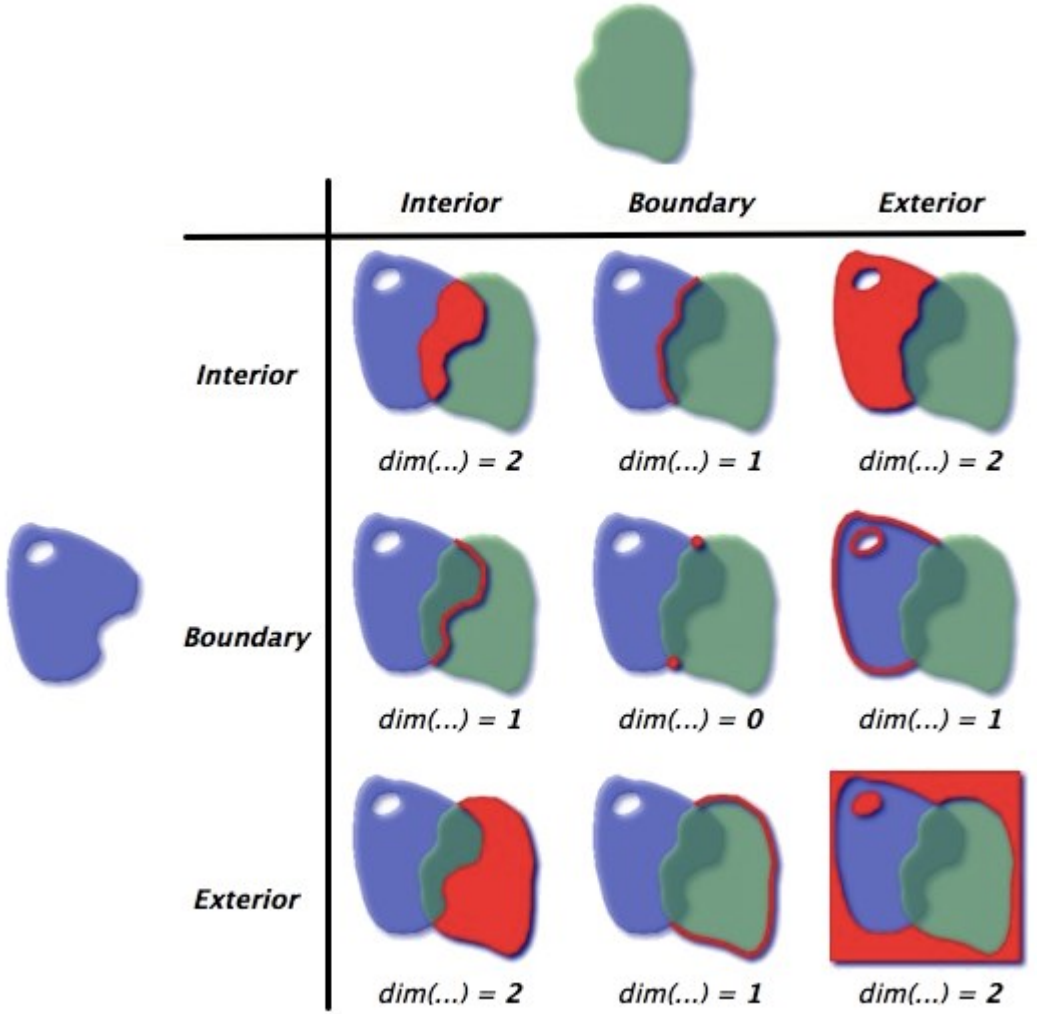
Polygon & Polygon

ST_Relate





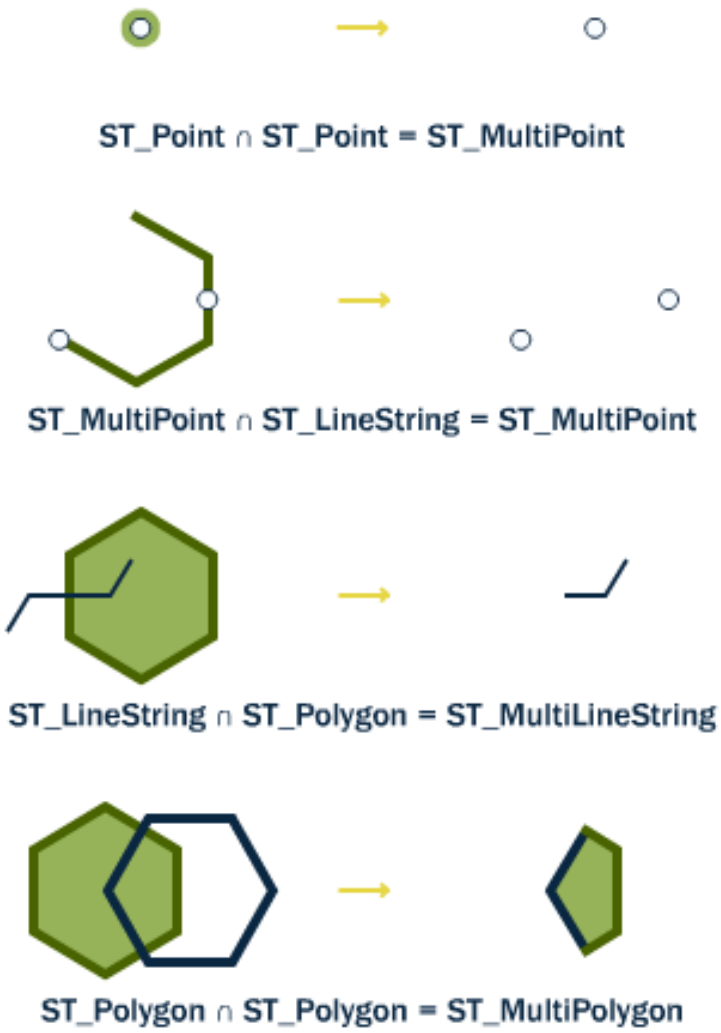




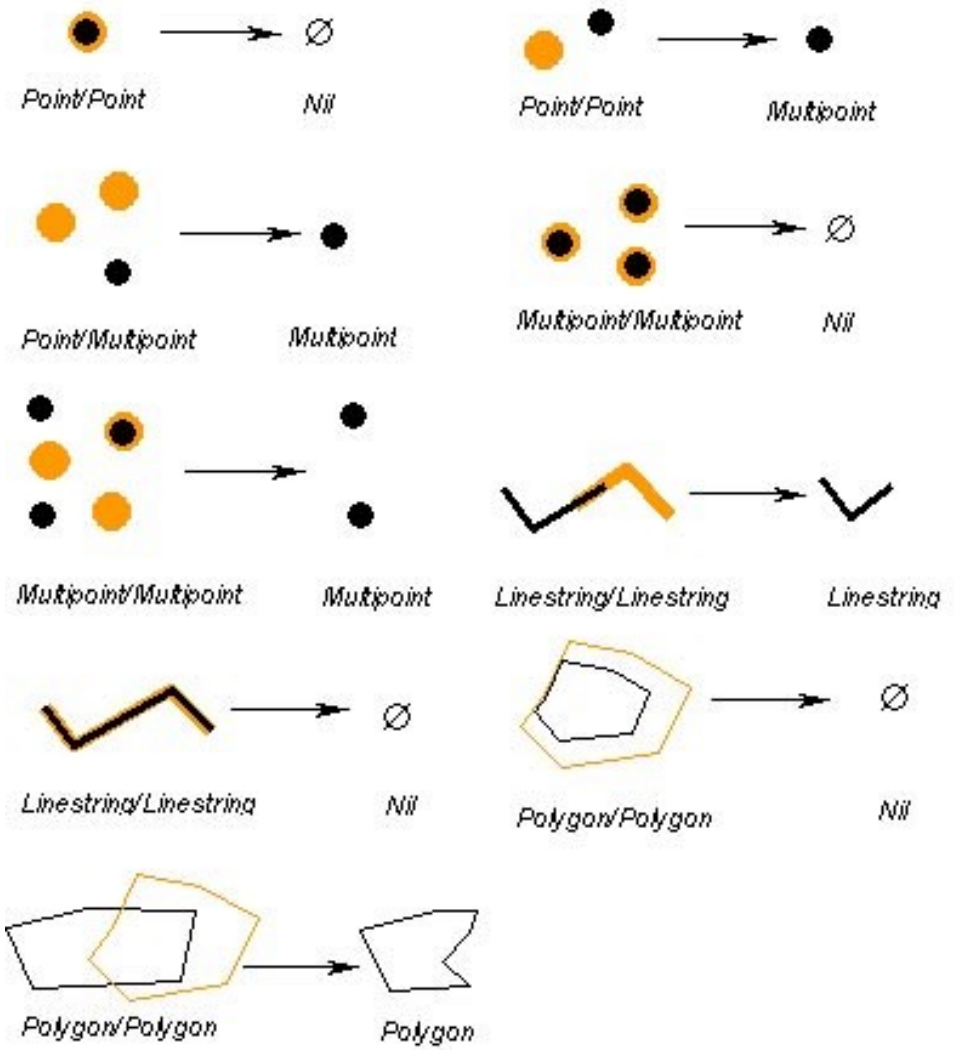
Exemplo

	I	B	E
I	1	F	F
B	0	0	F
E	2	1	2

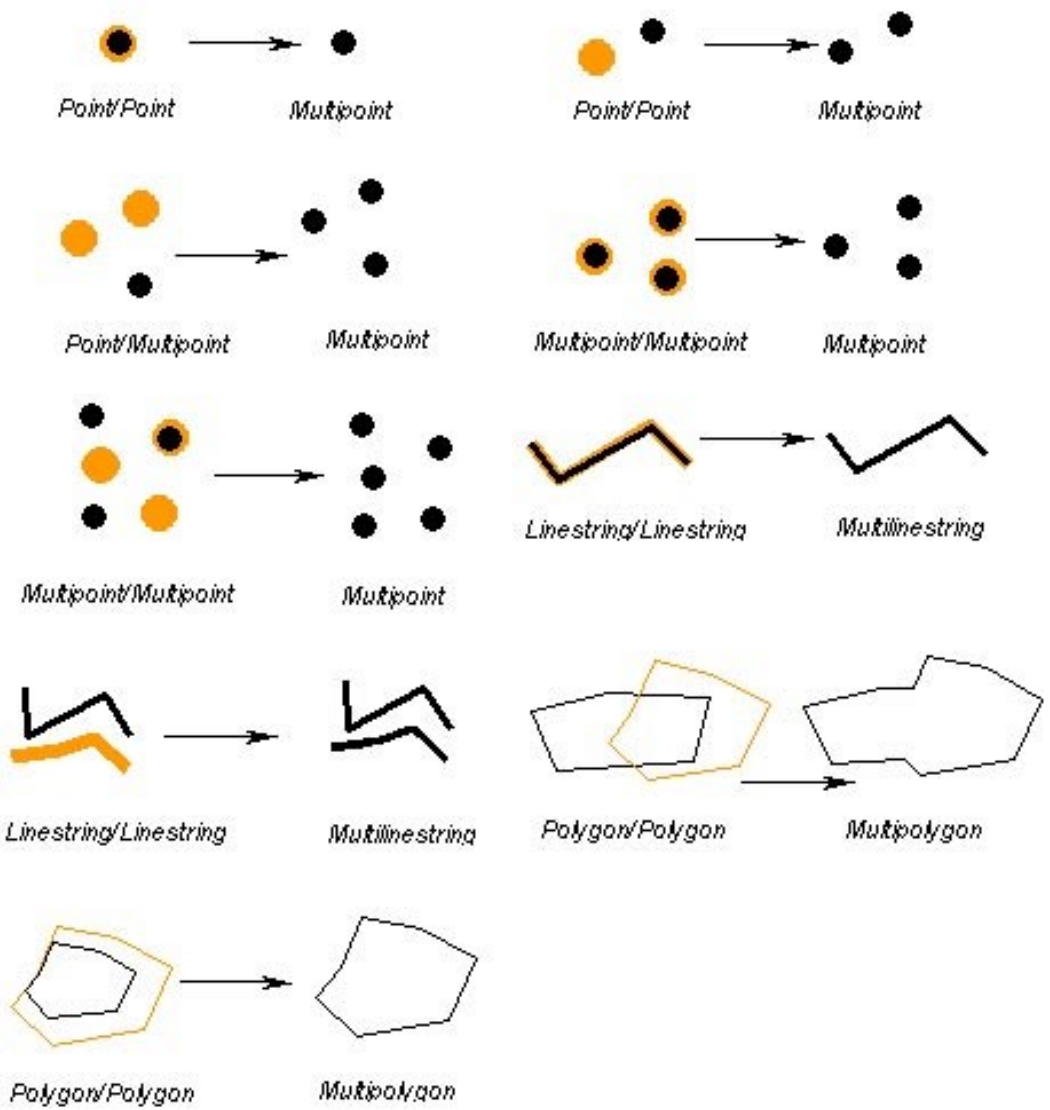
ST_Intersection



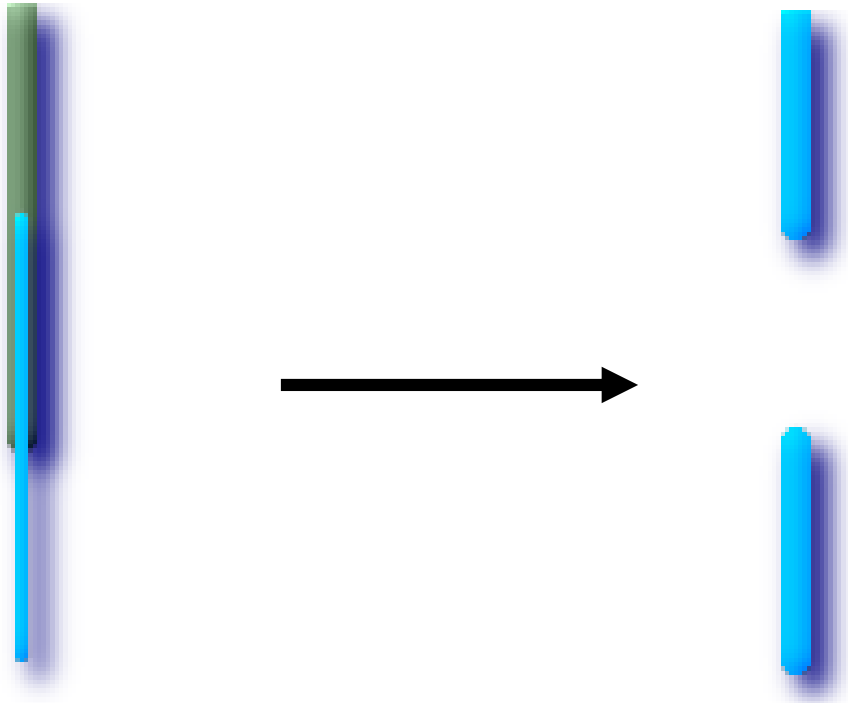
Difference



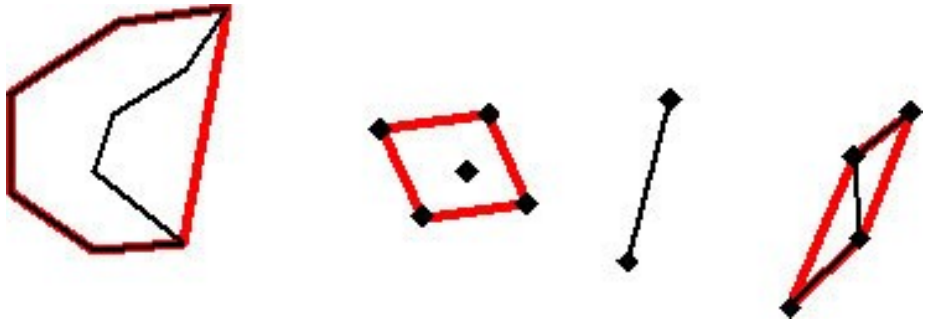
Union



Symmetric difference



Convex Hull



ST_Buffer

Buffering a point

Buffering a multipoint

Buffering a linestring

Buffering a polygon with one interior ring

Simple feature access – SQL option
Rotinas suportadas

Tipo Ponto

- ST_X;
- ST_Y;
- ST_Z;
- ST_M;
- Todos do tipo geometria.

Tipo Curva

- ST_StartPoint;
- ST_EndPoint;
- ST_IsRing;
- ST_Length;
- Todos do tipo geometria.

Tipo Linha

- ST_NumPoints;
- ST_PointN;
- Todos do tipo curva.

Tipo Polígono

- ST_ExteriorRing;
- ST_NumInteriorRing;
- ST_InteriorRingN;
- Todos do tipo geometria.

Tipo Superfície

- ST_Centroid;
- ST_PointOnSurface;
- ST_Area;
- Todos do tipo geometria.

Simple feature access – SQL option
Rotinas suportadas

Tipo Coleção de Geometrias

- ST_NumGeometries;
- ST_GeometryN.

Como Coleção de Geometrias:

- MultiPontos;
- Multilinhas;

Tipo Multicurvas

- ST_IsClosed;
- ST_Length;
- Todas do tipo Coleção de Geometrias.

Tipo Superfície

- ST_Centroid;
- ST_PointOnSurface;
- ST_Area;
- Todos do tipo coleção de geometrias.

Consultas SQL

Saídas de Geometria

ST_AsBinary

ST_AsEWKB

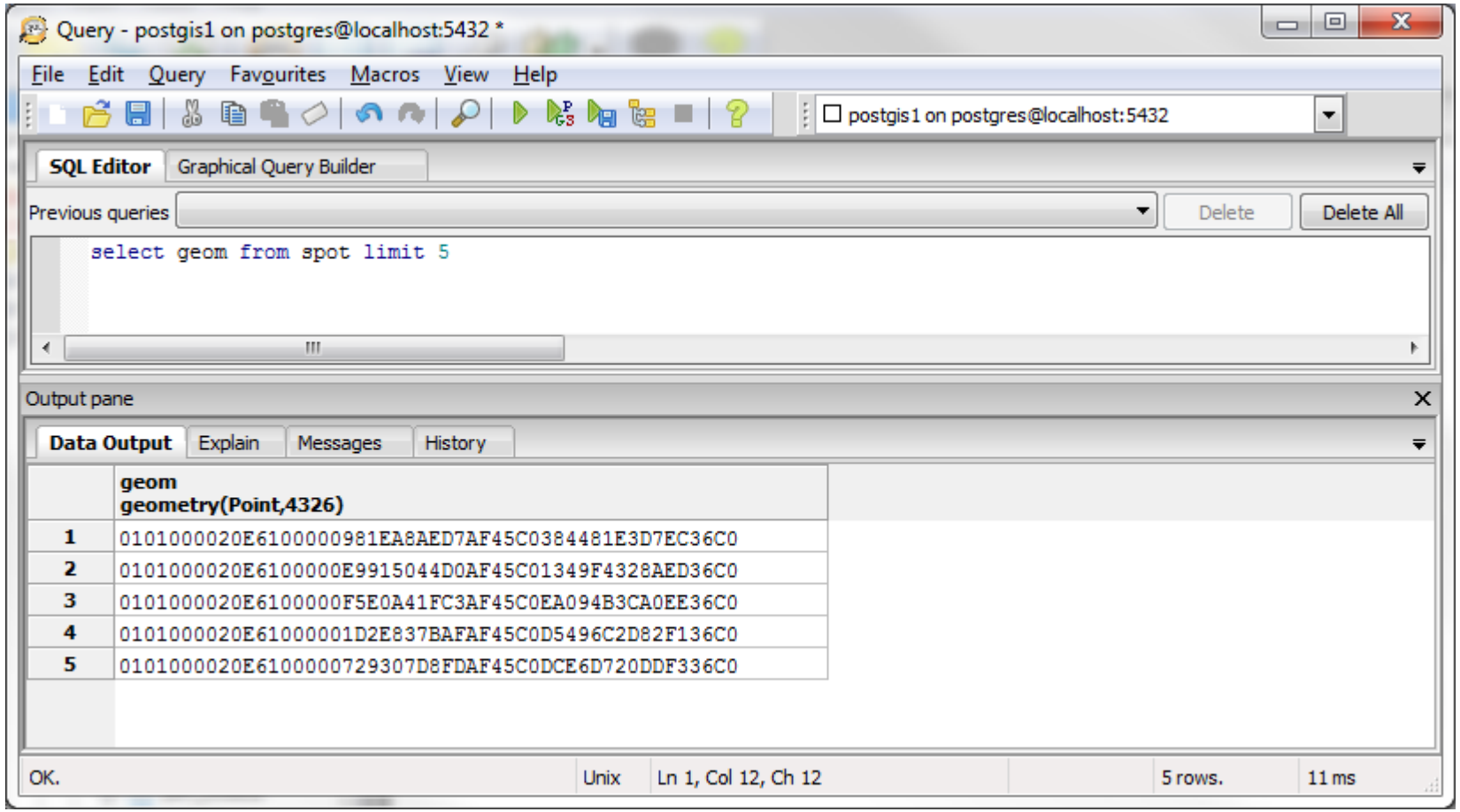
ST_AsEWKT

ST_AsGeoJSON

ST_AsGML

ST_AsHEXEWKB

```
SELECT ST_ASBINARY(GEOM)
FROM SPOT
```



Consultas SQL

Saídas de Geometria

ST_AsKML

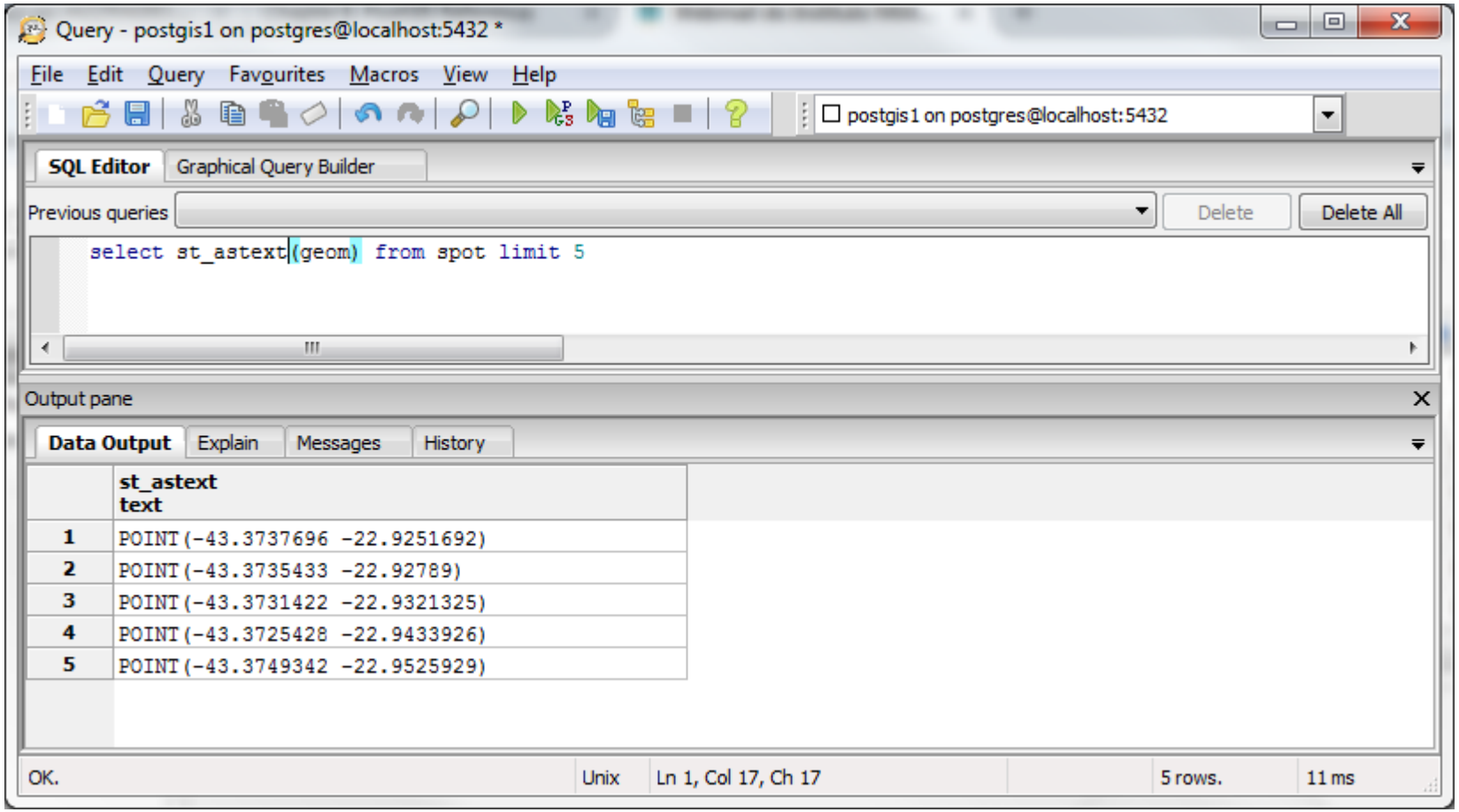
ST_AsSVG

ST_AsX3D

ST_GeoHash

ST_AsText

ST_AsLatLonText

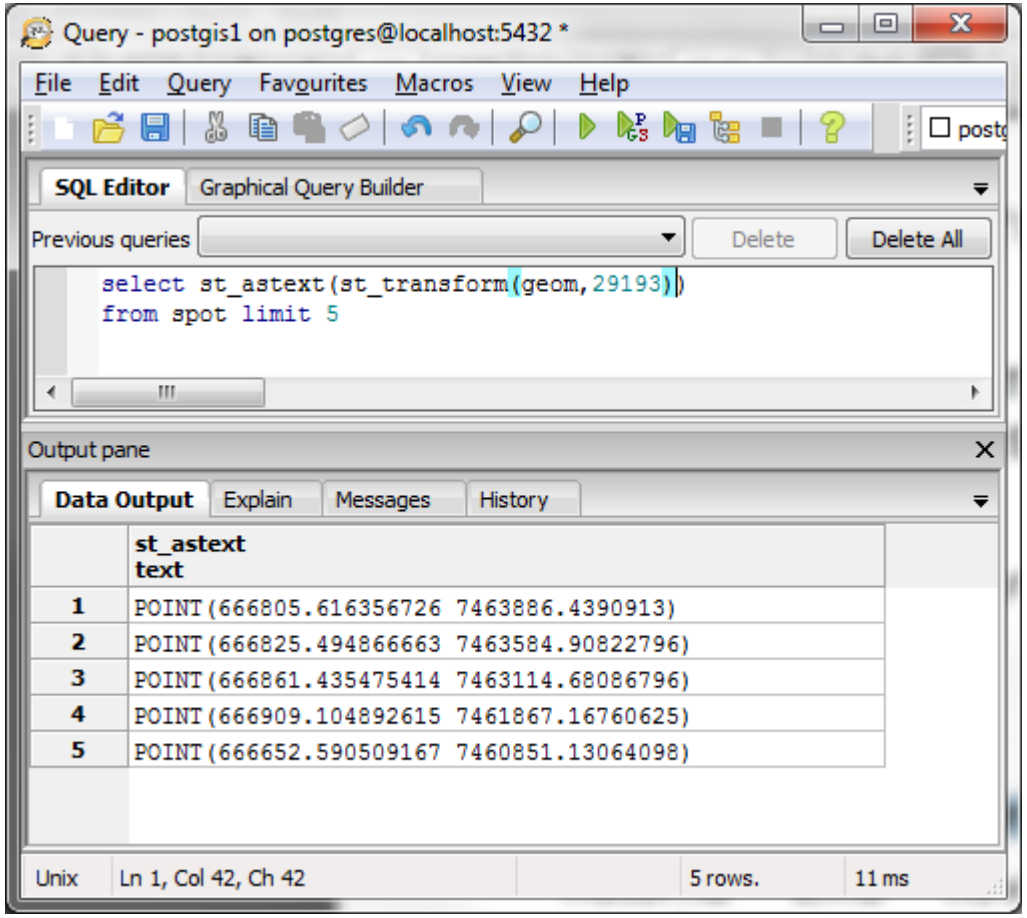


Consultas SQL

Suporte a projeções:
ST_Transform(geometria, srid)

Algumas operações só podem ser realizadas se as geometrias estiverem no mesmo Sistema de Coordenadas.

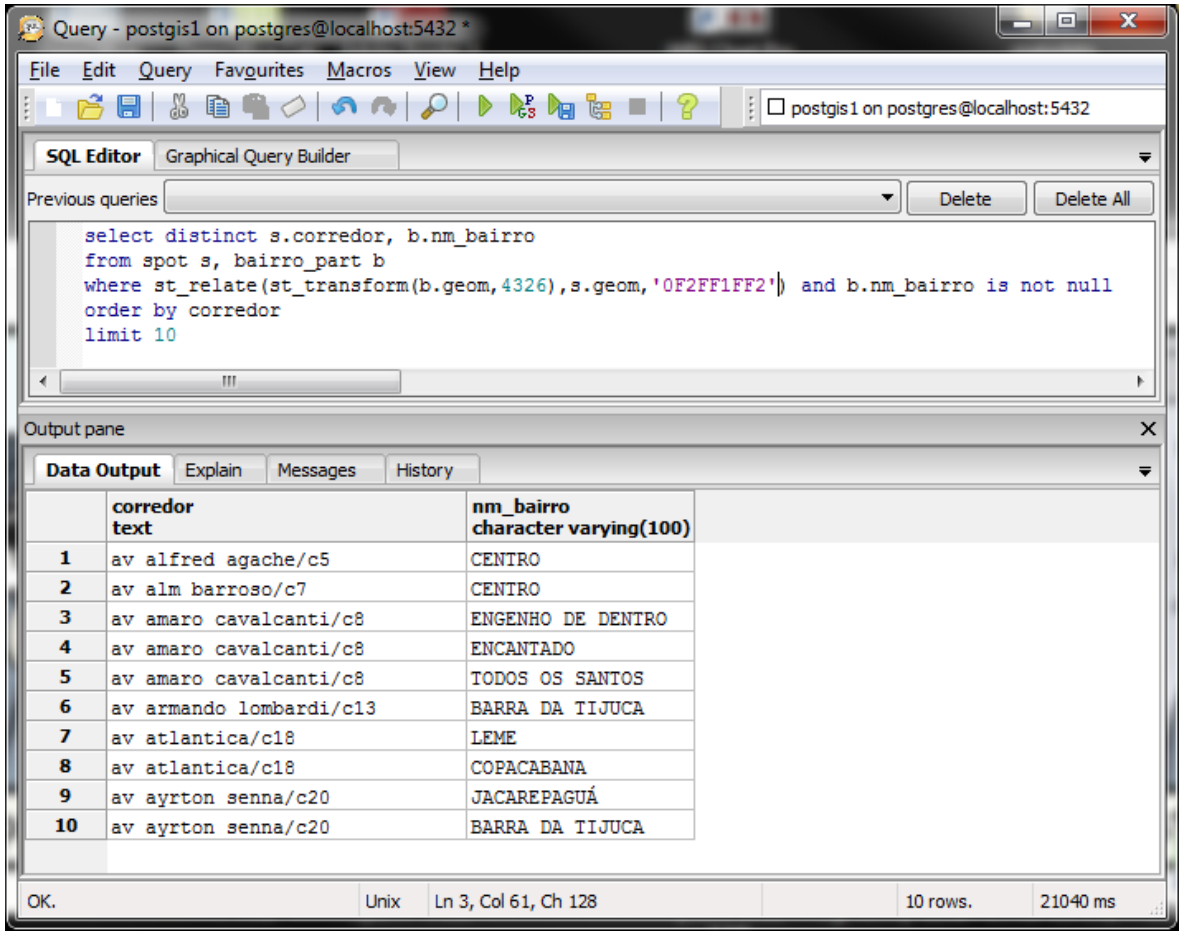
Distâncias e áreas são dadas na unidade do Sistema, podendo ser em comprimento de arco ou área elipsoidica. É necessário projetar a geometria se desejamos comprimento (ou área) em metros (quadrados).

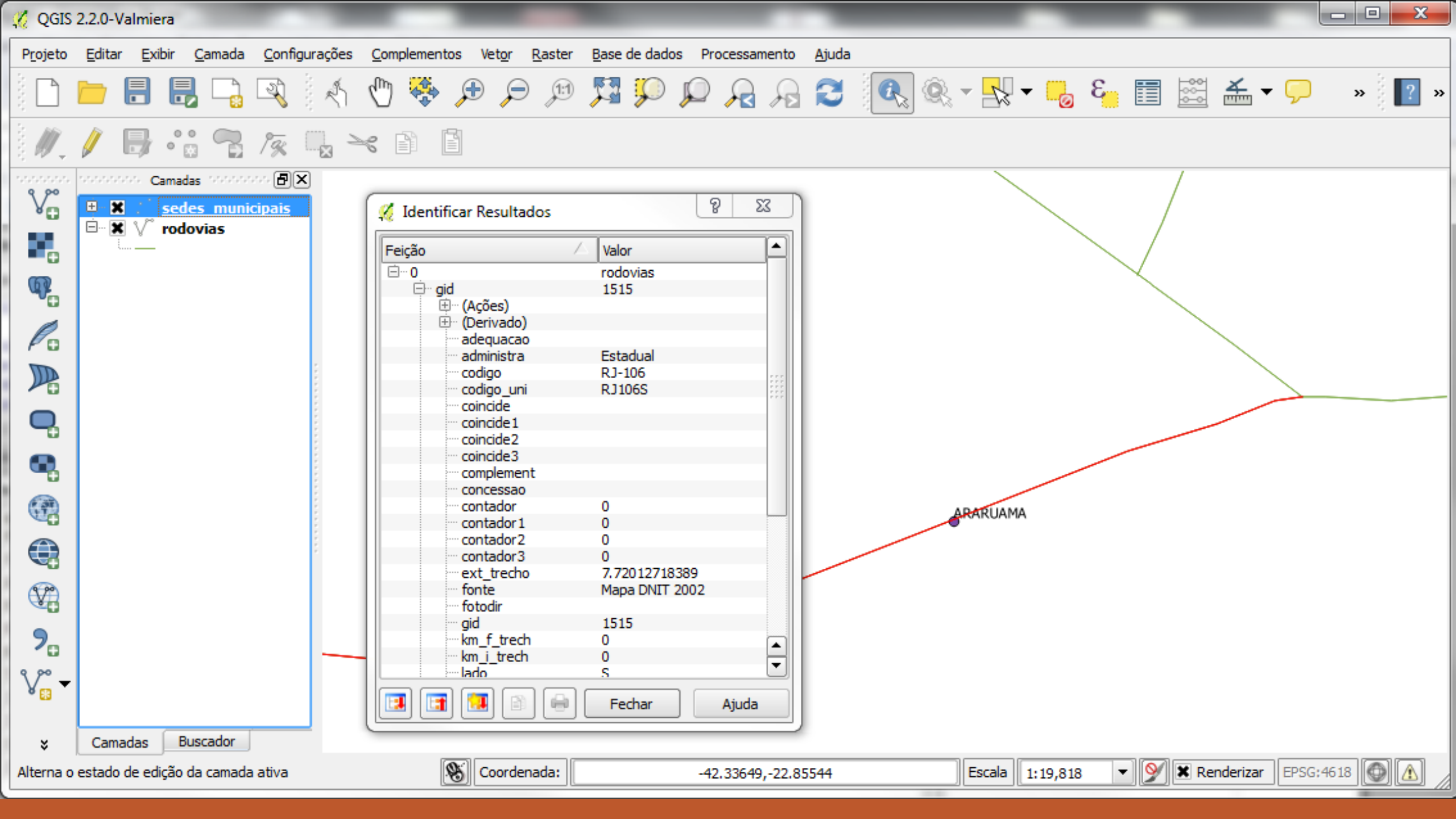


Consultas SQL

Condições espaciais

Nas cláusulas WHERE e JOIN ON podem ser empregados operadores e predicados espaciais em conjunto com as operações que empregam apenas dados alfanuméricos.

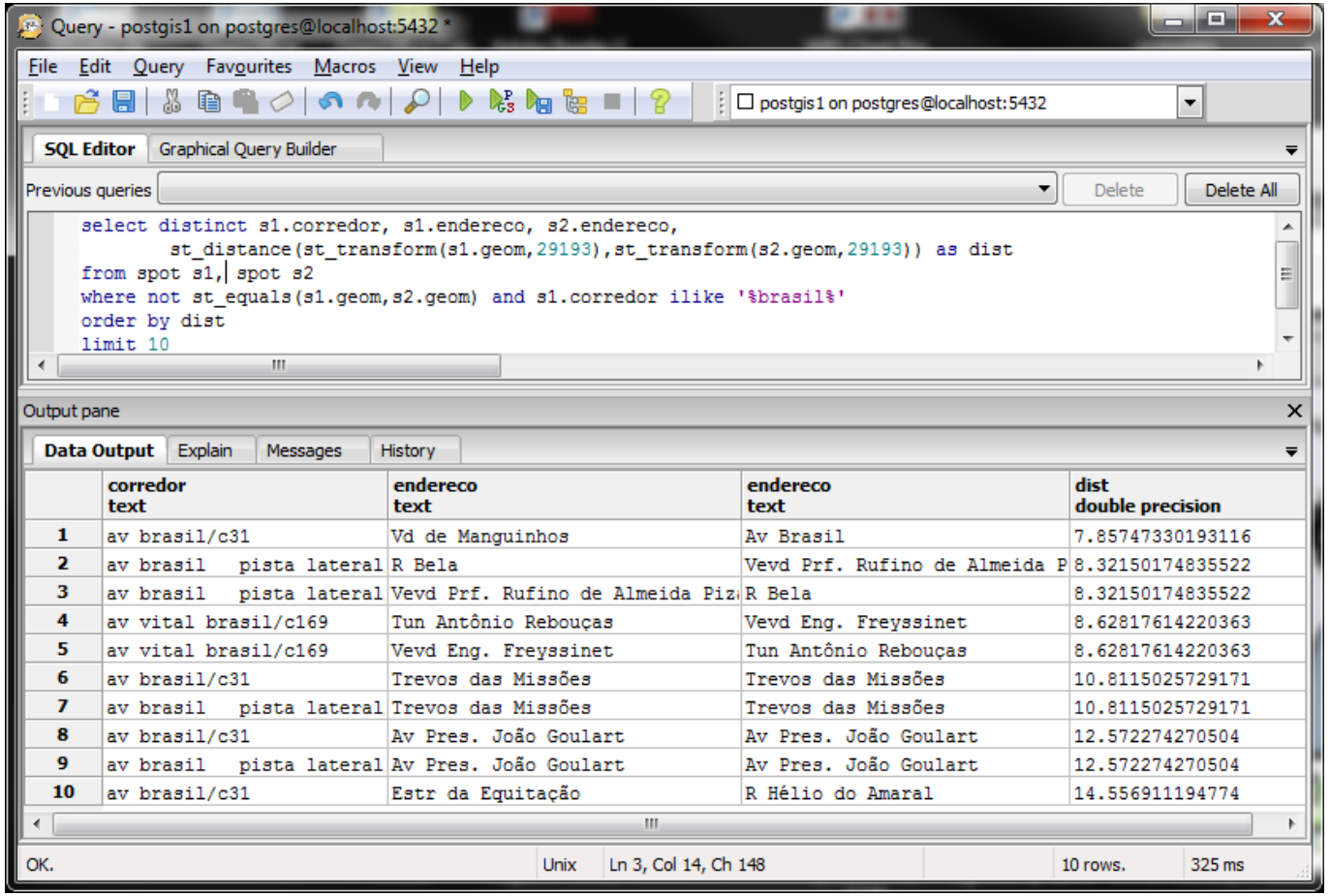



QGIS 2.2.0-Valmiera
Projeto Editar Exibir Camada Configurações Complementos Vetor Raster Base de dados Processamento Ajuda
Camadas
sedes_municipais
rodovias
Identificar Resultados
Feição
Valor
0
rodovias
gid
1515
(Ações)
(Derivado)
adequacao
administra
Estadual
codigo
RJ-106
codigo_uni
RJ106S
coincide
coincide1
coincide2
coincide3
complement
concessao
contador
0
contador1
0
contador2
0
contador3
0
ext_trecho
7.72012718389
fonte
Mapa DNIT 2002
fotodir
gid
1515
km_f_trech
0
km_i_trech
0
lado
S
Fechar
Ajuda
ARARUAMA
Camadas
Buscador
Alterna o estado de edição da camada ativa
Coordenada:
-42.33649,-22.85544
Escala
1:19,818
Renderizar
EPSG:4618

Consultas SQL

Condições espaciais

O uso de *alias* pode ser empregado para confrontar duas instâncias independentes da mesma tabela.



Consultas SQL

Funções Janela

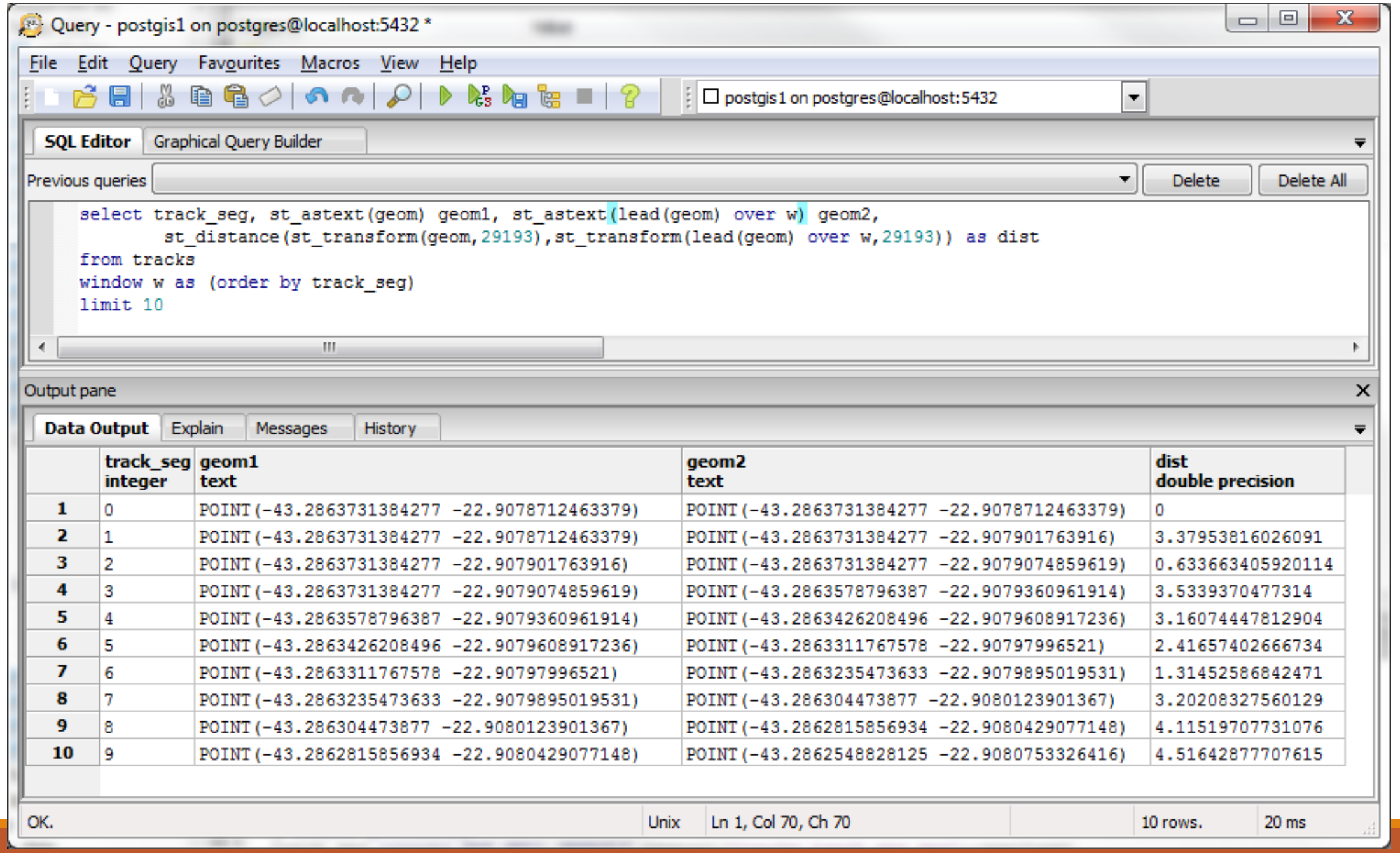
Definidas no padrão SQL:2003, a fim de potencializar as consultas em contextos OLAP, as funções de janela executam cálculos sobre um conjunto de tuplas que possuem algum tipo de relação. A tabela é particionada de acordo com um dos campos e os cálculos são realizados dentro de cada partição.

SELECT col1, col2, …, coln OVER w

FROM tabela

Window w as (PARTITION BY col(i) ORDER BY col(j) DESC)

Principais funções: funções agregadas, *rank()*, *first_value*, *last_value*, *lead*, *lag*, etc.


Query - postgis1 on postgres@localhost:5432 *
File Edit Query Favourites Macros View Help
postgis1 on postgres@localhost:5432
SQL Editor
Graphical Query Builder
Previous queries
Delete
Delete All
select track_seg, st_astext(geom) geom1, st_astext(lead(geom) over w) geom2,
st_distance(st_transform(geom,29193),st_transform(lead(geom) over w,29193)) as dist
from tracks
window w as (order by track_seg)
limit 10
Output pane
Data Output
Explain
Messages
History
track_seg integer
geom1 text
geom2 text
dist double precision
1 0 POINT(-43.2863731384277 -22.9078712463379) POINT(-43.2863731384277 -22.9078712463379) 0
2 1 POINT(-43.2863731384277 -22.9078712463379) POINT(-43.2863731384277 -22.907901763916) 3.37953816026091
3 2 POINT(-43.2863731384277 -22.907901763916) POINT(-43.2863731384277 -22.9079074859619) 0.633663405920114
4 3 POINT(-43.2863731384277 -22.9079074859619) POINT(-43.2863578796387 -22.9079360961914) 3.5339370477314
5 4 POINT(-43.2863578796387 -22.9079360961914) POINT(-43.2863426208496 -22.9079608917236) 3.16074447812904
6 5 POINT(-43.2863426208496 -22.9079608917236) POINT(-43.2863311767578 -22.90797996521) 2.41657402666734
7 6 POINT(-43.2863311767578 -22.90797996521) POINT(-43.2863235473633 -22.9079895019531) 1.31452586842471
8 7 POINT(-43.2863235473633 -22.9079895019531) POINT(-43.286304473877 -22.9080123901367) 3.20208327560129
9 8 POINT(-43.286304473877 -22.9080123901367) POINT(-43.2862815856934 -22.9080429077148) 4.11519707731076
10 9 POINT(-43.2862815856934 -22.9080429077148) POINT(-43.2862548828125 -22.9080753326416) 4.51642877707615
OK.
Unix
Ln 1, Col 70, Ch 70
10 rows.
20 ms