

Formation PostGIS

Vecteurs – analyse et structure / gestion des tables - Mars 2018

Fabien Guerreiro
Supports sous Licence Ouverte Etalab

Crédits : détails des crédits dans les supports.

INSTITUT NATIONAL SUPÉRIEUR DES SCIENCES AGRONOMIQUES, DE L'ALIMENTATION ET DE L'ENVIRONNEMENT

MEMBRE DE



Jointures attributaires

Les requêtes

- ... WHERE
 NomTable1.NomColonne1 =
 NomTable2.NomColonne2
- SELECT <colonnes> FROM <table1>
 JOIN <table2> ON <condition de
 jointure>

COMMUNE				
#	Name	Type	Null	Default
0	<u>PKUID</u>	INTEGER	Y	
1	Geometry	MULTIPOLYGON	Y	
2	ID_BDCARTO	INTEGER	Y	
3	NOM_COMM	TEXT(50)	Y	
4	INSEE_COMM	TEXT(5)	Y	
5	STATUT	TEXT(20)	Y	
6	X_COMMUNE	INTEGER	Y	
7	Y_COMMUNE	INTEGER	Y	
8	SUPERFICIE	INTEGER	Y	
9	POPULATION	INTEGER	Y	
10	INSEE_CANT	TEXT(2)	Y	
11	INSEE_ARR	TEXT(1)	Y	
12	NOM_DEPT	TEXT(30)	Y	
13	INSEE_DEPT	TEXT(2)	Y	
14	NOM_REGION	TEXT(30)	Y	
15	INSEE_REG	TEXT(2)	Y	

IRIS_extrait72				
#	Name	Type	Null	Default
0	<u>PKUID</u>	INTEGER	Y	
1	Geometry	MULTIPOLYGON	Y	
2	DepCom	TEXT(5)	Y	
3	Nom_Com	TEXT(40)	Y	
4	Iris	TEXT(4)	Y	
5	DcomIris	TEXT(9)	Y	
6	Nom_Iris	TEXT(40)	Y	
7	Typ_Iris	TEXT(1)	Y	
8	Origine	TEXT(1)	Y	

Exercice

A l'aide du constructeur graphique de requête, rechercher les IRIS des communes simples, et trier le résultat par nom d'IRIS.

Doivent apparaître dans le résultat, la géométrie des IRIS, leur nom, le nom de leur commune, et le nom de leur département.

Jointures spatiales

Comparaison

ST_Equals(geometry A, geometry B)

ST_Intersects(geometry A, geometry B)

ST_Disjoint(geometry A, geometry B)

ST_Crosses(geometry A, geometry B)

ST_Overlaps(geometry A, geometry B)

ST_Touches(geometry A, geometry B)

ST_Within(geometry A, geometry B)

ST_Contains(geometry A, geometry B)

Les fonctions suivantes sont également intéressantes :

ST_Dwithin(geometry A, geometry B, distance)

ST_Distance(geometry A, geometry B)

Géométrie dans les agrégations

GROUP BY

toutes les colonnes en sortie sauf celle du critère de regroupement doivent faire l'objet d'une fonction d'agrégation, ce doit être également le cas pour la géométrie

- **ST_UNION(geom)**
- **ST_Multi()**
- **ST_Linemerge()**

QGIS demande un identifiant unique, à recréer avec la requête :

- **row_number() over()**

Exercice pratique

1. Quels sont les ponctuels hydrographiques de la commune de La Flèche ?
2. Quelle est la longueur de la 'rivière le loir' dans chacune des communes intersectées par le cours d'eau ?
3. Sélectionner les 'PONCTUELS HYDROGRAPHIQUES' qui sont à moins de 5 km d'un établissement d'enseignement (couche ETABLISSEMENT)
4. Quel est l'établissement le plus proche du centroïde de la commune de la Flèche?

On utilisera les coordonnées X_COMMUNE et Y_COMMUNE et la fonction **ST_Makepoint()**. Le SRID (Identifiant du Système de Référence Spatial) est 2154, mais on pourra le cas échéant généraliser la requête à tout SRID en utilisant la fonction **ST_Srid()** qui récupère le srid d'une géométrie.

Définition d'une table

Dbmanager (QGIS)

Info Table Aperçu

commune

Informations générales

Type de relation : Table
 Propriétaire : postgres
 Pages : 5
 Lignes (estimation) : 19
 Lignes (comptées) : 19
 Privilèges : select, insert, update, delete

PostGIS

Colonne : geom
 Géométrie : MULTIPOLYGON
 Dimension : 2
 Réf. spatiale : WGS 84 (4326)
 Emprise : (inconnu) [calculer](#)

Champs

#	Nom	Type	Longue
1	id	int4	4
2	geom	geometry (MultiPolygon,4326)	
3	id_bdcarto	int4	4
4	nom_comm	varchar (50)	
5	insee_comm	varchar (5)	
6	statut	varchar (20)	
7	x_commune	int4	4
8	y_commune	int4	4
9	superficie	int4	4
10	population	int4	4
11	insee_cant	varchar (2)	
12	insee_arr	varchar (1)	
13	nom_dept	varchar (30)	
14	insee_dept	varchar (2)	
15	nom_region	varchar (30)	
16	insee_reg	varchar (2)	

Contraintes

Nom	Type	Colonne(s)
COMMUNE_pkey	Clé primaire	id

Index

Nom	Colonne(s)
sidx_commune_geom	geom

```
-- DROP TABLE commune;

CREATE TABLE commune
(
  id integer NOT NULL DEFAULT nextval('"COMMUNE_id_seq"'::regclass),
  geom geometry(MultiPolygon,4326),
  id_bdcarto integer,
  nom_comm character varying(50),
  insee_comm character varying(5),
  statut character varying(20),
  x_commune integer,
  y_commune integer,
  superficie integer,
  population integer,
  insee_cant character varying(2),
  insee_arr character varying(1),
  nom_dept character varying(30),
  insee_dept character varying(2),
  nom_region character varying(30),
  insee_reg character varying(2),
  CONSTRAINT "COMMUNE_pkey" PRIMARY KEY (id)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE commune
  OWNER TO postgres;

-- Index: sidx_commune_geom
-- DROP INDEX sidx_commune_geom;

CREATE INDEX sidx_commune_geom
ON commune
  USING gist
  (geom);
```

PgAdmin

Définition SQL d'une table

Base

```
CREATE TABLE commune  
(id integer NOT NULL, id_bdcarto integer, [...] );
```

Séquence

```
DEFAULT nextval("COMMUNE_id_seq"::regclass)
```

Clé primaire

```
ALTER TABLE commune  
ADD CONSTRAINT "COMMUNE_pkey" PRIMARY KEY(id);
```

Droits

```
ALTER TABLE commune OWNER TO postgres;
```

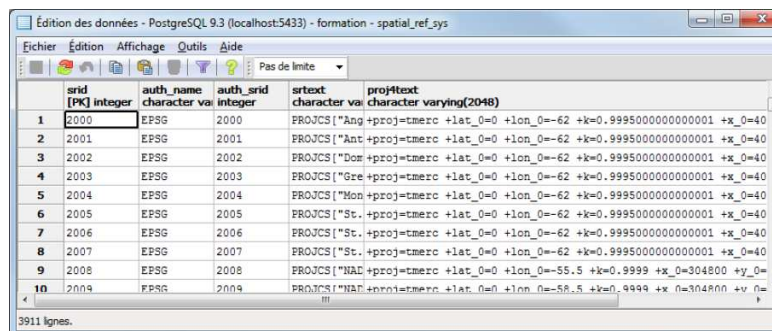
Géométrie

```
ALTER TABLE commune ADD COLUMN geom geometry(MultiPolygon,2154);
```

La géométrie d'une table

La table spatial_ref_sys

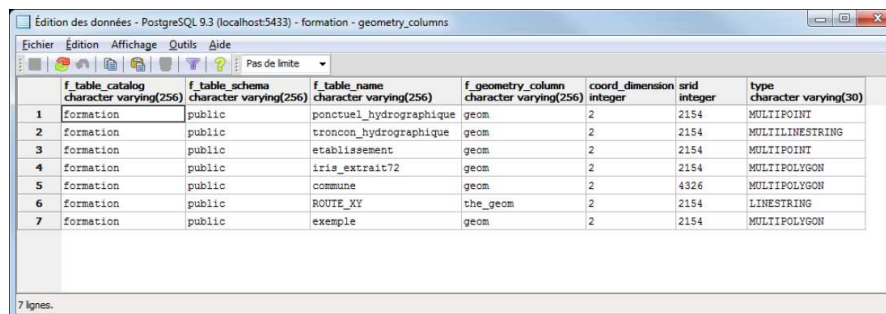
Références EPSG



srid	auth_name	auth_srid	srs_text	proj4text
[PK] integer	character varying	integer	character varying(256)	character varying(2048)
1	2000	EPSG	2000	PROJCS["Ang+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
2	2001	EPSG	2001	PROJCS["Ant+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
3	2002	EPSG	2002	PROJCS["Dom+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
4	2003	EPSG	2003	PROJCS["Gre+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
5	2004	EPSG	2004	PROJCS["Mon+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
6	2005	EPSG	2005	PROJCS["St.+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
7	2006	EPSG	2006	PROJCS["St.+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
8	2007	EPSG	2007	PROJCS["St.+proj=tmerc+lat_0=0+lon_0=-62+k=0.9995000000000001+x_0=40
9	2008	EPSG	2008	PROJCS["NAD+proj=tmerc+lat_0=0+lon_0=-55.5+k=0.9999+x_0=304800+y_0=
10	2009	EPSG	2009	PROJCS["NAD+proj=tmerc+lat_0=0+lon_0=-58.5+k=0.9999+x_0=304800+y_0=

La vue geometry_columns

Géométrie des tables



f_table_catalog	f_table_schema	f_table_name	f_geometry_column	coord_dimension	srid	type
character varying(256)	character varying(256)	character varying(256)	character varying(256)	integer	integer	character varying(30)
formation	public	pointuel_hydrographique	geom	2	2154	MULTIPOINT
formation	public	troncon_hydrographique	geom	2	2154	MULTILINESTRING
formation	public	etablissement	geom	2	2154	MULTIPOINT
formation	public	iris_extrait72	geom	2	2154	MULTIPOLYGON
formation	public	commune	geom	2	4326	MULTIPOLYGON
formation	public	ROUTE_XY	the_geom	2	2154	LINESTRING
formation	public	exemple	geom	2	2154	MULTIPOLYGON

L'index spatial

CREATE INDEX sidx_commune_geom

ON commune

USING gist

(geom);

La vue geography_columns (coordonnées sphériques)

La projection d'une table

geometry_columns

Vue des géométries

Modifier la projection d'une table

```
SELECT UpdateGeometrySrid ('NomTable','geometry' ,srid);
```

Reprojeter une table

```
ST_Transform(geometry,srid)
```

Effacer la géométrie

```
SELECT DropGeometryColumn ('NomSchema', 'NomTable','geom');
```

Effacer une table géométrique

```
DROP TABLE NomTable;
```

Exercice

Créer la table *commune_wgs84*, à partir de la table *commune* reprojétée en WGS84

Le schéma, la table et la vue

Le schéma

```
CREATE SCHEMA nom_du_schema ;  
SELECT * FROM nom_schema.nom_table ;
```

La table

```
CREATE TABLE nom_table AS [..]  
ALTER TABLE, INSERT INTO et DROP
```

La vue

```
CREATE VIEW nom_vue AS [..]
```

Exercice pratique

1. Charger les couches suivantes dans PostGIS comme dans l'exercice du module précédent :

/BD_TOPO/I_ZONE_ACTIVITE/PAI_SANTE.SHP

/BD_TOPO/H_ADMINISTRATIF/COMMUNE.SHP

/BD_TOPO/E_BATI/BATI_INDUSTRIEL.SHP

/BD_TOPO/F_VEGETATION/ZONE_VEGETATION.SHP

Créer une nouvelle vue **BATI_INDUSTRIEL10** et la charger dans QGIS en sélectionnant dans la table BATI_INDUSTRIEL les 'Bâtiment industriels' (attention à la majuscule!) dont la hauteur est d'au moins 10 m

2. Créer la vue **conifere** les 'Forêt fermée de conifères' de la commune de La Flèche. Ne pas oubliez de mettre une condition de jointure entre les deux couches... qui devra être ici spatiale.
3. Calculer la somme des surfaces des 'Forêt fermée de feuillus' de la commune de la Flèche en ha (1ha = 10 000 m²), en faisant attention à ne prendre en compte que les parties de surfaces des polygones réellement situées à l'intérieur de la commune.