

Banco de Dados Geográficos

316504 - Tópicos em Fund. Met. Computação

Programa de Posgraduação em Informática – PPGInf/CIC

+

Modelo objeto-relacional



+



+

Modelo objeto-relacional



Tabelas

Relacionamentos

+

Modelo objeto-relacional



Objetos

Característica de
Orientação Objeto

+

Modelo Relacional

TB_VEICULO

placa [PK] character	cpf character vari	codigo_mode integer	codigo_categ integer	chassi character vari	cor character vari	ano character(4)
FRS4567	98050346153	4	3	SDFSD43535G	PRATA	2007
JFA5849	69904723168	3	4	SDFSDFH645	BRANCA	2010
JHH1038	69904723168	1	2	SMDFSIDFJI3	PRATA	2010
JNH8467	12345678909	2	2	SDGSDHJDJER	VERMELHA	2011
SVX5930	12345678909	5	1	544543534YE	PRETA	2009

TB_MODELO

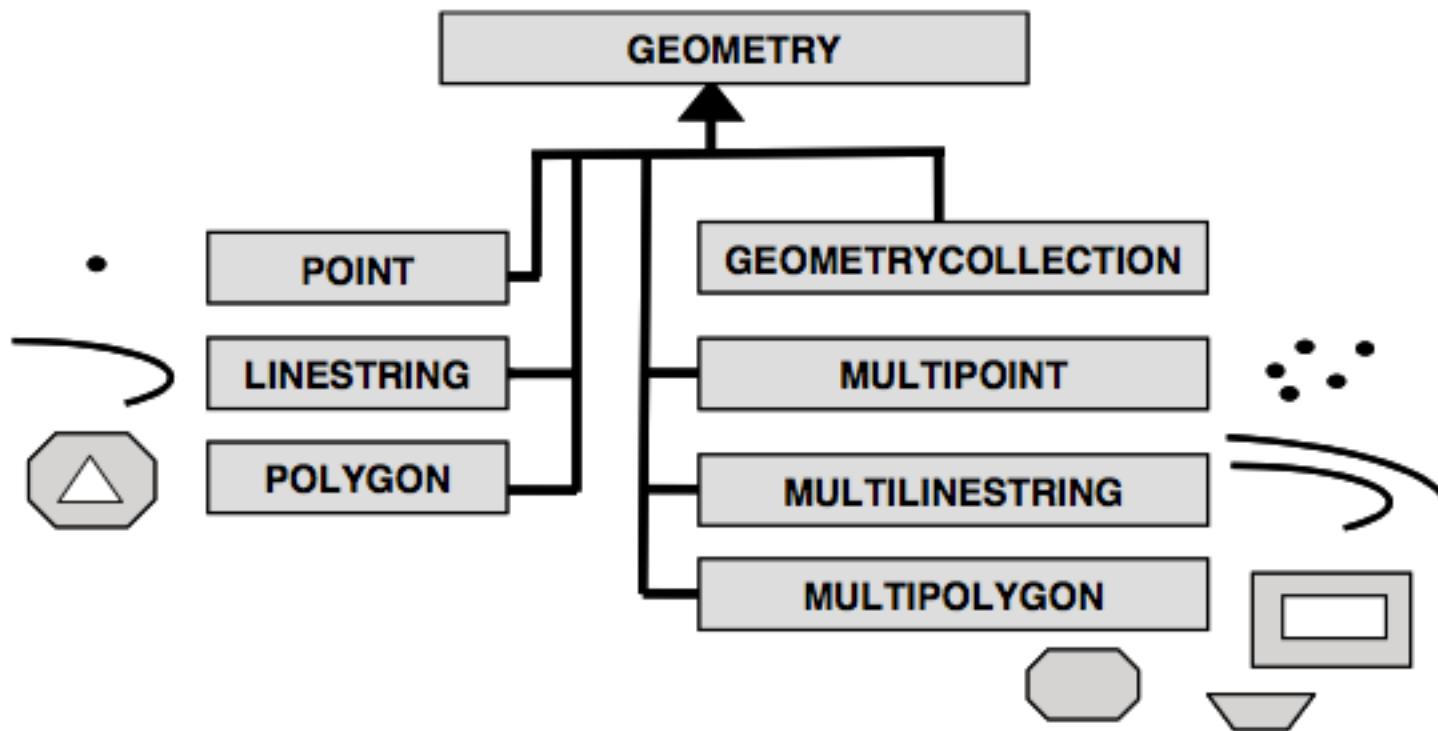
codigo_mode [PK] serial	nome character vari
1	FORD FOCUS
2	VW GOLF 1.6
3	AUDI A4 2.0
4	FORD F250
5	TOYOTA HILU
6	MITSUBISHI

TB_CATEGORIA

codigo_categoria [PK] serial	nome character vari
1	UTILITÁRIO
2	PASSEIO
3	PESADO
4	ESPORTIVO

+

Modelo Orientado a Objetos





Modelo de Dados Geográfico

codigo_municipio [PK] serial	codigo_uf integer	nome character varying(100)	codigo_ibge character varying(10)	the_geom geometry
29	35	UNIAO PAULISTA	3555700	010300000000
30	43	Formigueiro	4308410	010300000000
31	42	Princesa	4214150	010300000000
32	41	Cafezal do Sul	4103480	010300000000
33	41	Bom Jesus do Sul	4103160	010300000000
34	42	São João do Oeste	4216260	010300000000
35	43	Ivorá	4310750	010300000000
36	22	São Julião	2210300	010300000000



+ Mapeamento entre o conceitual e o Modelo de Implementação

Uma Entidade

Uma Tabela

Relacionamentos do MR

PK/FK

Relacionamentos Espaciais

Geometria

SFS - *Simple Feature Specification For SQL*

+ SFS - *Simple Feature Specification*

- Especificação criada pelo OGC (*Open Geospatial Consortium*) que trata das seguintes questões:
 - Armazenamento da componente espacial ***vetorial de dados*** geográficos em bancos de dados
 - Fornecimento de interface de acesso a esses dados: SQL, CORBA e COM

+ Esquema Para Tabela de Dados Geográficos

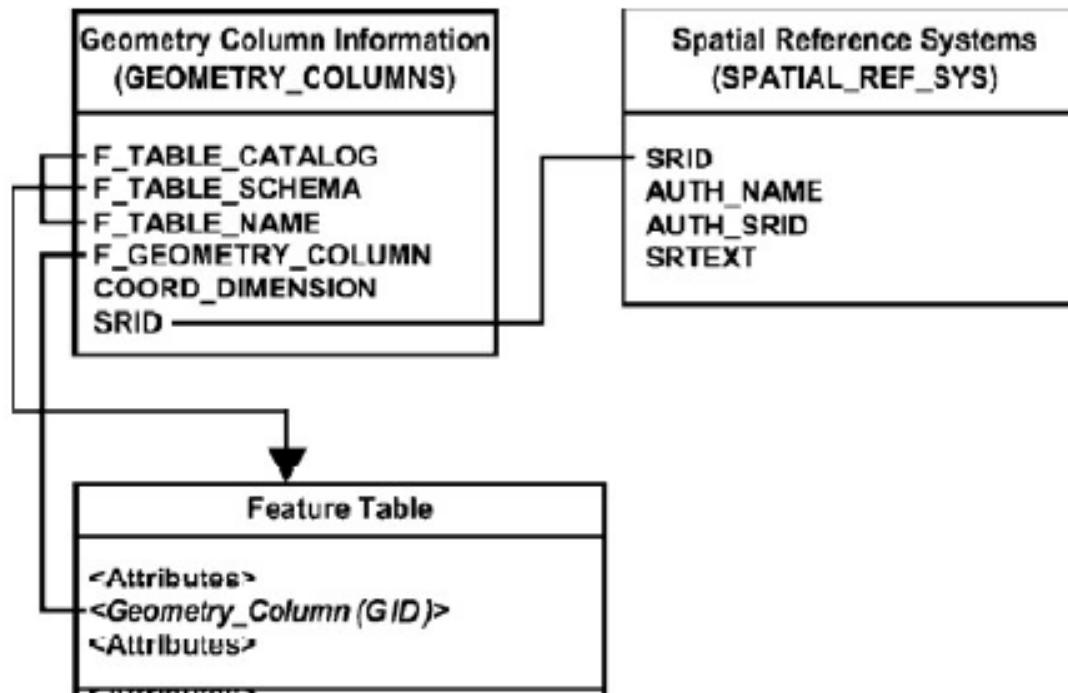




Tabela de SRS

```
CREATE TABLE SPATIAL_REF_SYS
(
    SRID          INTEGER NOT NULL PRIMARY KEY,
    AUTH_NAME     VARCHAR (256),
    AUTH_SRID     INTEGER,
    SRTEXT        VARCHAR (2048)
)
```



Tabela GEOMETRY_COLUMNS

```
CREATE TABLE GEOMETRY_COLUMNS (
    F_TABLE_CATALOG          VARCHAR (256)  NOT NULL,
    F_TABLE_SCHEMA             VARCHAR (256)  NOT NULL,
    F_TABLE_NAME               VARCHAR (256)  NOT NULL,
    F_GEOMETRY_COLUMN          VARCHAR (256)  NOT NULL,
    GEOMETRY_TYPE              INTEGER,
    COORD_DIMENSION            INTEGER,
    MAX_PPR                   INTEGER,
    SRID                      INTEGER REFERENCES SPATIAL_REF_SYS,
    CONSTRAINT GC_PK PRIMARY KEY
        (F_TABLE_CATALOG, F_TABLE_SCHEMA, F_TABLE_NAME, F_GEOMETRY_COLUMN)
)
```

+

Tabela de Features

```
CREATE TABLE <feature-name>  (
    <FID name>          <FID type>,
    <feature attributes> <other FID type> REFERENCES <other feature view>,
    ... (other FID based attributes for feature relations)
    ... (other attributes for feature)
    <geometry attribute 1> <GID type>,
    ... (other geometric attributes for feature)
    PRIMARY KEY <FID name>,
    ... (other geometric attributes foreign key statements)
    FOREIGN KEY <geometric attribute 1> REFERENCES <geometry-table-name-1>,
    FOREIGN KEY <FID relation name> REFERENCES <FEATURE table> <other FID name>,
    ... (other geometric attributes foreign key statements)
)
```



Criando Tabela Feature

```
CREATE TABLE uf
( codigo_uf      serial NOT NULL,
  nome            character varying(50) NOT NULL,
  sigla           character(2) NOT NULL,
  the_geom        geometry NOT NULL,
  CONSTRAINT pk_uf PRIMARY KEY (codigo_uf)
)
```

```
CREATE TABLE PARADA_ONIBUS
( id      INTEGER PRIMARY KEY,
  nome    VARCHAR(30),
  Geometria GEOMETRY)
```



Tipo de Dados Geométricos

Geometry Type	SQL Text Literal Representation	Comment
Point	'POINT (10 10)'	a Point
LineString	'LINESTRING (10 10, 20 20, 30 40)'	a LineString with 3 points
Polygon	'POLYGON ((10 10, 10 20, 20 20, 20 15, 10 10))'	a Polygon with 1 exterior ring and 0 interior rings
Multipoint	'MULTIPOINT (10 10, 20 20)'	a MultiPoint with 2 point
MultiLineString	'MULTILINESTRING ((10 10, 20 20), (15 15, 30 15))'	a MultiLineString with 2 linestrings
MultiPolygon	'MULTIPOLYGON (((10 10, 10 20, 20 20, 20 15, 10 10)), ((60 60, 70 70, 80 60, 60 60)))'	a MultiPolygon with 2 polygons
GeomCollection	'GEOMETRYCOLLECTION (POINT (10 10), POINT (30 30), LINESTRING (15 15, 20 20))'	a GeometryCollection consisting of 2 Point values and a LineString value

+ Tabela de Features

CREATE TABLE

```
AddGeometryColumn(  
    <schema_name>,  
    <table_name>,  
    <column_name>,  
    <srid>,  
    <type>,  
    <dimension>  
)
```

```
AddGeometryColumn(  
    <table_name>,  
    <column_name>,  
    <srid>,  
    <type>,  
    <dimension>  
)
```

+ Procedure

AddGeometryColumn

```
AddGeometryColumn (  
    <schema_name>,  
    <table_name>,  
    <column_name>,  
    <srid>,  
    <type>,  
    <dimension>  
)
```

- 1 – Garantir que o SRID existe na tabela SRS
- 2 – Adicionar a coluna na
GEOMETRY_COLUMNS
- 3 – Adicionar a coluna de geometria na tabela
- 4 – Adicionar a *constraint* de *check* para
o sistema de referencia na tabela de feature

+ Procedure

DropGeometryColumn

```
DropGeometryColumn(  
FEATURE_TABLE_CATALOG,  
FEATURE_TABLE_SCHEMA,  
FEATURE_TABLE_NAME,  
GEOMETRY_COLUMN_NAME)
```

1 – Apaga a entrada na tabela de
GEOMETRY_COLUMNS

2 – Apaga a coluna de geometria na
tabela

3 – Apagar a constraint de check para
o sistema de referencia na tabela de
feature



Exemplo: Tabela Rodovia

```
CREATE TABLE rodovia
  (id serial primary key,
  nome varchar(30))
```

```
SELECT AddGeometryColumn( 'rodovia', 'geometria', 4291, 'LINESTRING', 2 )
```

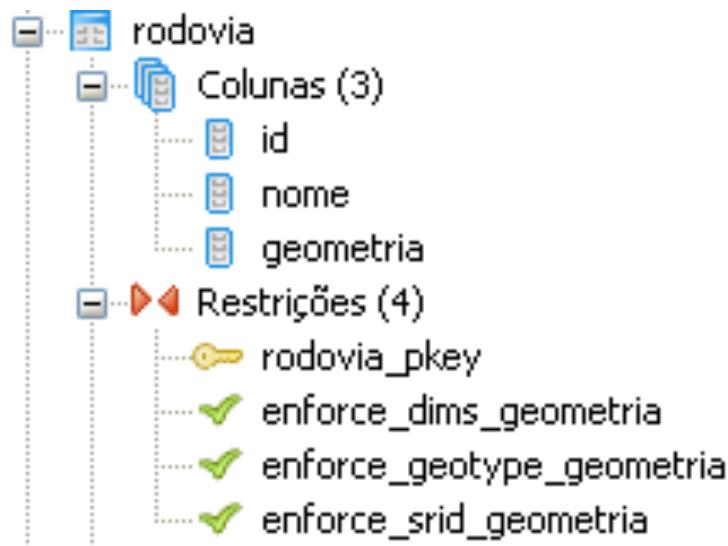




Tabela geometry_columns

Query - BancoDadosGeograficos em postgres@localhost:5432 *

Arquivo Editar Consulta Favoritos Macros Visualizar Ajuda

SQL Editor Graphical Query Builder

```
SELECT * FROM geometry_columns
```

Painel de saída

Saída de Dados Explain Mensagens Histórico

	f_table_catalog character var	f_table_schema character var	f_table_name character var	f_geometry_column character var	coord_dimension integer	srid integer	type character var
1		public	rodovia	geometria	2	4291	LINESTRING



WKB e WKT

- Duas maneiras de expressar dados espaciais:
 - WKT (well-know text)
 - WKB (well-know Binary)
- WKT
 - POINT
 - LINE
 - POLYGON
 - ...



Entrada/Saída

```
bytea WKB = ST_AsBinary(geometry);
text WKT = ST_AsText(geometry);
geometry = ST_GeomFromWKB(bytea WKB, SRID);
geometry = ST_GeometryFromText(text WKT, SRID);
```

```
INSERT INTO geotable ( the_geom, the_name )
VALUES ( ST_GeomFromText('POINT(-126.4 45.32)', 312), 'A Place');
```



INSERT

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



Inserindo Dados na tabela

Query - BancoDadosGeograficos em postgres@localhost:5432 - [D:\BDG\rodovia.sql]

Arquivo Editar Consulta Favoritos Macros Visualizar Ajuda

SQL Editor Graphical Query Builder

```
BEGIN;
INSERT INTO rodovia (id, geometria, nome)
  VALUES (1,ST_GeomFromText('LINESTRING(191232 243118,191108 243242)',-1),'Avenida Paulista');
INSERT INTO rodovia (id, geometria, nome)
  VALUES (2,ST_GeomFromText('LINESTRING(189141 244158,189265 244817)',-1),'Avenida Sao Joao');
INSERT INTO rodovia (id, geometria, nome)
  VALUES (3,ST_GeomFromText('LINESTRING(192783 228138,192612 229814)',-1),'Rua Brig. Pessoa Ramos');
INSERT INTO rodovia (id, geometria, nome)
  VALUES (4,ST_GeomFromText('LINESTRING(189412 252431,189631 259122)',-1),'Rua das Borboletas');
INSERT INTO rodovia (id, geometria, nome)
  VALUES (5,ST_GeomFromText('LINESTRING(190131 224148,190871 228134)',-1),'Rua Hermes');
INSERT INTO rodovia (id, geometria, nome)
  VALUES (6,ST_GeomFromText('LINESTRING(198231 263418,198213 268322)',-1),'Avenida Salgado Filho');
COMMIT;
```

Painel de saída

Saída de Dados Explain Mensagens Histórico

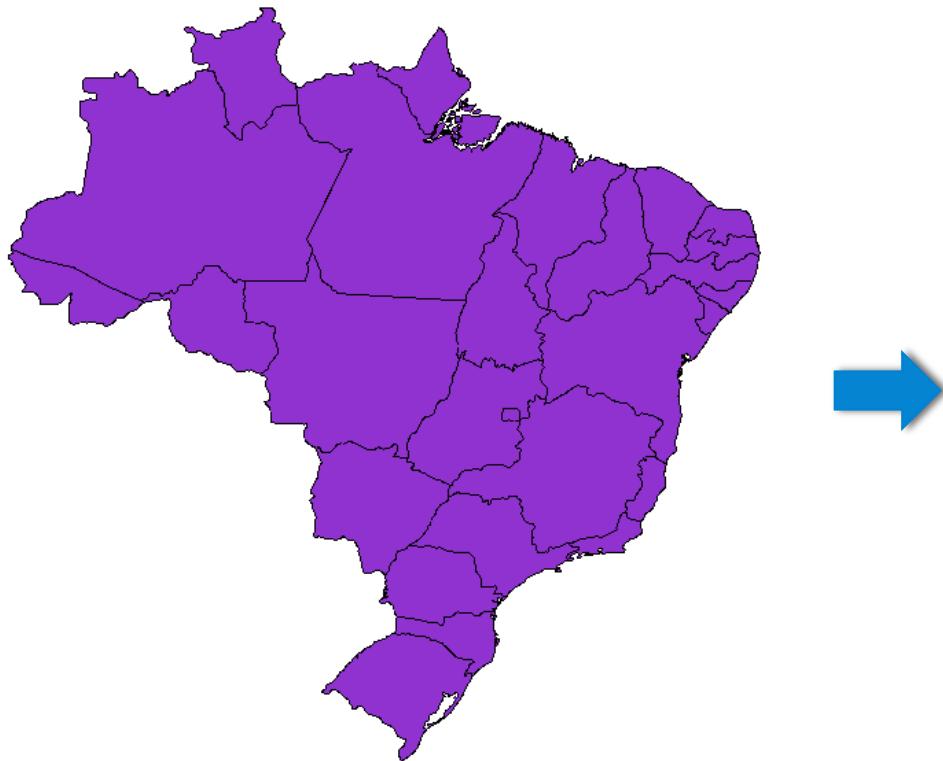
ERRO: novo registro da relação "rodovia" viola restrição de verificação "enforce_srid_geometria"

***** Erro *****

ERRO: novo registro da relação "rodovia" viola restrição de verificação "enforce_srid_geometria"

SQL state: 23514

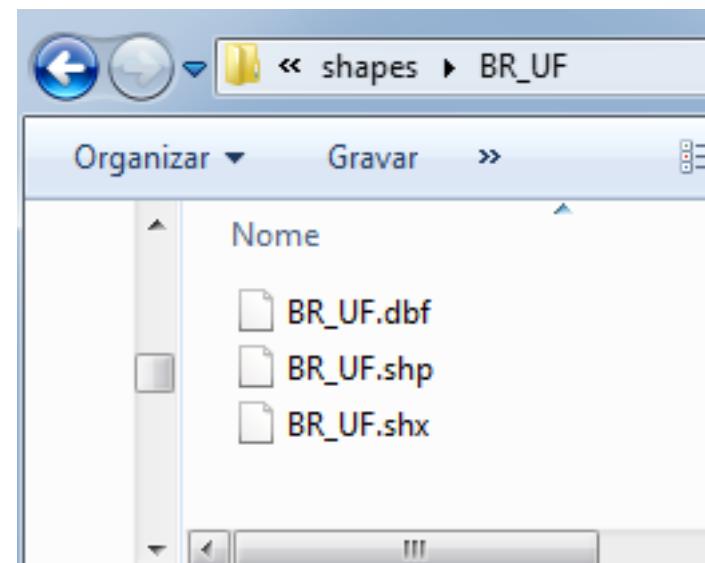
+ Populando uma tabela de Feature com um arquivo shape





Arquivo Shapefile

- O shapefile arquivo que representa uma feição (ponto, linha ou polígono).
- Os arquivos shapefile foram originados, desenvolvidos e regulamentados pela ESRI
- Arquivos de um shape:
 - “.shp”: armazena a geometria (obrigatório)
 - “.shx”: arquivo de índices da geometria (obrigatório)
 - “.dbf”: informações de atributos das feições (obrigatório)





Passos

1. Transforma o arquivo shape em um script.sql
2. Executa o script no pgAdmin



shp2pgsql

- shp2pgsql: transforma shp to sql
 - C:\BDG\shp2pgsql -i BR_UF.shp TB_BR_UF > estados.sql
 - C:\BDG\shp2pgsql -i -s 4326 BR_UF.shp TB_BR_UF > estados.sql

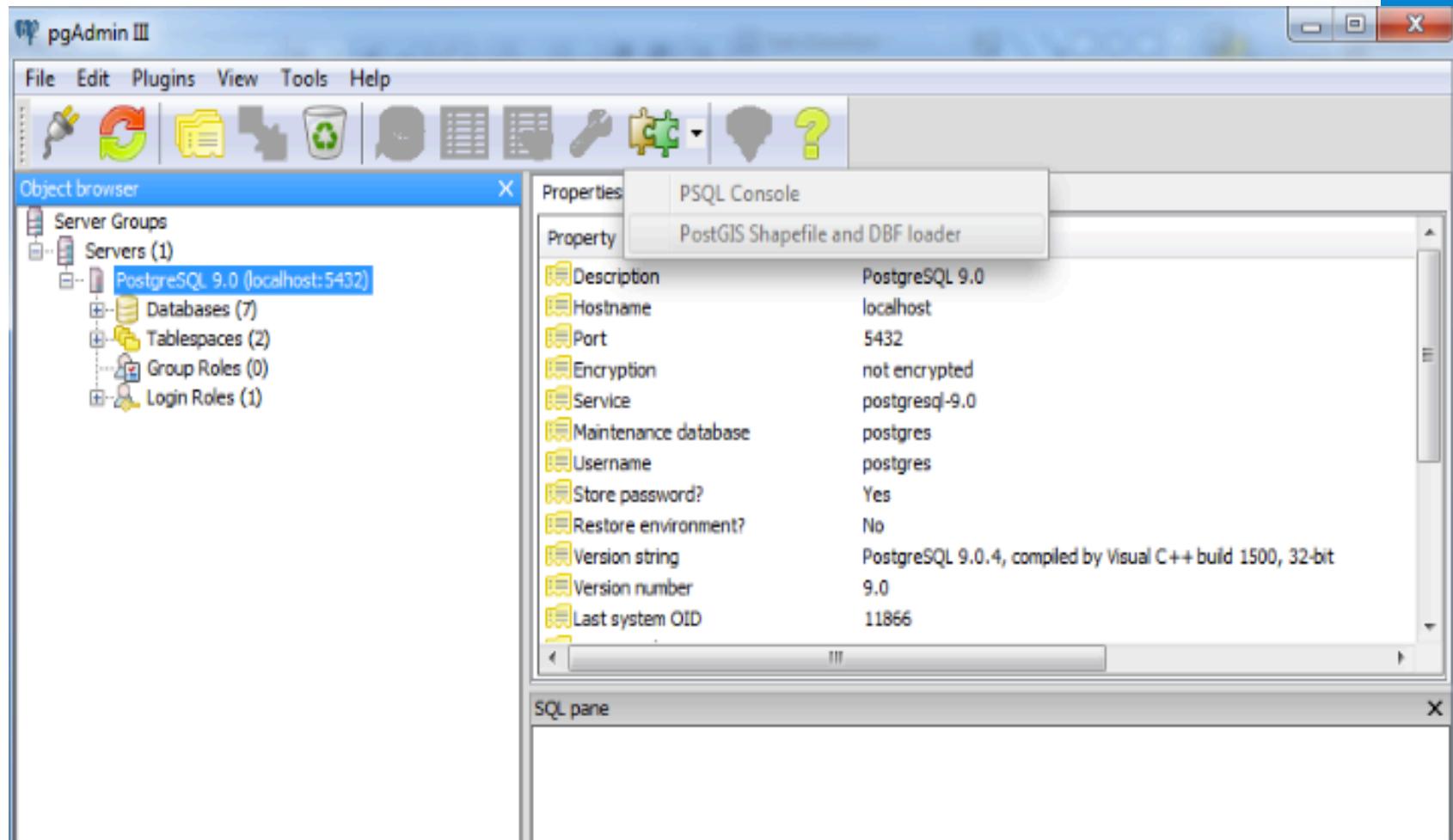


Script de um arquivo shape

```
SET CLIENT_ENCODING TO UTF8;
SET STANDARD_CONFORMING_STRINGS TO ON;

BEGIN;
CREATE TABLE "radares" (gid serial PRIMARY KEY,
"id" int4,
"codigo_rad" int4);
SELECT AddGeometryColumn",'radares','the_geom','4326','POINT',2);
INSERT INTO "radares" ("id","codigo_rad",the_geom) VALUES
('0','1','0101000020E610000058BFB3C148F247C07E60C2D6C5932FC0');
INSERT INTO "radares" ("id","codigo_rad",the_geom) VALUES
('0','2','0101000020E61000000374464694F147C015C2386BCF942FC0');
INSERT INTO "radares" ("id","codigo_rad",the_geom) VALUES
('0','3','0101000020E610000057487034FAF147C063255909CA952FC0');
INSERT INTO "radares" ("id","codigo_rad",the_geom) VALUES
('0','4','0101000020E6100000E1F79D005BF047C01D0FB78B8C962FC0');
INSERT INTO "radares" ("id","codigo_rad",the_geom) VALUES
('0','5','0101000020E61000002DEA4DB3D2F047C0D781D18000972FC0');
INSERT INTO "radares" ("id","codigo_rad",the_geom) VALUES
('0','6','0101000020E61000004107CD291BF347C0C1820A989A922FC0');
```

+ Populando o banco com o PgAdmIII





Sites com Mapas

www.gismaps.com.br/mundo/mundo.htm

ScienceDirect - Comp. Apple gis_agente apartamento Other Bookmark

GISMAPS 10 anos

Mapas Digitais

Canal Gismaps

home / ajuda / e-mail

Mundo

Mapa do Mundo

Mapa do mundo apresentando os países e respectivas províncias/estados, capitais e principais cidades, meridianos e paralelos, principais rios e lagos; dados gerais sobre população, moeda e área; referência 2002.

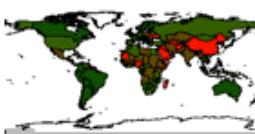
Fonte: ESRI
Versão: 2005.1
Tamanho: 5.201 MB
Formato: Shape



Índice de Sustentabilidade Ambiental - ESI

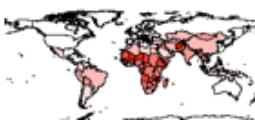
Mapas temáticos apresentando o Índice de Sustentabilidade Ambiental (ESI - 2002 Environmental Sustainability Index) de 142 países (26 índices).

Fonte: World Economic Forum
Versão: 2003.2
Tamanho: 1.610 Kb
Formato: Shape



Perspectivas da População Mundial

Mapas temáticos apresentando o relatório Perspectivas da População Mundial - WPP - World Population Prospects; estimativas (1950-2000) e projeções (2000-2050) de indicadores demográficos de várias



Escolha outros temas

► **Gismaps**
Neste site são disponibilizados para download vários arquivos Shape, organizados em 24 temas diferentes, para serem usados em softwares de geoprocessamento.

► **CD da Gismaps**
Solicite o CD da Gismaps e receba todos os arquivos Shape do site. Clique [aqui](#).

► **Gismaps Viewer**
Para gerar os mapas temáticos é necessário instalar em seu computador o software Gismaps Viewer. Faça o download informando:

Nome:
Empresa:
e-mail:

► **Tutoriais**
A melhor maneira de conhecer todos os recursos do software Gismaps Viewer, e aprender a fazer os melhores mapas temáticos, é seguir os [tutoriais on-line](#).

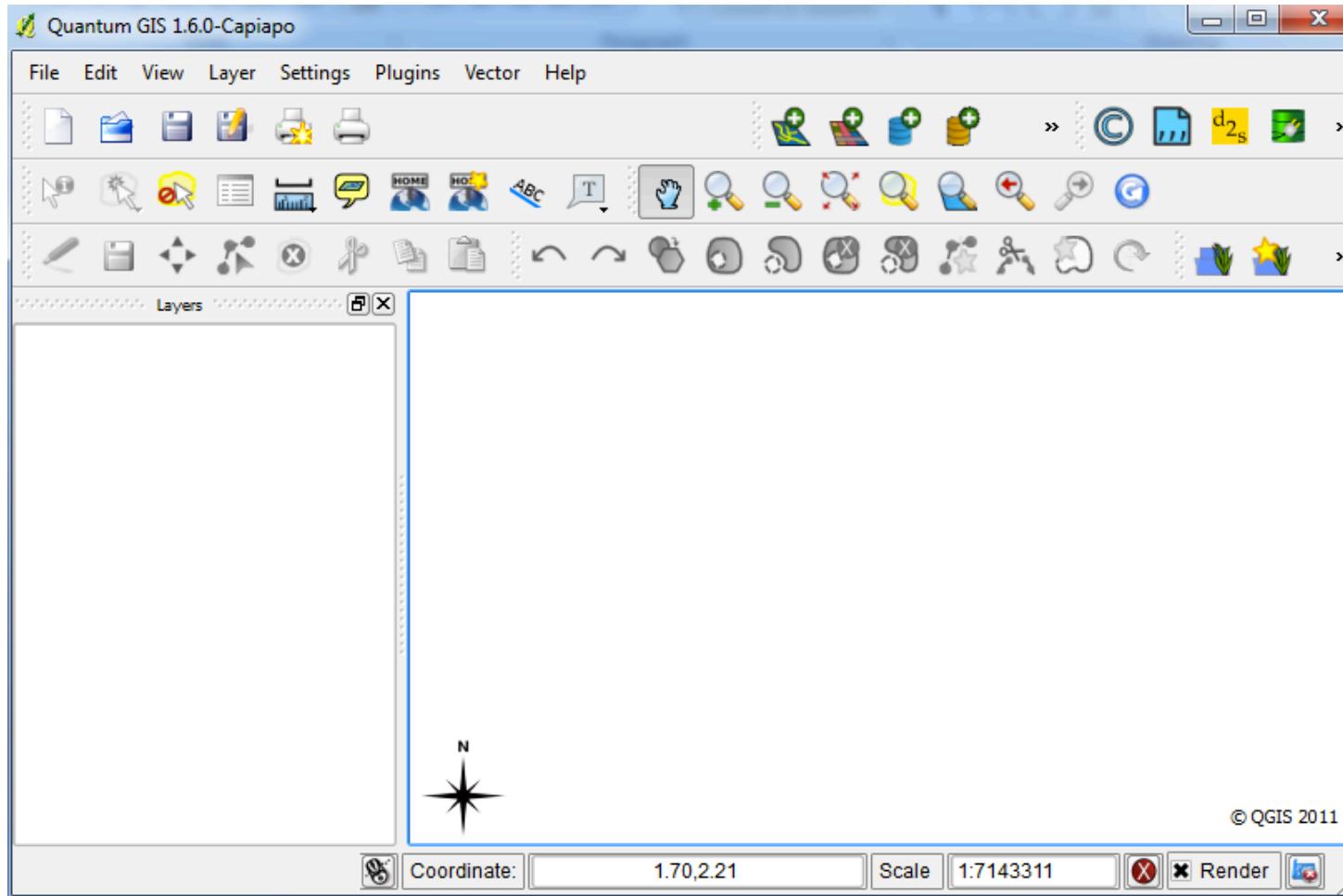
► **Marketing / Comercial**
Verifique como utilizar os mapas da Gismaps para atividades de [Marketing e Comercial](#).

+

Visualizando os mapas

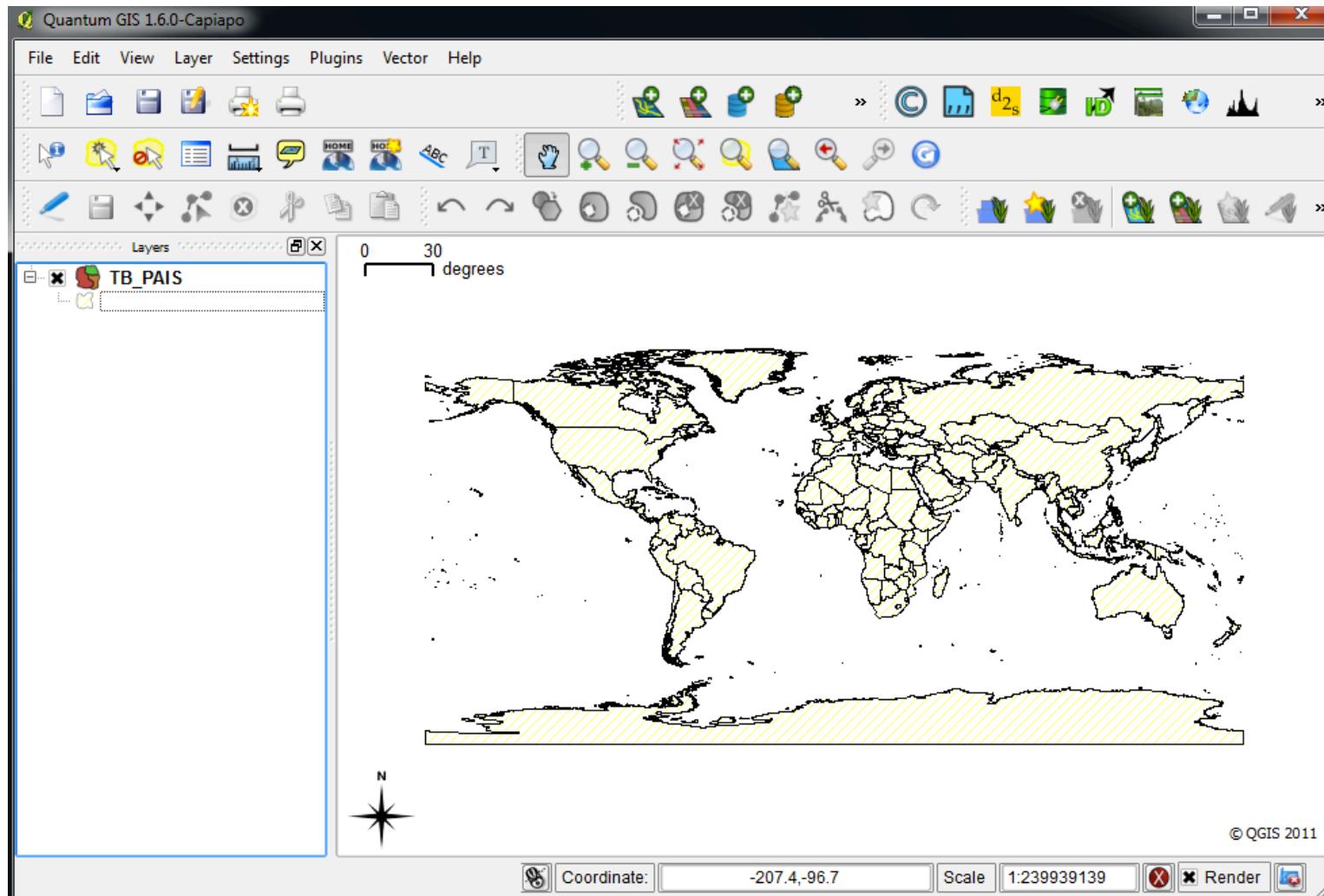
- Ferramentas
 - uDIG
 - Quantum GIS (QGis)
 - gvSIG

+ Ferramenta de Vizualização de Dados Geográficos





QGis





QGis

Quantum GIS 1.6.0-Capiapo

File Edit View Layer Settings Plugins Vector Help

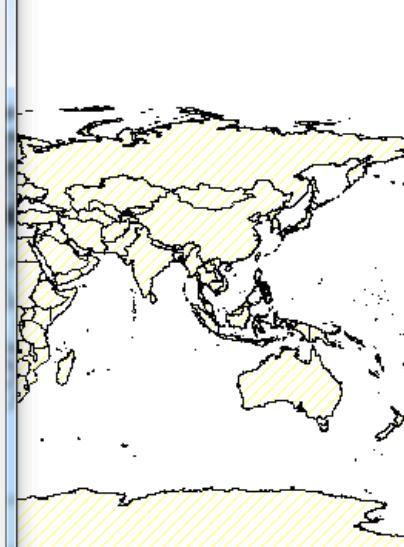
Attribute table - TB_PAIS (1 matching features)

	iso_2digit	iso_3digit	cntry_name	long_name	sovereign	po
24	IT	ITA	Italy	Italy	Italy	
25	SI	SVN	Slovenia	Slovenia	Slovenia	
26	MX	MEX	Mexico	Mexico	Mexico	
27	HU	HUN	Hungary	Hungary	Hungary	
28	FR	FRA	France	France	France	
29	RO	ROU	Romania	Romania	Romania	
30	SM	SMR	San Marino	San Marino	San Marino	
31	MC	MCO	Monaco	Monaco	Monaco	
32	ES	ESP	Spain	Spain	Spain	
33	GE	GEO	Georgia	Georgia	Georgia	
34	BG	BGR	Bulgaria	Bulgaria	Bulgaria	
35	AM	ARM	Armenia	Armenia	Armenia	
36	UM	NULL	Jarvis I.	Jarvis Island	United States	
37	KG	KGZ	Kyrgyzstan	Kyrgyzstan	Kyrgyzstan	
38	AD	AND	Andorra	Andorra	Andorra	
39	MK	MKD	Macedonia	Macedonia	Macedonia	
40	TV	TUV	Tuvalu	Tuvalu	Tuvalu	

Look for razil in cntry_nar Search

Show selected only Search selected only Case sensitive Advanced search ?

Coordinate: 112.5,-31.8 Scale 1:239939139 Render



© QGIS 2011



Referência

- Documentação ESRI : <http://resources.arcgis.com/>
- Especificações OGC (<http://www.opengeospatial.org/>)
- Livro de Banco de Dados Geográficos do INPE
- Manual do PostGIS: <http://postgis.refractions.net/>