PostgreSQL / Postgis Premiers pas

Mars 2012

Ressources, territoires, habitats et logement Énergie et climat Développement durable don des risques Infrastructures, transports et m_{er}

Présent

pour l'avenir

Conseillers en Management des SIG du Sud-Est Centre de Prestations et d'ingénierie Informatiques



Contexte

Un besoin des services de découvrir Postgis

- Le SGBD à composante spatiale Postgis est de plus en plus présent dans les applications utilisées par les services
- Certains souhaitent avoir une vision de ce à quoi sert Postgis et de son utilité éventuelle dans les projets locaux

La démarche des CMSIG du Sud-Est

- Les CMSIG ont donc participé à une formation d'introduction à Postgis (oct. 2011)
- Afin d'en faire un retour, une première présentation a été réalisée en groupe interrégional le 16 novembre 2011
- Ce document retrace le cas d'usage présenté lors de ce groupe

Le cas d'usage : description

- Imaginons un utilisateur (Toto) qui a besoin de la couche des objets géographiques
 EPCI sur sa zone d'action
- Les données sources sont gérées par deux personnes : un administrateur des référentiels (qui gère les contours communaux) et un gestionnaire métier qui tient à jour un tableau de correspondance « communes – EPCI »

2

L'objectif est d'utiliser Postgis pour que Toto puisse avoir ses contours d'EPCI automatiquement avec les mises à jour en temps réel



Plan du diaporama

Cliquer sur les étapes pour accéder directement à la diapo correspondante

- Installation et principe de fonctionnement de Postgis
 - > L'installation et la création d'une base de données à composante spatiale
 - L'organisation d'une base de données
- Intégrer les données dans Postgis
 - Présentation des données utilisées
 - Les données attributaires
 - Créer une table
 - Copier les données depuis un CSV
 - Les données vecteurs : utilisation du plugin SPIT de QuantumGIS
 - Créer une connexion à la base de données
 - Importer les données
- Faire une requête SQL simple (jointure)
- Créer une table issue d'une requête et l'afficher dans QGIS
- Créer une vue issue d'une requête
- Créer la vue (dynamique) des EPCI par fusion d'objets

Liberd - Egolind - Franceiste REFURLIQUE FRANÇAISE

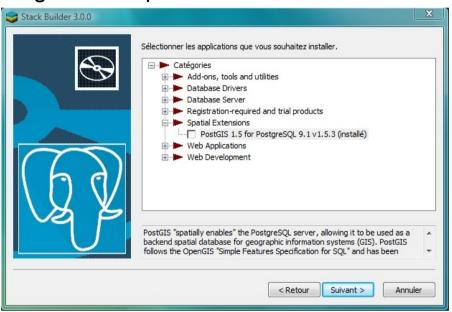
Ministère de l'Écologie, du Développement durable, des Transports et du Logèment



Installation

Installation de Postgres et de Postgis

- > PostgreSQL : télécharger l'installateur « one click » en suivant ce lien.
- <u>Postgis</u>: aller dans menu démarrer => PostgreSQL => Application stack builder Lancer la recherche d'extensions (attention : il faut rentrer les paramètres du proxy); choisir Postgis dans Spatial Extension



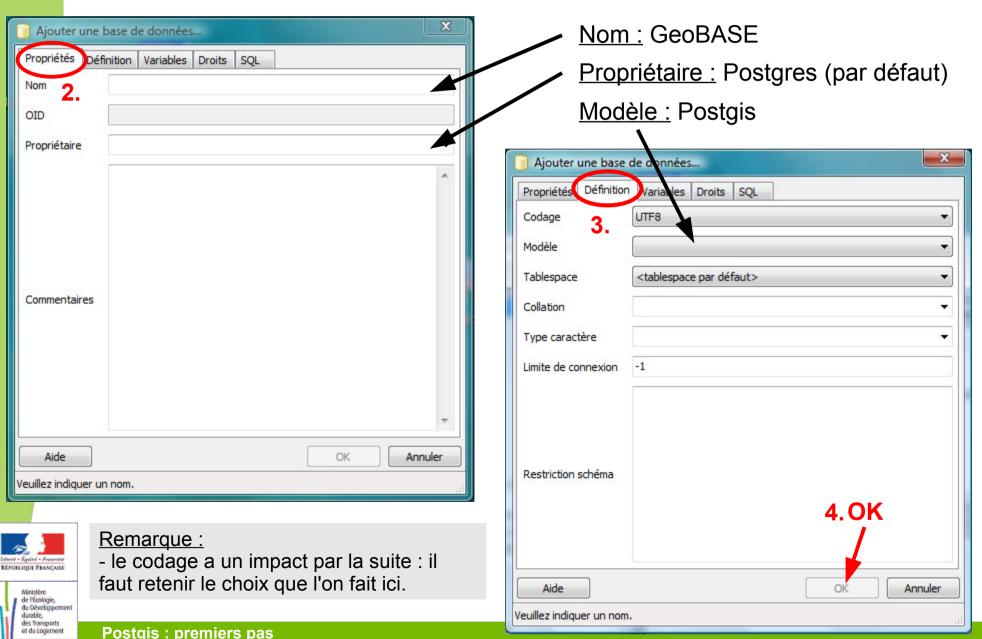
pgAdmin

- pgAdmin est une interface d'administration des bases de données Postgres
 - Il s'est installé en même temps que Postgres
 - On le trouve dans le menu démarrer ou en icône sur le bureau
- Lancer pgAdmin
 - > Dans la colonne de gauche on choisit notre serveur (taper le mot de passe)
 - Par défaut on a une base « Postgis » : elle va nous servir de modèle pour créer notre base de travail



Création d'une base de données

1. (dans pgAdmin) Clic droit sur Bases de données => Ajouter...





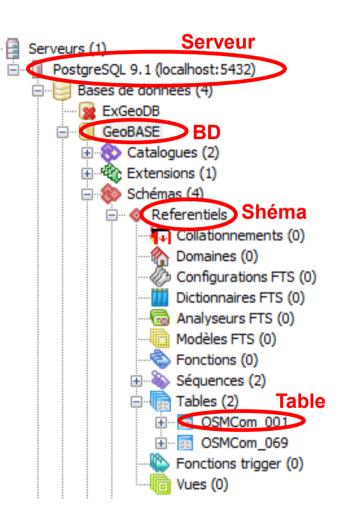
L'organisation d'une BD

Une organisation hiérachisée

- Le <u>serveur</u> est organisé en plusieurs bases de données
- Chaque <u>base de données</u> l'est en schémas (par défaut chacune a un schéma « public »
- Les schémas contiennent des tables

Interprétation

- Il est impossible de faire des requêtes avec des tables appartenant à plusieurs bases de données : attention donc lorsque l'on crée plusieurs bases
- Les schémas peuvent servir d'arborescence de données (mais à un seul niveau)
- Exemple : une base principale pour toutes les données, et 3 schémas : production, diffusion, referentiels







Présentation des données

Voici les données qui permettent de réaliser les exemples proposés

Rappel du problème

- On a un gestionnaire des contours communaux
- Une autre personne tient à jour un tableau de correspondance communes-EPCI
- On voudrait que les objets géographiques EPCI se mettent à jour dynamiquement

Les données attributaires EPCI

- Données utilisées : les données EPCI en Rhône-Alpes de territoires.gouv.fr (l'Observatoire des Politiques Territoriales)
- > Pour ce faire : cliquer ici, choisir EPCI et cliquer sur la carte (Rhône-Alpes)

Les contours communaux du Rhône

- > Lien de téléchargement des contour communaux : cliquer ici
- Dans les exercices, il est proposé de travailler avec le département du Rhône (le nom donné à la couche dans les exercices est OSMCom 069)
- Attention : le système de coordonnées n'est pas indiqué (le shapefile n'a pas de .prj) : il s'agit du WGS 84 (EPSG 4326)



Import de données attributaires

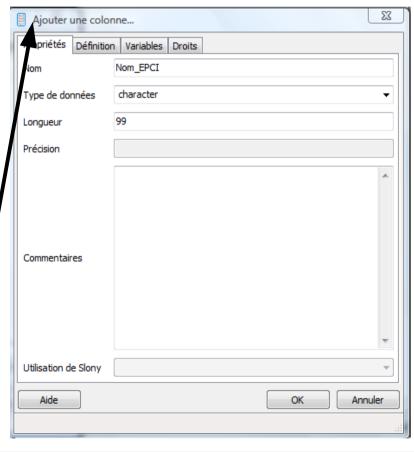
1. Créer une table

Pour importer des données attributaires

- > Il faut d'abord avoir créé la structure de la table dans Postgis
- Ensuite on copie les données depuis un CSV

Création de la structure de la table

- 1. Dans le bon schéma (ici « public ») clic droit sur tables puis Ajouter...
- 2. Rentrer le nom (OPT_epci_RA) et le propriétaire (postgres)
- 3. Dans l'onglet Colonnes remplir les colonnes à l'aide du bouton « Ajouter »
 - Code_INSEE : character (5)
 - Code SIREN: character (9)
- Nom_EPCI : character (99)
- 4. Cliquer sur OK en ignorant l'avertissement
- 5. Créer une clé primaire : clic droit sur la couche, Ajouter un objet puis dans l'onglet colonne on sélectionne Code_INSEE et Ajouter puis OK





<u>La clé primaire</u>: il s'agit du champ identifiant les objets (qui obéit donc à une règle d'unicité). Pour pouvoir modifier la table dans pgAdmin (par la commande « afficher les données ») il faut une clé primaire.

Import de données attributaires

2. Copier les données

Les données au format CSV

- > Il faut bien faire attention à avoir les mêmes types de champs dans le CSV et dans la table créée précédemment, et dans le même ordre
- > II ne faut pas de ligne avec des titres
- > Pour plus de facilité, les données pré-traitées (en csv) à partir de l'export de territoires.gouv.fr sont disponibles ici en CSV (séparateur : [,] ; délimiteur de texte : ["] ; encodage comme celui de la base : [UTF8]) : le premier champ est le code INSEE, le deuxième le code SIREN, le troisième le nom de l'EPCI

• Le requêteur SQL 🐸

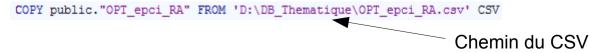


> Permet de faire des requêtes SQL, avec aussi un mode graphique



Copier les données

Exécuter la requête suivante :



Les modèles des requêtes réalisées dans ce diaporama sont disponibles en cliquant ici.





Import de données vecteur

1. Créer une connexion à la base dans QGIS

Le plugin SPIT de QGIS

- Ce plugin permet d'importer des shapefiles
- Activer le plugin s'il ne l'est pas par défaut
- Lancer le plugin et cliquer sur Nouveau

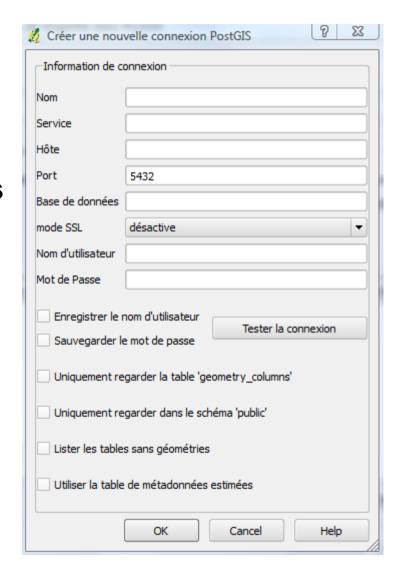
Paramétrage d'une connexion postgis

- Nom de la connexion : au choix
- Hôte (serveur) : localhost
- > Port : 5432 par défaut
- Base de données : GeoBASE
- Nom d'utilisateur : postgres (par défaut)
- Mot de passe : à renseigner selon le choix lors de l'installation de postgres

Tester puis OK

La connexion à notre base est enregistrée pour tous les outils de QGIS qui travaillent avec Postgis

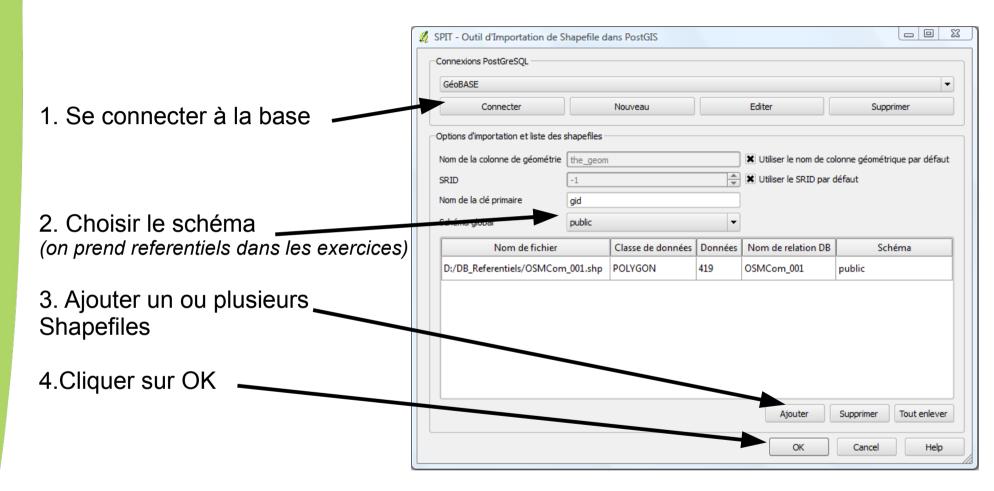






Import de données vecteur

2. Importer des shapefiles







On peut ensuite aller dans pgAdmin:

- la ou les tables apparaissent dans le schéma (il est peut-être utile de rafraîchir)
- dérouler la table permet de voir les différents champs (dans le menu colonne)
- clic droit sur la table et afficher les données permet de voir les données



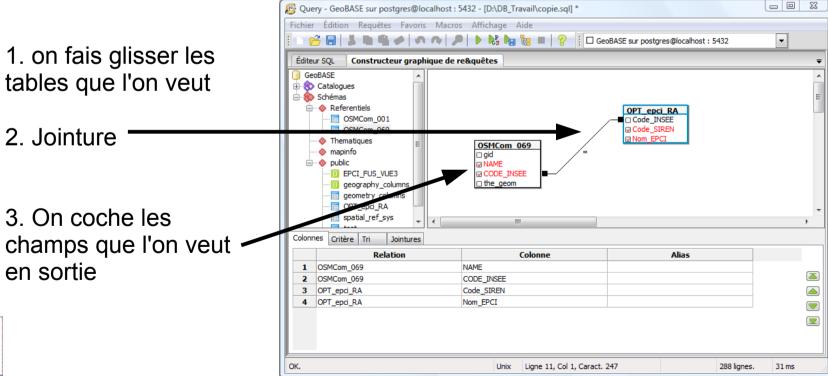
Requête attributaire simple (1/2)

Le mode graphique

Exercice

- On dispose des contours communaux du Rhône et de la correspondance communes-EPCI en Rhône-Alpes
- On souhaite avoir le tableau de correspondance communes-EPCI pour le Rhône

Onglet « Constructeur graphique de requêtes »



Zibers + Egistel - Pracentis
REPUBLIQUE FRANÇAISE

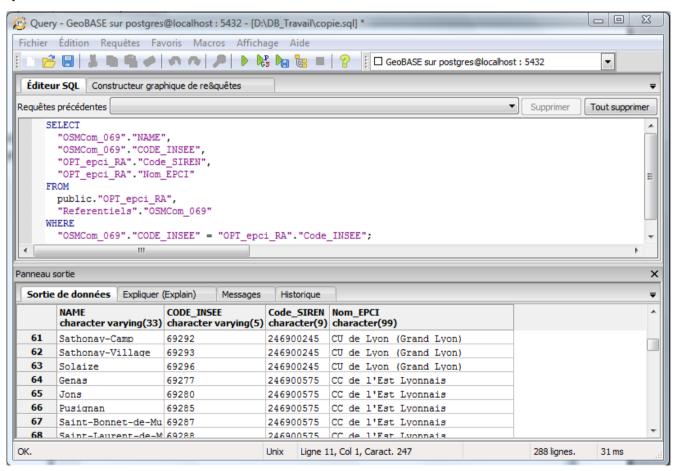
Ministère
de l'Écologie,
du Développement
durable,
des Temports
et du Logement



Requête attributaire simple (2/2)

Réalisation de la requête

- Retour à l' « éditeur SQL »
 - On peut vérifier la requête
 - > Et on peut l'exécuter





Les modèles des requêtes réalisés dans ce diaporama sont disponibles en cliquant ici.



Requête: création de table (1/2)

Réalisation de la requête

> Exercice

On souhaite créer la table des communes du Rhône avec un attribut EPCI

La requête avec création de table

- On part de la requête précédente, mais en prenant les colonnes « the_geom » (la géométrie des objets) et « gid » (l'identifiant des objets)
- Il suffit de rajouter avant la requête« CREATE table nom_de_ma_table AS »

```
CREATE table public."Com_EPCI_69" AS

SELECT

"OSMCom_069"."NAME",

"OSMCom_069"."CODE_INSEE",

"OPT_epci_RA"."Code_SIREN",

"OPT_epci_RA"."Nom_EPCI",

"OSMCom_069".gid,

"OSMCom_069".the_geom

FROM

public."OPT_epci_RA",

"Referentiels"."OSMCom_069"

WHERE

"OSMCom_069"."CODE_INSEE" = "OPT_epci_RA"."Code_INSEE";
```



Dans le nom de la table, il faut bien faire référence au schéma sous la forme "schéma"."nomtable"

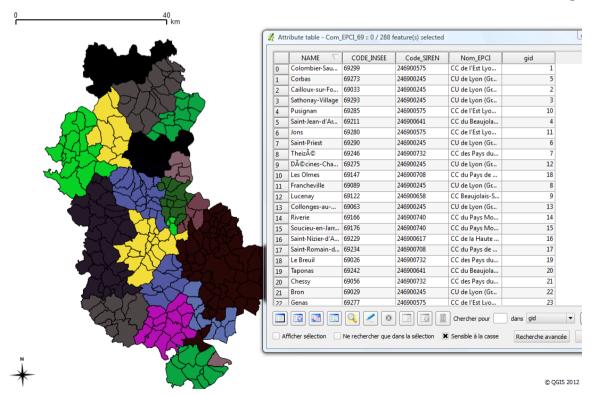
Les guillemets ["] servent à indiquer à Postgis qu'il faut respecter la casse



Requête: création de table (2/2)

Affichage des données dans QGIS

- Conecter QGIS à la base de données et ouvrir la table
 - Dans QGIS cliquer sur ajouter une couche postgis
 - On retrouve notre connexion créée en diapo 8
 - Connexion puis choisir la table précédemment créée
- On peut ensuite l'utiliser comme une couche shp







Création d'une vue

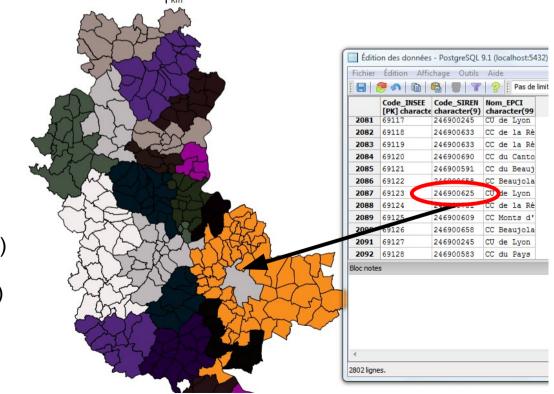
Le principe de la vue

- > Permet de travailler en dynamique : la vue est recalculée en temps réel
- > Donc si l'on modifie les tables d'origine, la vue est modifiée de la même façon
- Pour créer une vue, il suffit de remplacer « table » par « view » dans la requête précédente (en changeant le nom)

Exemple de traitement d'une vue dans QGIS

- 1. Ouvrir la vue dans QGIS (même procédure que pour une table)
- 2. Afficher les données de la table source OPT_epci_RA dans pgAdmin
- 3. Modifier une donnée de la table source
- 4. Rafraîchir la fenêtre carte de QGIS : la vue se met à jour

Exemple : affecter Lyon (INSEE 69123) à la communauté de communes du Pays de l'Arbresle (SIREN 246900625) au lieu du Grand Lyon (SIREN 246900245)







Fusion d'objets

La fonction ST_Union

- > Notre objectif de départ est de créer les objets EPCI
- Nous allons procéder par fusion des objets communes
- La fonction ST_Union(champ_objet) de Postgis permet de faire cela

La requête (voir ci-dessous)

- On met le champ the geom en argument de ST_Union
- > ST_Union est une fonction d'agrégation, il faut donc une clause GROUP BY
- > L'utilisation de AS dans la clause select permet de donner un alias aux colonnes
- > Il faut un champ identifiant des objets « gid » obligatoirement pour l'ouverture de la vue dans QGIS : nous prenons le maximum des « gid » des objets communes

```
CREATE view public."EPCI_69_vue" AS

SELECT

ST_Union("OSMCom_069".the_geom) AS the_geom,

"OPT_epci_RA"."Code_SIREN",

max("OSMCom_069"."gid") AS gid

FROM

"Referentiels"."OSMCom_069",

public."OPT_epci_RA"

WHERE

"OSMCom_069"."CODE_INSEE" = "OPT_epci_RA"."Code_INSEE"

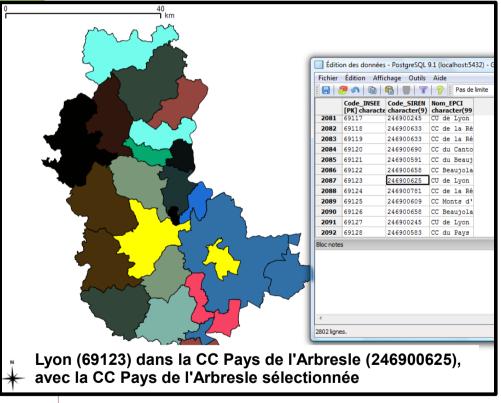
GROUP BY "OPT_epci_RA"."Code_SIREN";
```

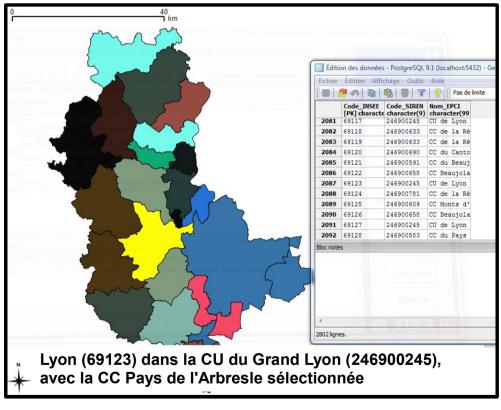


Résultat final

> En ouvrant dans QGIS et en modifiant la table source

- > On voit quand on rafraîchit que les objets sont calculés dynamiquement
- > Evidemment le temps de rafraîchissement est plus long car il y a recalcul
- On aurait pu utiliser ST_Union pour créer une table (non dynamique, mais affichage plus rapide)







Conclusion : on a réalisé une jointure avec fusion, tout cela dans une vue dynamique