



Taller introducción base de datos geográfica

Postgis 2

1 – Introducción - Definiciones

1.1) PostgreSQL

Es un Sistema de Gestión de Bases de Datos (SGBD) relacional orientado a objetos y libre.

SGBD, es un conjunto de programas que permiten el almacenamiento, modificación y extracción de información en una base de datos. Poseen métodos para mantener la integridad de los datos, administrar el acceso de usuarios a los datos y para recuperar la información en caso de fallo del sistema.

1.2) Postgis

Es un módulo que añade soporte de objetos geográficos a PostgreSQL, convirtiéndolo en una **base de datos espacial** (permite definir la localización y relación espacial entre los objetos).

Los objetos deben estar referenciados en un determinado Sistema de Referencia Espacial (SRS) para definir las relaciones entre ellos.

Es posible utilizar datos y consultas espaciales compartiendo o utilizando información almacenada de forma tradicional en la base de datos.

2 – Shapefile vs Postgis

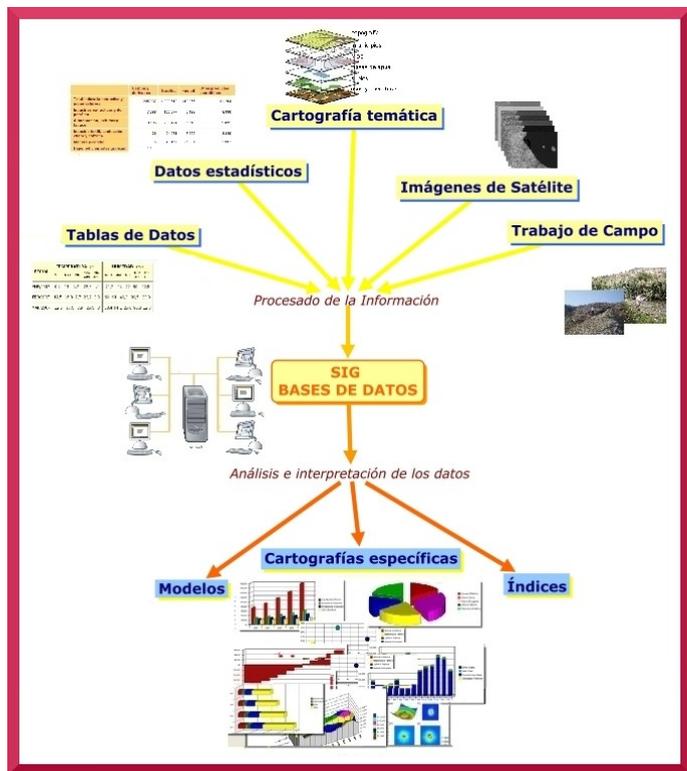


Shapefile, formato de archivo propietario desarrollado por la compañía Esri.

Archivos locales almacenadas en el pc de usuario o red.

No permite edición de múltiples usuarios simultáneamente.

Difícil mantenimiento e integridad de datos.



Base de Datos PostgreSQL + Postgis, software libre.
Almacenamiento centralizado de información, geográfica o de otro tipo.

Relaciones entre objetos y datos.

Permite consulta y edición simultáneamente a múltiples usuarios.

Permite establecer distintos permisos para diferentes usuarios o grupos.

Respaldos programables en forma automática.

Recuperación de datos (último, por fecha, totales, etc.).

3 – Instalaciones

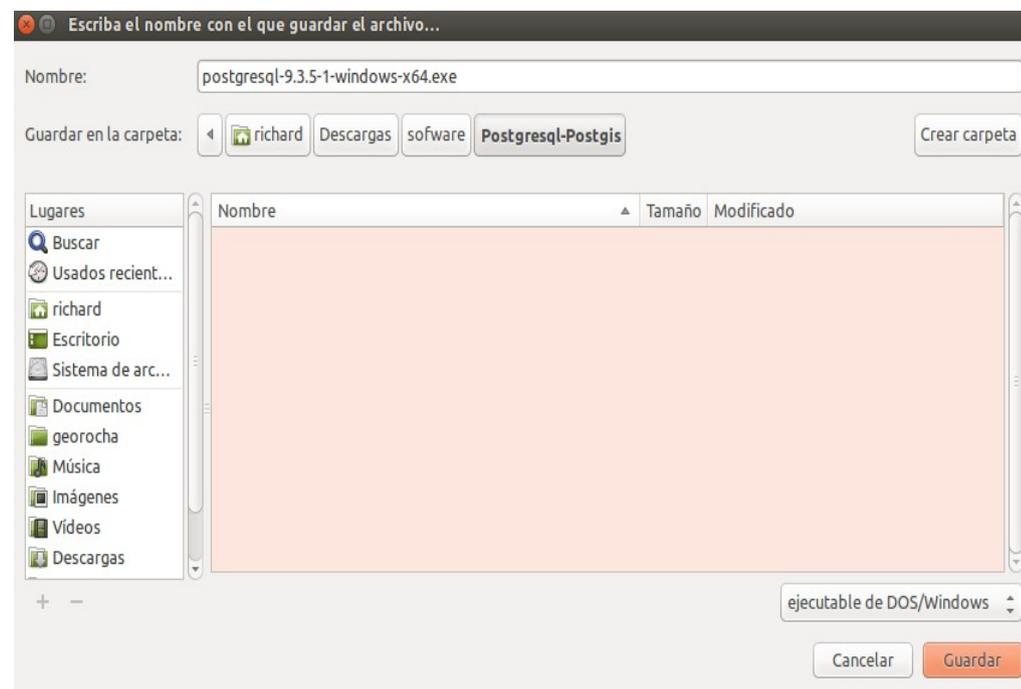
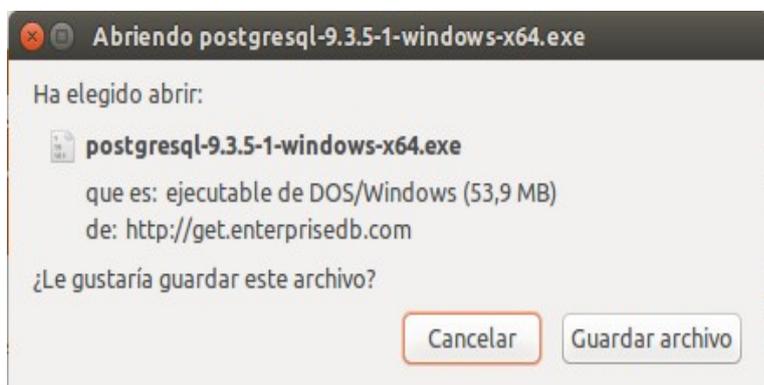


PostgreSQL

3.1) PostgreSQL

- Información y documentación: <http://www.postgresql.org.es/>
- Descargar archivo ejecutable desde:

<http://www.enterprisedb.com/products-services-training/pgdownload#windows>
(elegir Sistema Operativo y versión para descargar)



Elegimos el directorio donde guardar el archivo de instalación, posteriormente procedemos a la instalación del mismo.

3 – Instalaciones



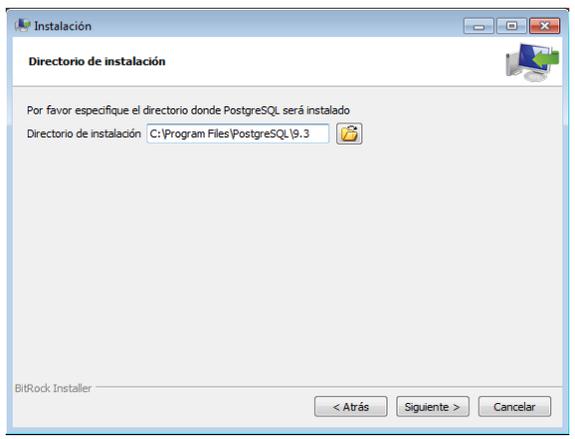
PostgreSQL

3.2) Instalar PostgreSQL en Windows

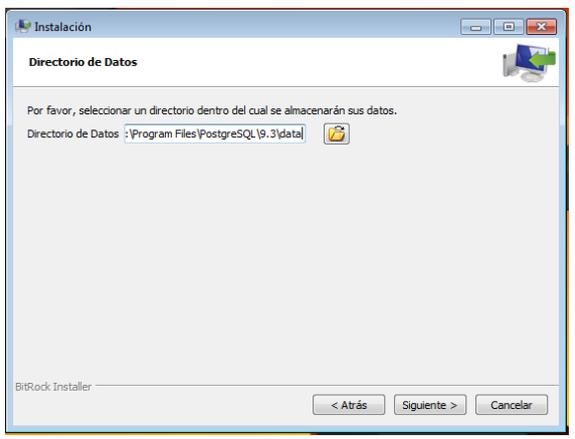
- Ejecutar archivo “*postgresql-9.5.5-1-windows-x64.exe*” (nombre archivo puede variar según sistema operativo o versión descargada, debemos tener permisos de administrador para poder instalar programas):



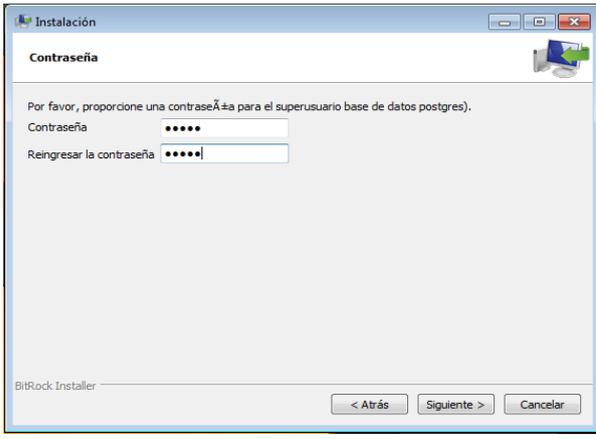
1 - Iniciar instalación



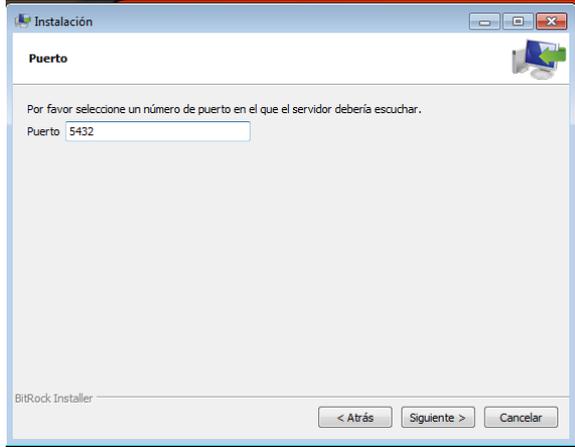
2 - Elegir directorio de instalación



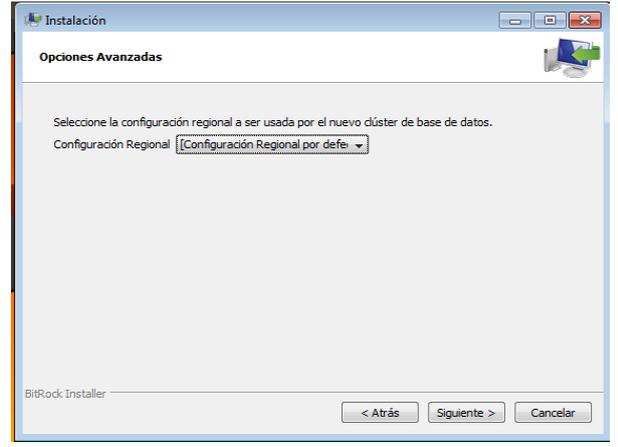
3 - Elegir directorio de datos



4 – Elegir contraseña para usuario postgres



5 – Elegir puerto instalación, por defecto 5432



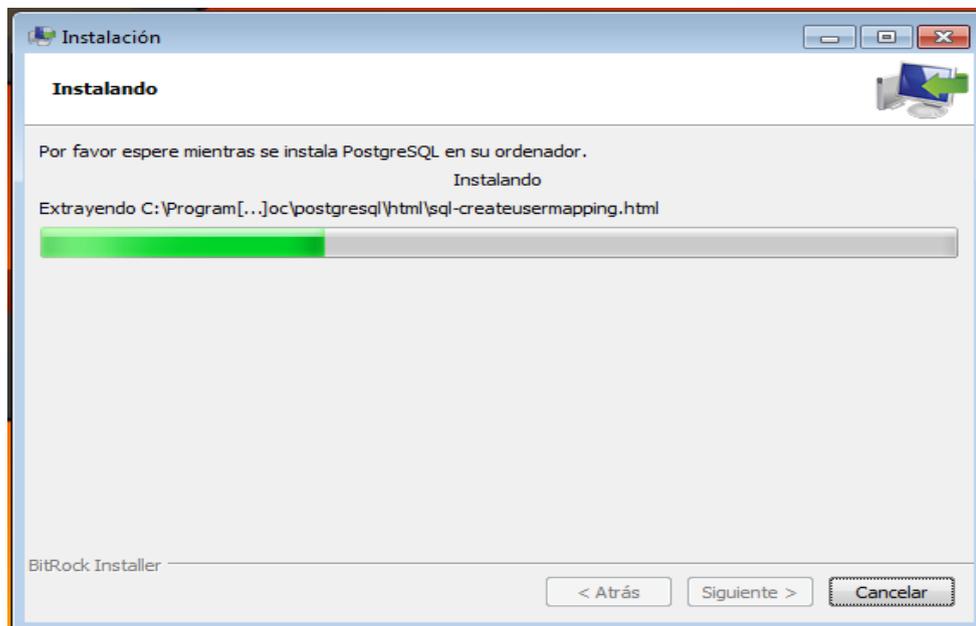
6 – Elegir configuración de caracteres (dejar default sistema)

3 – Instalaciones



3.2) Instalar PostgreSQL en Windows

PostgreSQL



7 – Luego de presionar “Siguiete” comienza el proceso de instalación

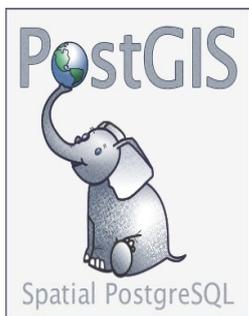


8 – Opcionalmente podemos instalar otras herramientas, podemos desmarcar esta opción y presionar “Terminar” para culminar la instalación de PostgreSQL.

NOTA: por defecto la instalación en Windows instala la herramienta de administración PgAdmin 3, en Linux puede ser necesario instalar PgAdmin 3 separado.

3 – Instalaciones

3.3) Postgis



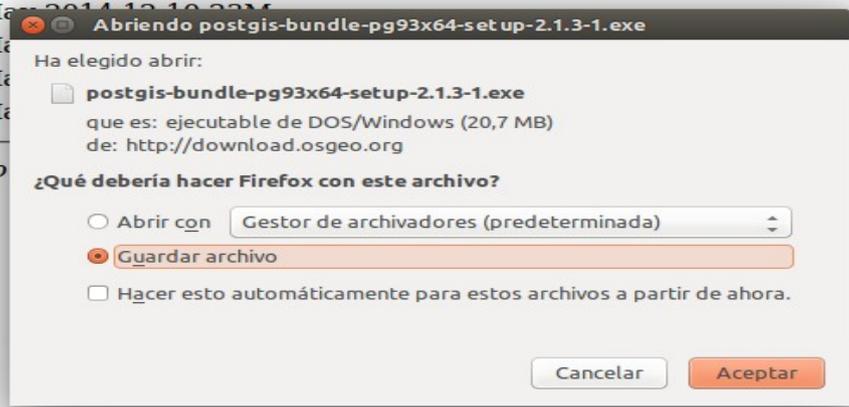
- Documentación: <http://postgis.net/documentation>
- Descargar archivo ejecutable desde:
<http://download.osgeo.org/postgis/windows/pg93/>

(elegir versión para descargar, ej: Windows 64 bits archivo *postgis-bundle-pg93x64-setup-2.1.3-1.exe*)

Index of /postgis/windows/pg93

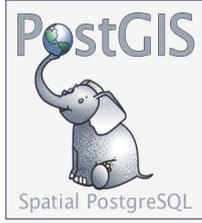
<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
archive/	18-May-2014 12:07	-	
buildbot/	15-Jan-2014 22:06	-	
extras/	15-Jan-2014 22:05	-	
postgis-bundle-pg93x32-2.1.3.zip	18-May-2014 12:10:22		
postgis-bundle-pg93x32-setup-2.1.3-1.exe	18-May-2014 12:10:22		
postgis-bundle-pg93x64-2.1.3.zip	18-May-2014 12:10:22		
postgis-bundle-pg93x64-setup-2.1.3-1.exe	18-May-2014 12:10:22		

Apache/2.2.16 (Debian) Server at download.osgeo.org



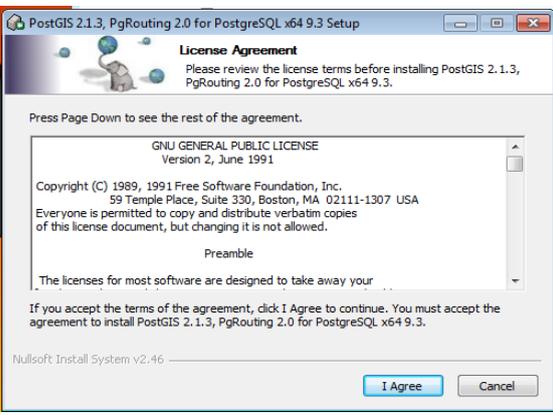
Elegimos el directorio donde guardar el archivo de instalación, posteriormente procedemos a la instalación del mismo.

3 – Instalaciones

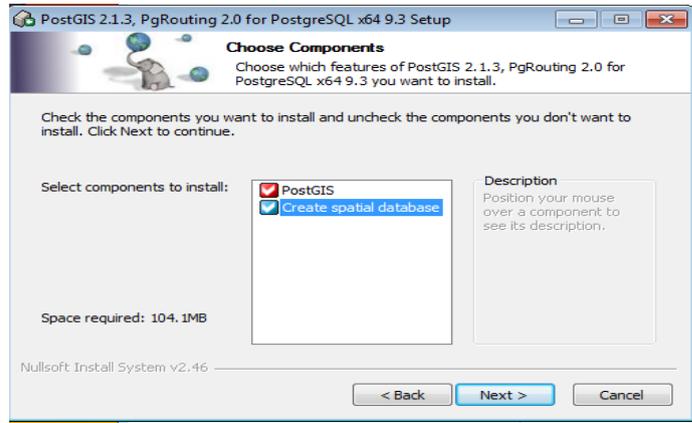


3.3) Postgis

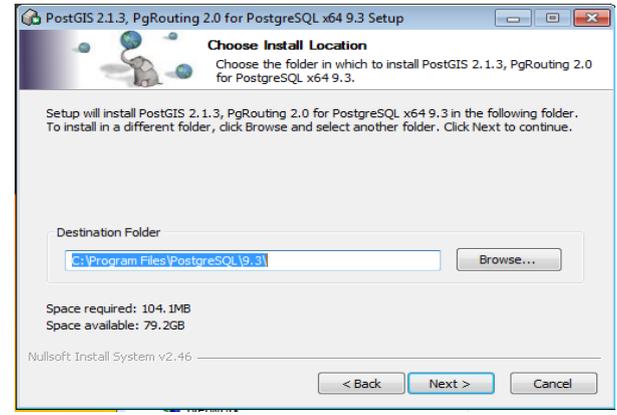
- Ejecutar archivo “*postgis-bundle-pg93x64-setup-2.1.3-1.exe*” (nombre archivo puede variar según Sistema Operativo o versión).



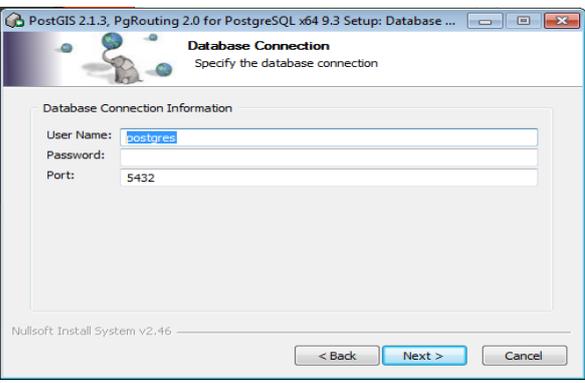
1 – Aceptar contrato de licencia



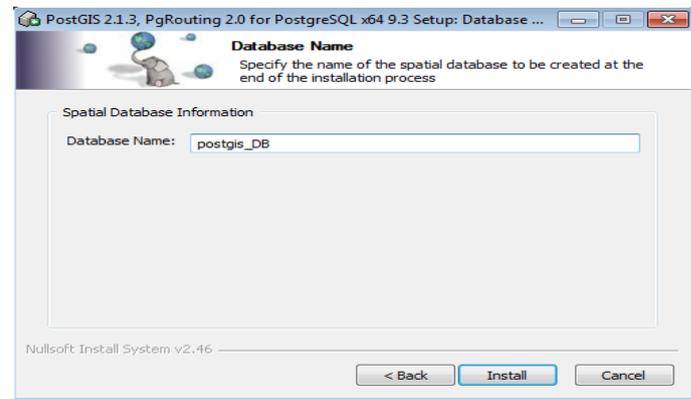
2 – Marcar la opción “*Create spatial database*”, presionar “Next”



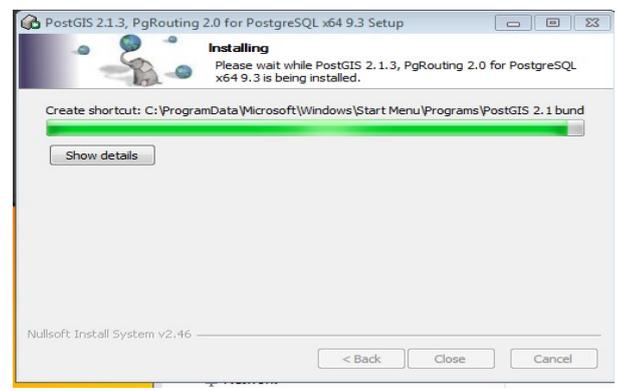
3 – Elegir el directorio para la instalación



4 – Ingresar la contraseña administrador postgres ingresada anteriormente

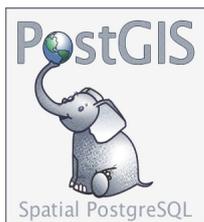


5 – Ingresar nombre nueva “**Base de Datos**”, presionar “Install”.

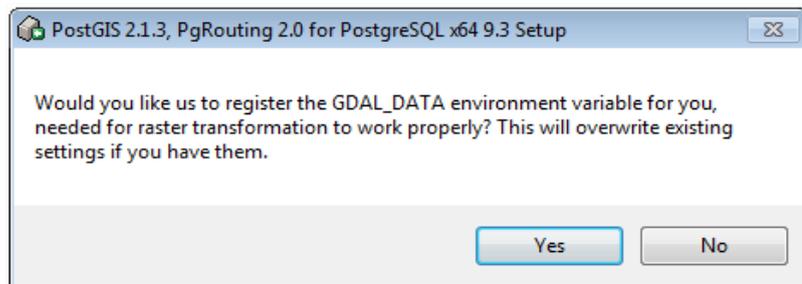


6 – Progreso de instalación

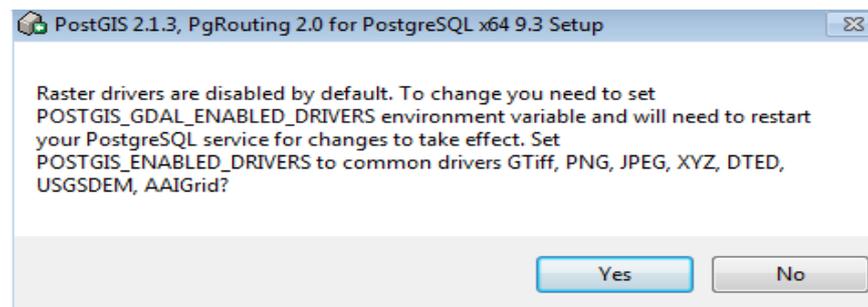
3 – Instalaciones



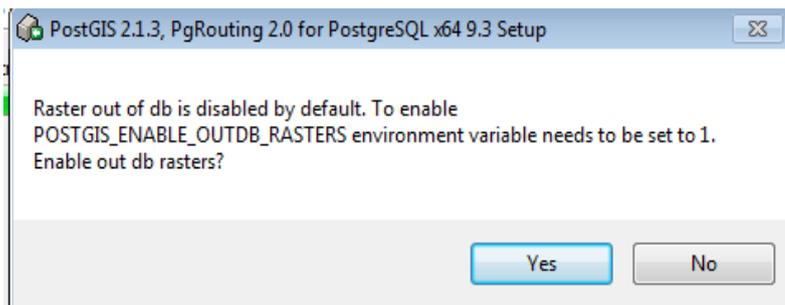
3.3) Postgis



7 – Presionar “Yes” para aceptar la sugerencia de registrar la variable GDAL_DATA

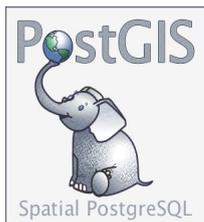


8 – Presionar “Yes” para instalar el soporte de datos raster.



9 – Presionar “Yes” para instalar el soporte de datos raster.

4 – Crear una base de datos geográfica



4.1) conectarse a postgresql Utilizando pgAdmin

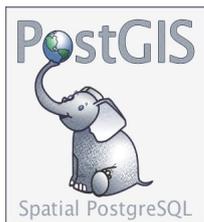
File -> Add Server

Por línea de comandos

```
# psql -U postgres -d template1
```

```
richard@debianGIS: ~ 78x8
template1=# \q
postgres@debianGIS:/home/richard$ psql -U postgres -d template1
psql (9.3.4)
Type "help" for help.
template1=#
```

4 – Crear una base de datos geográfica



4.2) crear una nueva base para almacenar los datos con geometrias: Utilizando pgAdmin

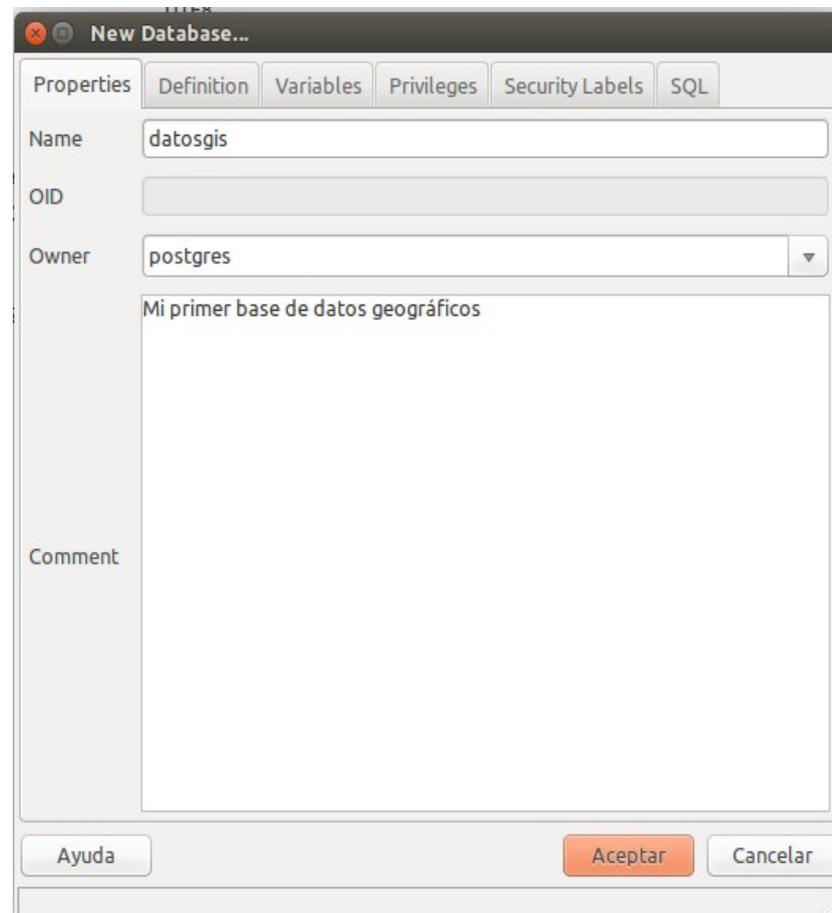
Edit -> Create

Por línea de comandos

```
# CREATE DATABASE datosgis WITH
ENCODING='UTF8' OWNER=postgres
CONNECTION LIMIT=-1 ;
```

```
richard@debianGIS: ~
postgres@debianGIS:/home/richard$ psql -U postgres -d template1
psql (9.3.4)
Type "help" for help.

template1=# CREATE DATABASE datosgis WITH ENCODING='UTF8' OWNER=postgres CONNECTION LIMIT=-1 ;
CREATE DATABASE
template1=#
```

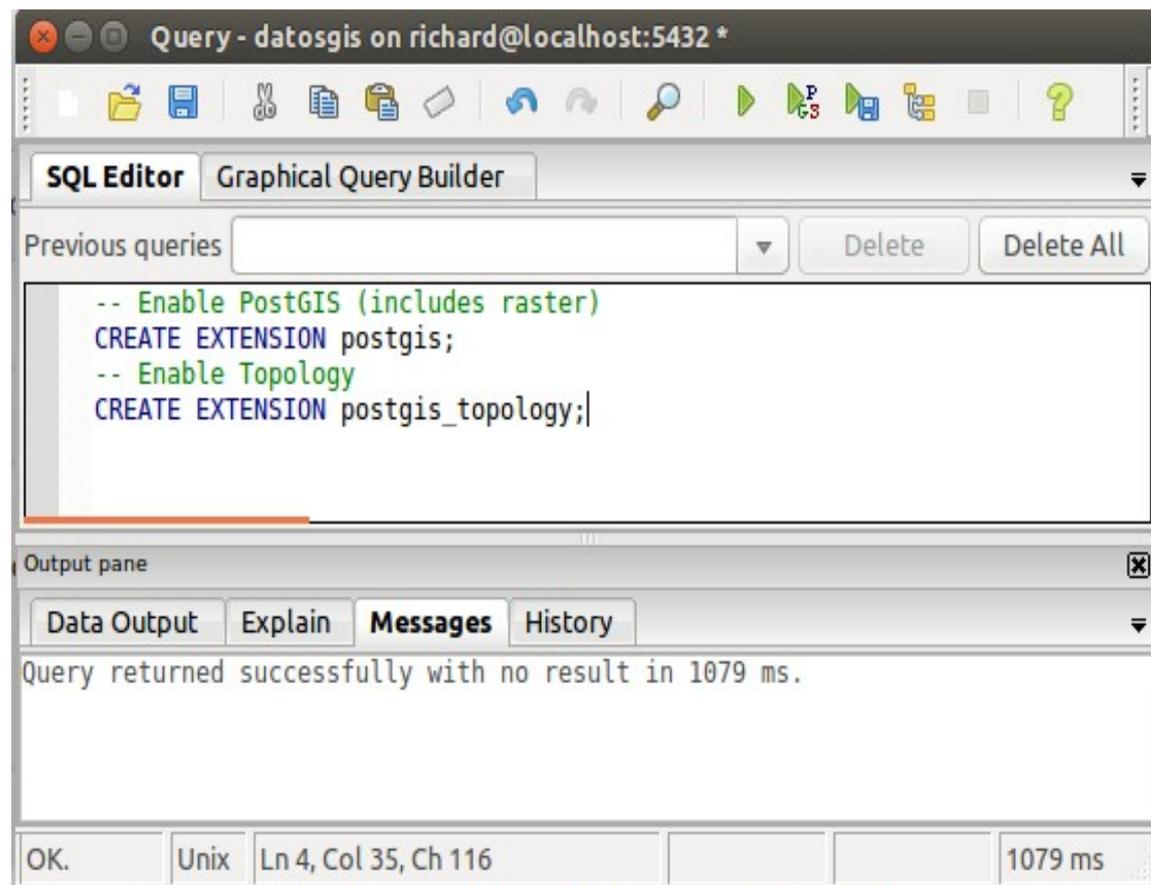


4 – Crear una base de datos geográfica



4.3) crear extensión para soporte Postgis, raster y topología:
Utilizando pgAdmin

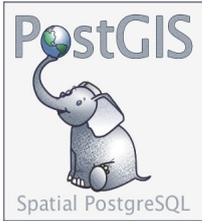
Tools -> Query Tool



Por línea de comandos

```
$ psql -U postgres -d datosgis ;
# CREATE EXTENSION postgis ;
# CREATE EXTENSION postgis_topology ;
```

5 – Gestión de datos



5.1) Tipo de datos espaciales: almacenan formas (shapefile) como puntos, líneas y polígonos en columnas de geometría.

5.2) Funciones espaciales: sirven para consultar las propiedades y relaciones espaciales.

5.3) Índices espaciales: se utilizan para el procesamiento eficiente de las operaciones espaciales.

5.4) Tipos de geometrías soportadas: POINT, LINESTRING, GEOMETRYCOLLECTION, MULTIPOINT, MULTILINESTRING, MULTIPOLYGON

Type		Type	
POINT		MULTIPOINT	
LINESTRING		MULTILINESTRING	
POLYGON		MULTIPOLYGON	
COLLECTION			

5 – Gestión de datos



5.5) Tabla *spatial_ref_sys*: contiene la definición de los distintos Sistemas de Referencia existentes, ej: UTM Zona 21S Datum Sirgas 2000 (EPSG:31981)

5.6) Tabla (view) *geometry_columns*: almacena para cada tabla el nombre, de cual es su columna geométrica, las dimensiones de la misma (2D, 3D, 4D), el código correspondiente al Sistema de Referencia (se relaciona con la tabla *spatial_ref_sys*) y el tipo de geometría.

	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)
1	sigrocha	public	cat_padrones_chuy	geom	2	31982	POLYGON
2	sigrocha	public	cat_padrones_lascano	geom	2	31981	POLYGON
3	sigrocha	public	cat_zonas_chuy	geom	2	31982	POLYGON
4	sigrocha	public	cat_zonas_lascano	geom	2	31981	POLYGON
5	sigrocha	public	pot_directrices_lasc	geom	2	31981	MULTIPOLYGON
6	sigrocha	public	v_cat_chuy_completo	geom	2	31982	POLYGON
7	sigrocha	public	v_cat_lascano_comple	geom	2	31981	POLYGON
8	sigrocha	public	cat_zonas_castillos	geom	2	31981	POLYGON
9	sigrocha	public	cat_padrones_castill	geom	2	31981	POLYGON
10	sigrocha	public	fr_manzanas_el_pedre	geom	2	31981	POLYGON
11	sigrocha	public	fr_manzanas_18_julio	geom	2	31981	POLYGON
12	sigrocha	public	cab_departamentos	geom	2	31981	POLYGON
13	sigrocha	public	cab_municipios	geom	2	31981	POLYGON

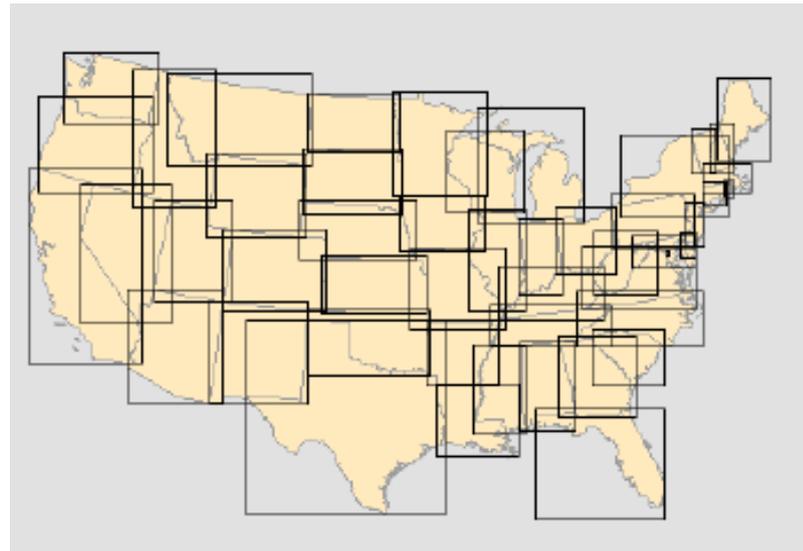
165 rows.

6 – Índice espaciales



6.1) Los índices espaciales son una de las funcionalidades más importantes de una base de datos espacial, estos consiguen que las búsquedas espaciales en un gran número de datos sean eficientes.

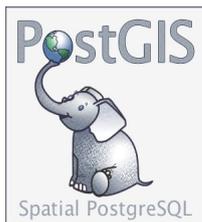
Los índices espaciales calculan y comparan básicamente el cuadro delimitador (bounding box) de cada elemento, este es el rectángulo de tamaño más pequeño capaz de contener un determinado elemento.



Crear un índice geométrico:

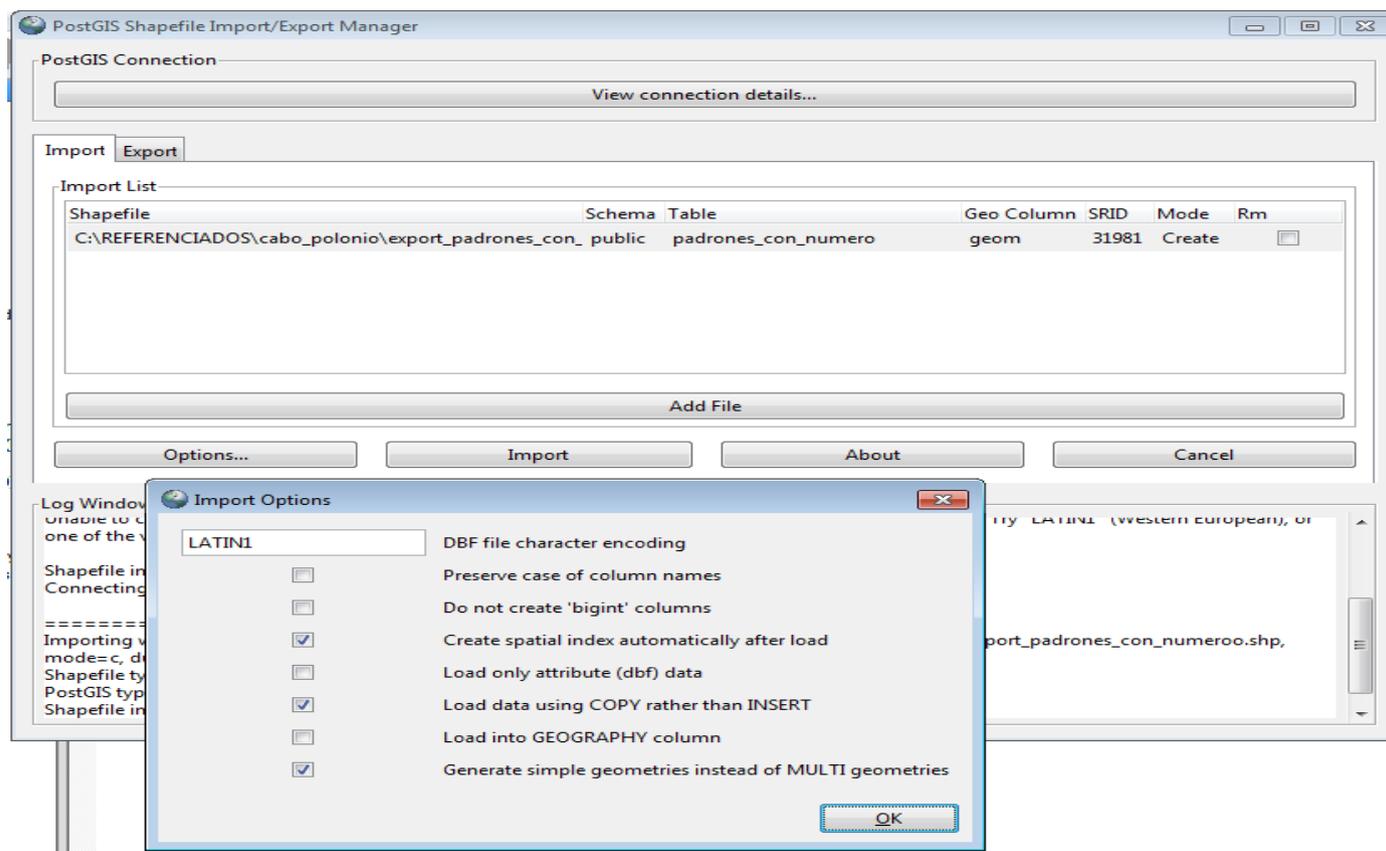
```
# CREATE INDEX index_gist_my_table ON my_table USING gist (geom) ;
```

7 – Análisis espacial



7.1) Importar archivo shapefile a la base de datos:

PgAdmin -> Plugins -> *Postgis Shapefile Import/Export Manager*



- Presionar “Add File” y elegir archivo shapefile
- Seleccionar “Options...” para establecer opciones avanzadas
- Presionar “Import” para proceder a la importación

7 – Análisis espacial



7.2) Importar archivos en formato shapefile a la base de datos: Por línea de comandos:

```
$ shp2pgsql -s 31981 -S -W LATIN1 archivo.shp nombre_tabla > nombre_tabla.sql
```

Notas:

- s para el Sistema de referencia
- S crea geometrías simples
- W para establecer la codificación de caracteres

```
$ psql -h localhost -U postgres -d datosgis -f nombre_tabla.sql
```

```
richard@debianGIS: ~/datosGIS/REFERENCIADOS/la_coronilla 135x16
richard@debianGIS:~/datosGIS/REFERENCIADOS/la_coronilla$ shp2pgsql -s 31981 -S -W LATIN1 cat_padrones_la_coronilla.shp cat_padrones_la_coronilla > cat_padrones.sql
Shapefile type: Polygon
Postgis type: POLYGON[2]
richard@debianGIS:~/datosGIS/REFERENCIADOS/la_coronilla$ psql -U richard -d sigrocha -f cat_padrones.sql █
```

7 – Análisis espacial



7.3) Importar archivos shapefile a la base de datos:

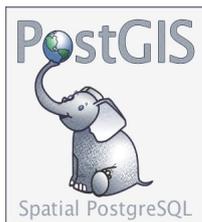
Ejercicios:

1 – Se cuenta con datos pertenecientes a la cartografía del departamento de Montevideo, se pide cargar el siguiente listado de capas en se encuentran en shapefile a nuestra base de datos geográfica “**datosgis**”.

- Límite departamental (depto_p.shp)
- Manzanas de Montevideo (v_mdg_manzanas.shp)
- Ejes de calles (v_mdg_vias.shp)
- Espacios libres de Montevideo (v_mdg_espacios_libres)
- Servicios Comerciales (serv_comerciales)

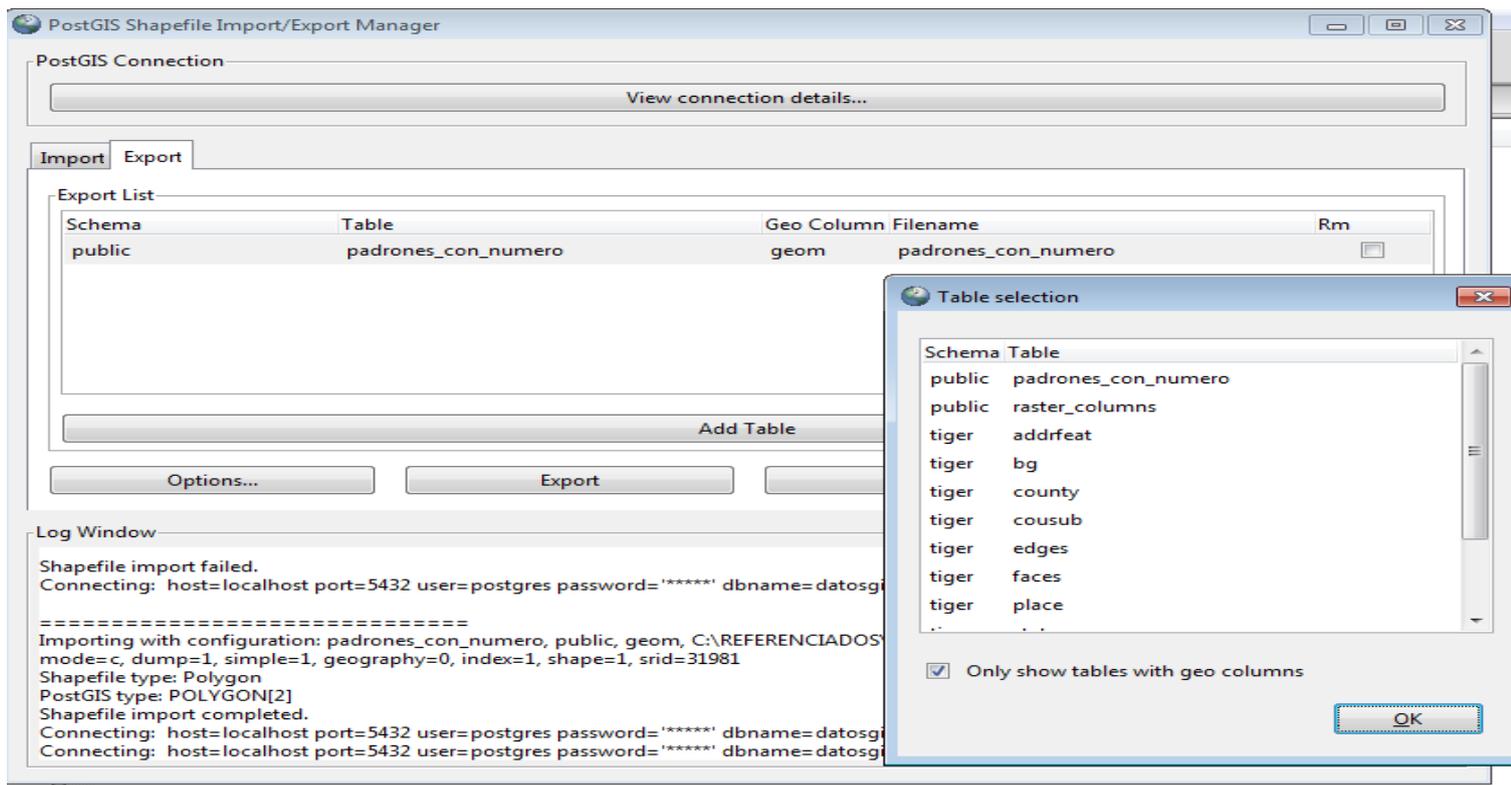
IMPORTANTE: las geometrías cargadas deberán ser del tipo “Single”, **no** multi geometry, el Sistema de Referencia de las tablas importadas deberá estar establecido en UTM Zona 21 Sur datum SIRGAS 2000 (EPSG:31981) y deberán contar con su índice de tipo gist correspondiente.

7 – Análisis espacial



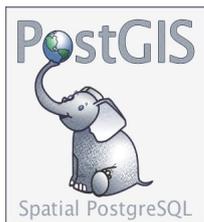
7.4) Exportar tabla desde base de datos a shapefile:

PgAdmin -> Plugins -> *Postgis Shapefile Import/Export Manager*



- Seleccionar la pestaña “Export”
- Presionar sobre “Add Table” y elegir la tabla a exportar
- Seleccionar “Options...” para establecer opciones avanzadas (opcional)
- Presionar “Export” para proceder a la exportación

7 – Análisis espacial



7.5) Exportar tabla desde base de datos a shapefile: Por línea de comandos:

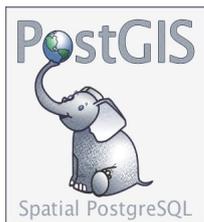
```
$ pgsq2shp -h localhost -u postgres -P password -f archivo.shp datosgis  
nombre_tabla
```

Notas:

- h (indica nombre del servidor)
- u (usuario de conexión a la base de datos)
- P (password usuario de base de datos)
- f (nombre del archivo shapefile a guardar)
- Datosgis (nombre base de datos)
- nombre_tabla (nombre de la tabla almacenada en la base de datos)

```
richard@debianGIS: ~/datosGIS/REFERENCIADOS/la_coronilla  
richard@debianGIS: ~/datosGIS/REFERENCIADOS/la_coronilla 133x12  
Dumping: XXXXXXXXXXXX [1163 rows].  
richard@debianGIS:~/datosGIS/REFERENCIADOS/la_coronilla$ clear  
  
richard@debianGIS:~/datosGIS/REFERENCIADOS/la_coronilla$ pgsq2shp -h localhost -u richard -P rife -f archivo.shp sigrocha cat_pa  
illa  
Initializing...  
Done (postgis major version: 2).  
Output shape: Polygon  
Dumping: XXXXXXXXXXXX [1163 rows].  
richard@debianGIS:~/datosGIS/REFERENCIADOS/la_coronilla$
```

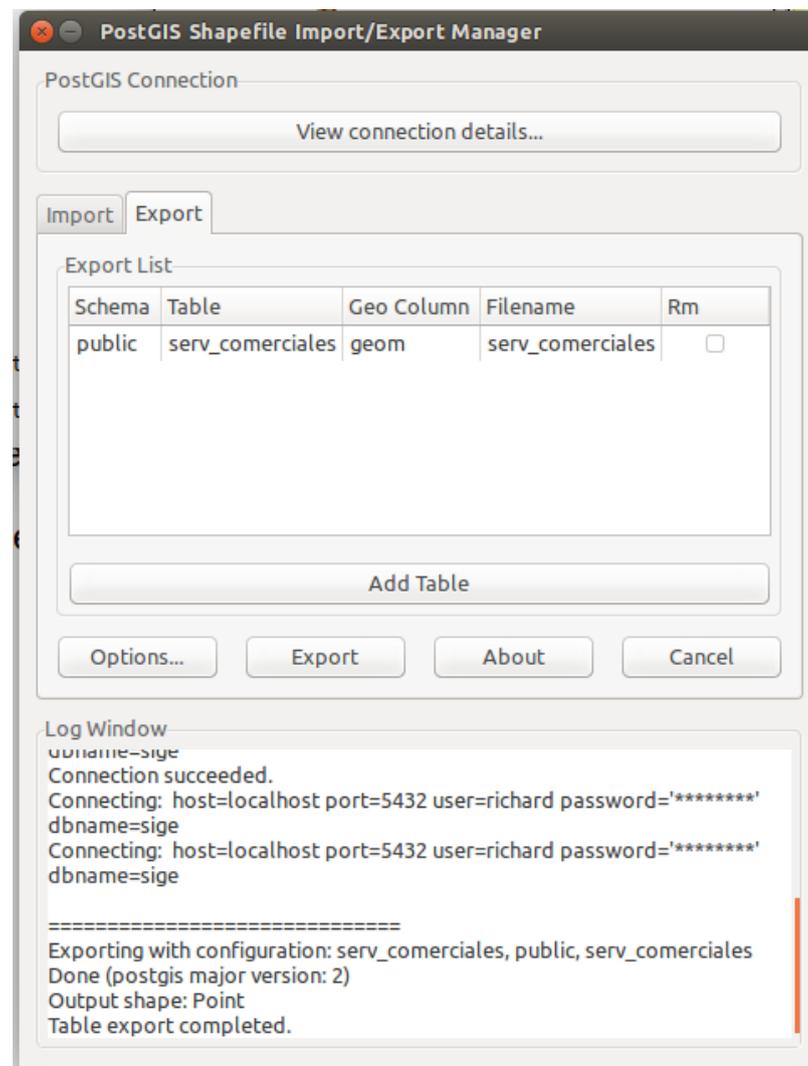
7 – Análisis espacial



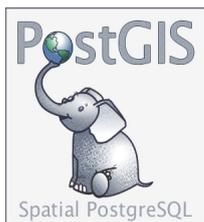
7.6) Exportar tabla desde base de datos a shapefile:

Ejercicio:

- Exportar nuevamente en shapefile la tabla con la ubicación de los Servicios Comerciales de Montevideo cargada anteriormente.



7 – Análisis espacial



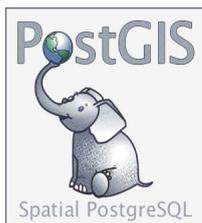
7.7) Crear una tabla en la base de datos para agregar nuevas geometrías: Desde PgAdmin

The screenshot shows the PgAdmin interface with a tree view on the left and a context menu on the right. The tree view shows a database named 'datosgis' with several objects, including a 'Tables (1)' folder containing 'spatial_ref_sys'. The 'Tables (1)' folder is selected, and a right-click context menu is open. The 'New Object' option is highlighted, and a sub-menu is displayed with 'New Table...' as the first option.

Refresh	
Count	
New Object	New Table...
Delete/Drop...	New Column...
Drop cascaded...	New Foreign Key...
Truncate	New Exclusion Constraint...
Truncate Cascaded	New Unique Constraint...
Reset table statistics	New Check...
Scripts	New Index...
View Data	New Rule...
Reports	New Trigger...
Maintenance...	
Backup...	
Restore...	
Import...	
Properties...	

- Presionar sobre “Tables” , clic con botón derecho del mouse, “New Object” -> “New Table”

7 – Análisis espacial



7.7) Crear una tabla en la base de datos para agregar nuevas geometrías: Desde PgAdmin

- Asignamos nombre y propietario de tabla

New Table...

Properties Definition Inherits Like Columns Constraints Auto-vacuum Privileges

Name:

OID:

Owner:

Schema:

Comment:

Use Slony:

Ayuda Aceptar Cancelar

New Table...

Properties Definition Inherits Like Columns Constraints Auto-vacuum Privileges

Column name	Definition	Inherited f
gid	bigint NOT NULL	
nombre	character varying(50)	
numero	integer	
ubicacion	character varying(120)	

Ayuda Add Remove Aceptar Cancelar

- Presionamos “Aceptar” para terminar

- Definimos el nombre y tipo de cada columna

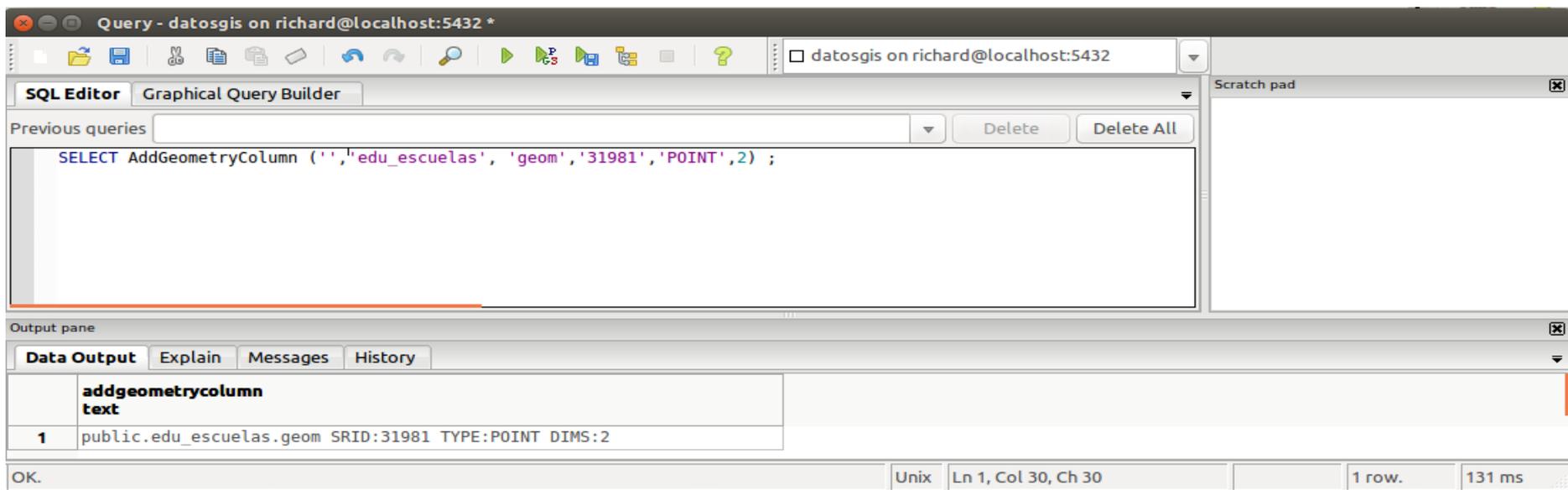
7 – Análisis espacial



7.7) Crear una tabla en la base de datos para agregar nuevas geometrías: Desde PgAdmin

- Creamos la columna geométrica (geom) para guardar nuestros datos con geometrías.

- Tools -> Query Tool



SELECT AddGeometryColumn ("','nombre_tabla', 'geom', '31981', 'POINT', 2) ;

Notas: **geom**: nombre de columna geométrica
31981: código EPSG del Sistema de Referencia
POINT: tipo de geometría (POINT, LINESTRING, POLYGON....)
2: dimensión del dato (ej: 2D, coordenadas X e Y)

7 – Análisis espacial

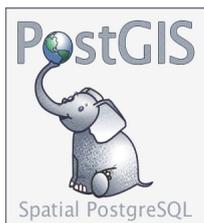


7.8) Crear una tabla en la base de datos para agregar nuevas geometrías: Utilizando línea de comandos:

```
# CREATE TABLE edu_escuelas  
(  
  gid bigint NOT NULL,  
  nombre character varying(50),  
  numero integer,  
  ubicacion character varying(120)  
);  
ALTER TABLE public.edu_escuelas OWNER TO postgres;  
  
# SELECT AddGeometryColumn ('', 'edu_escuelas', 'geom', '31981', 'POINT', 2);
```

Notas: **geom**: nombre de columna geométrica
31981: código EPSG del Sistema de Referencia
POINT: tipo de geometría (POINT, LINESTRING, POLYGON....)
2: dimensión del dato (ej: 2D, coordenadas X e Y)

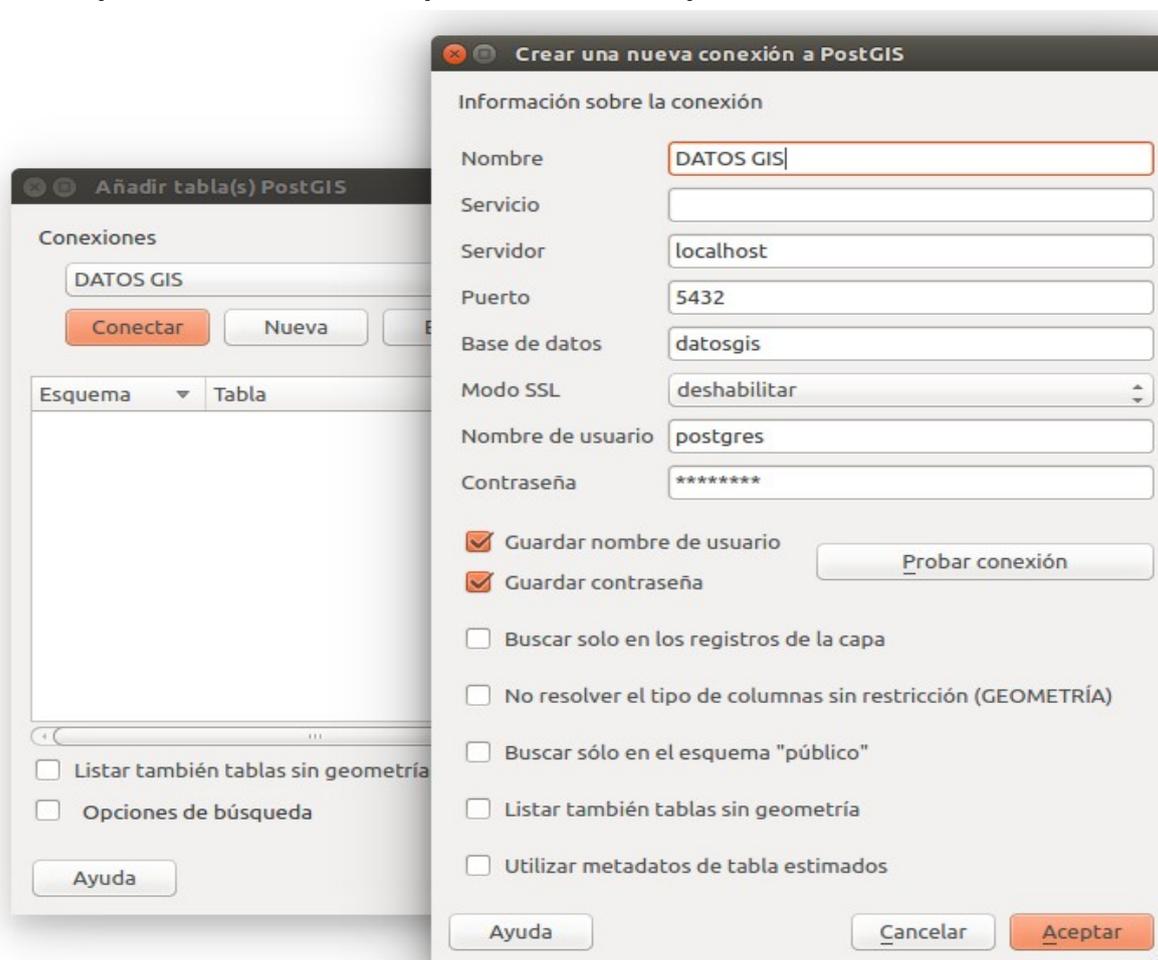
7 – Análisis espacial



7.9) Conexión desde un software SIG de escritorio.

- Ejemplo de conexión con QGIS.

Menú Capa -> Añadir capa PostGis, presionar botón "Nueva".



Ingresar los datos:

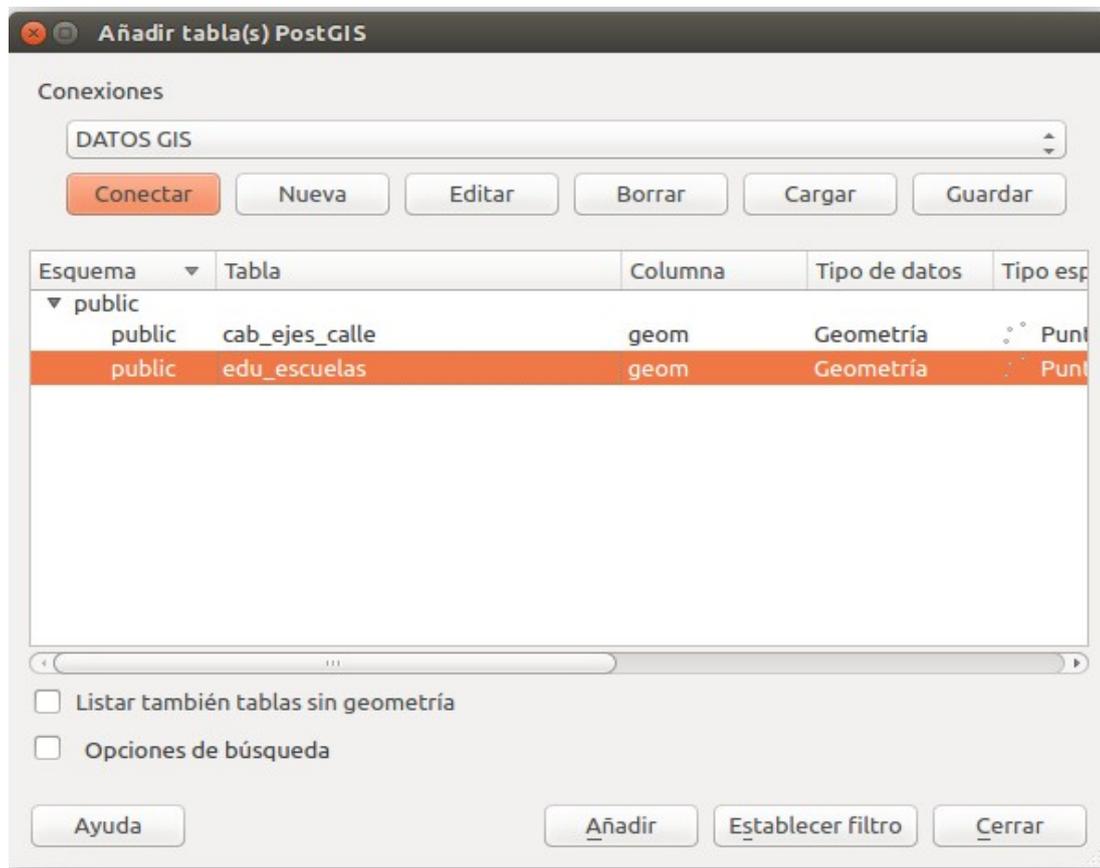
- Servidor
- Puerto
- Base de datos
- Nombre de usuario
- Contraseña

7 – Análisis espacial



7.9) Conexión desde un software SIG de escritorio.

- Ejemplo de conexión con QGIS.



- Presionar “Conectar” para establecer la conexión y seleccionar las tablas.
- Luego presionar “Añadir” para cargar la tabla seleccionada.

7 – Análisis espacial



7.10) Insertar, actualizar o eliminar elementos.

- Ejercicios:

- 1 – Cargar los datos importados en la base de datos desde el cliente SIG de escritorio QGIS.
- 2 – Cargar la tabla **edu_escuelas** creada anteriormente.
- 3 – Agregar una nueva escuela en la intersección de 18 de Julio y Carlos Roxlo, la misma deberá contar con los siguientes datos:
 - Nombre = Escuela Roxlo
 - Número = 100
 - Ubicación = 18 de Julio esq. Carlos Roxlo
- 4 – Guardar los cambios de la capa.
- 5 – Agregar1 nuevo campo “**tipo**” de tipo texto variable, ingresar el valor 'Escuela Pública'.
- 6 – Guardar los cambios y terminar la edición de la capa.
- 7 – Verificar en PgAdmin la creación de este nuevo campo y que contenga el valor ingresado.

7 – Análisis espacial

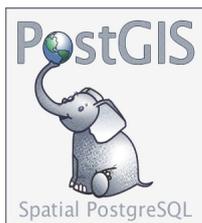


7.11) Geprocesos geográficos.

- Ejercicios:

- 8 – Establecer un área de influencia de 500 metros para la escuela creada, guardar el resultado en shapefile (realizar proceso desde menú Vectorial -> Herramientas de geoprocesos -> Buffer(s).
Ej. SQL - *SELECT gid,ST_Buffer(geom,500) as geom FROM edu_escuelas ;*
- 9 – Establecer cual es la carpeta catastral perteneciente a la ubicación de los distintos locales correspondientes a los Servicios Comerciales.
- Crear un nuevo shapefile con el resultado (realizar geoproceso desde menú QGIS)
Ej. SQL - *SELECT a.gid,a.geom,a.tipo, b.carpeta_ca FROM serv_comerciales a JOIN v_mdg_manzanas b ON ST_intersects(a.geom,b.geom) ;*
- 10 – Agregar en tabla geométrica v_mdg_espacios_libres una columna para el área y otra para el perímetro, llenar las filas con los valores correspondientes (realizarlo desde las herramientas de QGIS).
Ej. SQL - *SELECT gid, nom_parque, ST_perimeter(geom) as perimetro from v_mdg_espacios_libres ;*
- Ej. SQL resultado en Km2 -
*SELECT gid, nom_parque, (ST_perimeter(geom)*0.01) as perimetro from v_mdg_espacios_libres ;*

8 - RASTER

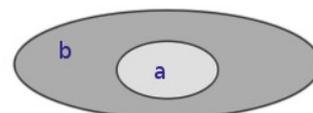


- Es posible cargar y manipular datos raster, el objetivo de PostGIS Raster es hacerlo lo más parecido posible el tipo Geometry, ofrecer un único conjunto de funciones que operen de manera transparente ya se trate de datos vectoriales .
- Una columna de tipo RASTER es una cobertura completa, con metadatos y geo referenciada.
- Permite la carga y teselado de coberturas completas, pero cada tesela por separado contiene sus propios metadatos y puede ser tratada como un objeto raster individual.

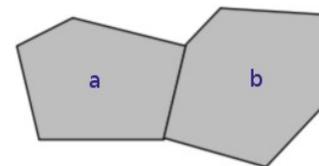
9 - TOPOLOGÍA

Proporciona métodos para verificar el cumplimiento de determinadas relaciones entre geometrías distintas, indican si cumplen o no una determinada relación espacial.

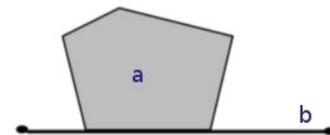
Within(a,b)



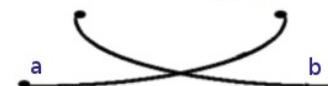
Touches(a,b)



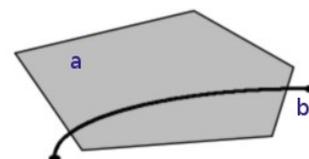
Touches(a,b)



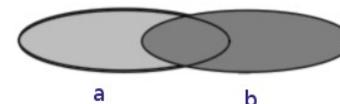
Crosses(a,b)



Crosses(a,b)



Overlaps(a,b)

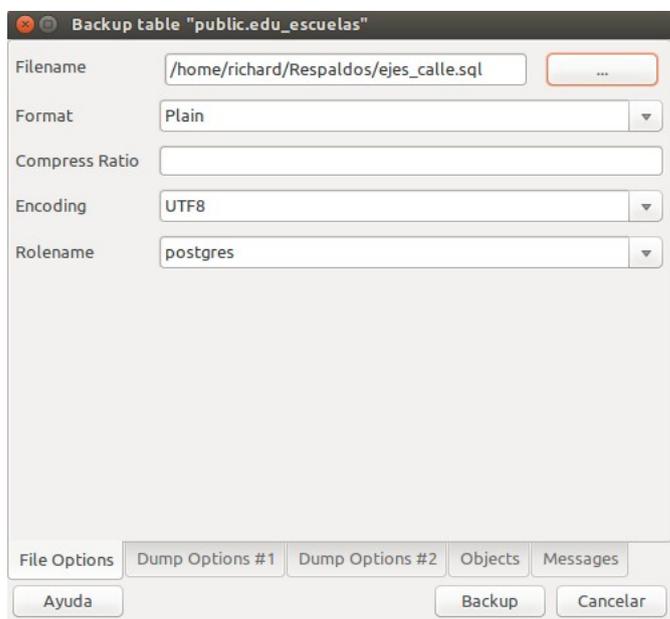


10 – COPIA DE SEGURIDAD

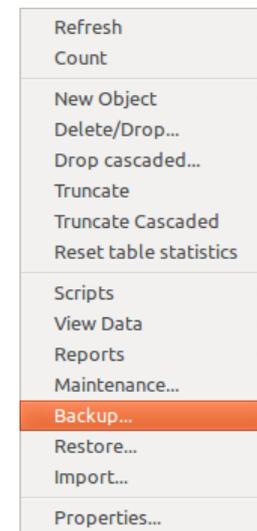
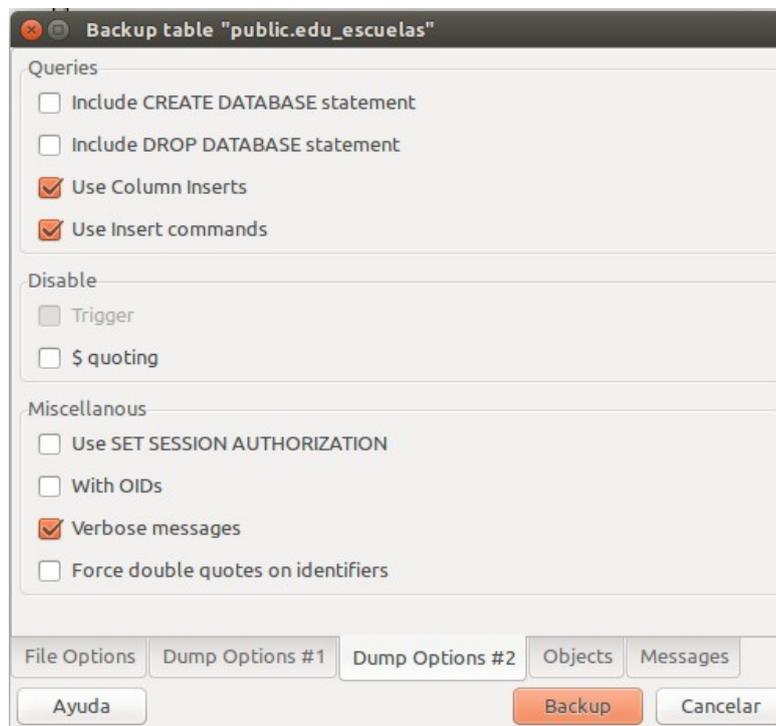


10.1) Respaldar una tabla.

- Desde pgAdmin:
 - Clíc botón derecho sobre nombre de tabla -> “Backup...”
 - Elegimos nombre de archivo, encoding, rolename



- Seleccionamos “Dump Options #2” y marcamos las opciones “Use Column inserts” y “Use insert commands”.



- Presionar “Backup” para guardar el archivo de respaldo.

10 – COPIA DE SEGURIDAD



10.2) Restaurar un respaldo guardao.

- Desde pgAdmin:
 - *Tools -> Query Tools*

```
Query - datogsis on richard@localhost:5432 - [/home/richard/datosGIS/REFERENCIADOS/aguas_dulces/cat_padrones_aguas_dulces.sql]
SQL Editor Graphical Query Builder
Previous queries [Delete] [Delete All]
SET CLIENT_ENCODING TO UTF8;
SET STANDARD_CONFORMING_STRINGS TO ON;
BEGIN;
CREATE TABLE "cat_padrones_aguas_dulces" (gid bigint NOT NULL DEFAULT nextval('gid_global'::regclass),
"padron" integer,
"localidad" varchar(50));
ALTER TABLE "cat_padrones_aguas_dulces" ADD PRIMARY KEY (gid);
SELECT AddGeometryColumn('','cat_padrones_aguas_dulces','geom','31981','POLYGON',2);
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51947','AGUAS DULCES','0103000020ED7C0000010000002000000065C7CEEE...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51971','AGUAS DULCES','0103000020ED7C00000100000005000000065C7CEEE...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51970','AGUAS DULCES','0103000020ED7C0000010000000500000005347A871...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51950','AGUAS DULCES','0103000020ED7C0000010000000500000003FF6C2811...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51969','AGUAS DULCES','0103000020ED7C000001000000050000000558B28512...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51951','AGUAS DULCES','0103000020ED7C00000100000005000000073EB44DA3...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51968','AGUAS DULCES','0103000020ED7C0000010000000500000002642DD1A4...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51952','AGUAS DULCES','0103000020ED7C00000100000005000000058CDF4A35...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51967','AGUAS DULCES','0103000020ED7C000001000000050000000F7C88E458...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51953','AGUAS DULCES','0103000020ED7C00000100000005000000043FA46D6...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51966','AGUAS DULCES','0103000020ED7C000001000000050000000C84F40A67...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51954','AGUAS DULCES','0103000020ED7C0000010000000500000002B9154378...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51965','AGUAS DULCES','0103000020ED7C0000010000000500000006EC8E6D9...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51955','AGUAS DULCES','0103000020ED7C0000010000000500000004CB0D2FB9...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51964','AGUAS DULCES','0103000020ED7C0000010000000500000003F534037A...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51956','AGUAS DULCES','0103000020ED7C000001000000050000000779182C58...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51963','AGUAS DULCES','0103000020ED7C00000100000005000000010DAF100C...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51957','AGUAS DULCES','0103000020ED7C000001000000050000000A272328FD...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51962','AGUAS DULCES','0103000020ED7C000001000000050000000E160A3CAD...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51958','AGUAS DULCES','0103000020ED7C000001000000050000000CC53E258E...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51961','AGUAS DULCES','0103000020ED7C000001000000021000000B2E75494F...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51960','AGUAS DULCES','0103000020ED7C0000010000000500000001DB68B2F...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('51959','AGUAS DULCES','0103000020ED7C000001000000050000000357FFDFA0...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('50238','AGUAS DULCES','0103000020ED7C000001000000050000000809E16490...');
INSERT INTO "cat_padrones_aguas_dulces" ("padron","localidad","geom") VALUES ('50187','AGUAS DULCES','0103000020ED7C000001000000021000000665FDC030...');
```

- Seleccionamos *File -> Open*, elegimos el archivo guardado anteriormente
 - Luego seleccionamos *Query -> Execute...*
- De esta manera se restauran los datos respaldados.