

# Manipulation d'une base de données PostgreSQL

## avec l'extension spatiale PostGIS

Pour réaliser ce TP, vous devez avoir un serveur PostgreSQL avec l'extension PostGIS (et QGIS, si vous souhaitez visualiser les résultats) installés ainsi qu'avoir effectué le chargement des données disponibles. Pour chaque thème, vous trouverez la(les) question(s) et le résultat attendu

### 1 Gestion de la métabase

#### 1.1 Donnez le type de l'attribut *the\_geom* de la relation *ville*

```
udt_schema | udt_name
-----+-----
public     | geometry
(1 row)
```

#### 1.2 Donnez les types géométriques des villes

```
id | geometrie
---+-----
1  | ST_Polygon
2  | ST_MultiPolygon
3  | ST_Polygon
(3 rows)
```

### 2 Gestion des données physiques

#### 2.1 Combien de sommets forment la route 1?

```
nbrepoints
-----
3
(1 row)
```

#### 2.2 Donnez les bâtiments pour lesquels le centroïde n'est pas dans le bâtiment et la distance aux bords

id	centroide	distance
3	POINT(2.10714285714286 2.14285714285714)	0.142857142857143

(1 row)

**2.3** *Donnez le point associe dans l'objet de la question 2.2 pour mettre un libelle sur l'objet*

label
POINT(2.75 2.25)

(1 row)

**2.4** *Donnez la superficie de la ville 1*

id	surface
1	16

(1 row)

**2.5** *Donnez les villes dont la superficie est inférieure strictement à la moyenne des superficies des villes*

id
3

(1 row)

**2.6** *Donnez le périmètre de la ville 2*

id	perimetre
2	20

(1 row)

**2.7** *Donnez le point de départ et le point d'arrivée de la route 2*

depart	arrivee
POINT(11 7)	POINT(11 9)

(1 row)

**2.8 *Donnez la distance de la route 3 à la ville 2***

```
distance
-----
      3
(1 row)
```

**2.9 *Quelles villes contiennent un trou ou qui ne sont pas d'un seul tenant ?***

```
id
---
  2
  3
(2 rows)
```

**2.10 *Donnez les coordonnées du rectangle englobant du Magasin 4***

```
xmin | xmax | ymin | ymax
-----+-----+-----+-----
    0 |    1 |   10 |   11
(1 row)
```

**2.11 *Où se coupent les routes 1 et 2 ?***

```
intersection
-----
POINT(11 7.45454545454545)
(1 row)
```

**2.12 *Donnez la longueur de la route 2***

```
longueur
-----
      2
(1 row)
```

**2.13 *Quelle est la distance relative entre le début de la route 2 et son croisement avec la route 1 ?***

```
distancerelative
-----
0.227272727272727
(1 row)
```

**2.14**      *Dans quelle ville suis-je si je parcours la moitié de la route 1 ?*

```
ville
-----
  2
(1 row)
```

**2.15**      *Donnez les coordonnées de la deuxième moitié de la route 1. Validez le résultat obtenu (aux arrondis près).*

```
route
-----
LINESTRING(7.81684335937158 5.71827819602086,12 8,12 5)
(1 row)
```

**2.16**      *Donnez les coordonnées du rectangle englobant tous les magasins*

```
xmin | xmax | ymin | ymax
-----+-----+-----+-----
  0 | 3 | 1 | 11
(1 row)
```

### **3 Gestion des données logiques**

**3.1** *Donnez les routes qui traversent la ville '1'.*

```
id
----
  1
(1 row)
```

**3.2** *Donnez les routes qui se croisent.*

```
id | id
----+----
  1 | 2
(1 row)
```

**3.3** *Donnez les identifiant et nom des bâtiments dans la ville 2.*

```
id | nom
----+-----
  1 | Mairie
  2 | Ecole
(2 rows)
```

**3.4 Donnez les 'Mairie' ou 'Radio' installées en bordure de villes.**

```
batiment
-----
      1
(1 row)
```

**4 Analyse spatiale**

**4.1 Donnez la route la plus proche du bâtiment 'Mairie'.**

```
id
---
  1
(1 row)
```

**4.2 Donnez les villes contenant un magasin.**

```
id | nom
----+-----
  1 | V1
(1 rows)
```

**4.3 Donnez pour chaque route la longueur de sa partie urbaine**

```
id | longueur
----+-----
  3 |          0
  1 | 5.69543822097349
  2 |          0
(3 rows)
```

**4.4 Donnez les villes qui ne sont pas impactées par le bruit d'une route (i.e., à plus de 3 unités d'une route).**

```
id | nom
----+-----
  3 | V3
(1 row)
```

**4.5 Le bâtiment de la radio est-il visible de l'école ? (2D)**

```
ecole | radio | batiment | reponse
-----+-----+-----+-----
    2 |    5 |    1 | NON
(1 row)
```

**4.6 Le bâtiment de la radio est-il visible du bâtiment de la TV ? (2D)**

```
tv | radio | batiment | reponse
-----+-----+-----+-----
```

6 | 5 | -1 | OUI  
(1 row)

#### 4.7 Créer une relation urbaine qui contient les parties urbaines des routes avec leur nom

# 1 tuple par intersection route/ville  
# visualiser le résultat sous QGIS

## 5 Composition d'opérateurs

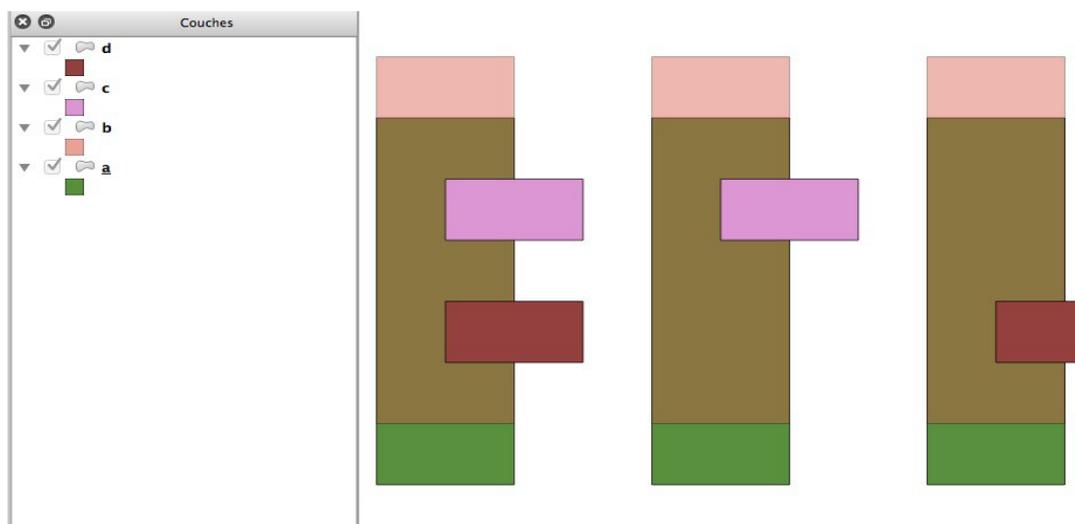
### 5.1 Approche fonctionnelle

Créez et chargez les relations. A partir de ces tuples effectuez les opérations permettant de calculer :

- Dans AIB : l'intersection de A et de B avec les identifiants associés
- Dans CIAIB : l'intersection de C avec l'intersection de A et de B en une seule requête.
- Dans DIAIB : l'intersection de D avec l'intersection de A et de B en une seule requête.

Puis supprimer les relations.

Structure de la base d'essais :



Marron : superposition de A et de B

Schémas des relations :

```
CREATE TABLE A (ID INTEGER, the_geom GEOMETRY);
CREATE TABLE B ( ID INTEGER, the_geom GEOMETRY);
CREATE TABLE C ( ID INTEGER, the_geom GEOMETRY);
CREATE TABLE D ( ID INTEGER, the_geom GEOMETRY);
CREATE TABLE AIB (AID INTEGER, BID INTEGER, the_geom GEOMETRY);
CREATE TABLE CIAIB ( AID INTEGER, BID INTEGER,
                      CID INTEGER, the_geom GEOMETRY);
CREATE TABLE DIAIB ( AID INTEGER, BID INTEGER,
                      DID INTEGER, the_geom GEOMETRY);
```

Insertions des tuples :

```

INSERT INTO A VALUES
  (10, ST_GeometryFromText ('POLYGON ((1 1, 1 7, 3 7, 3 1, 1 1))',-1));
INSERT INTO B VALUES
  (20, ST_GeometryFromText ('POLYGON ((1 2, 1 8, 3 8, 3 2, 1 2))',-1));
INSERT INTO C VALUES
  (30, ST_GeometryFromText ('POLYGON ((2 5, 2 6, 4 6, 4 5, 2 5))',-1));
INSERT INTO D VALUES
  (40, ST_GeometryFromText ('POLYGON ((2 3, 2 4, 4 4, 4 3, 2 3))',-1));

INSERT INTO A VALUES
  (11, ST_GeometryFromText ('POLYGON ((5 1, 5 7, 7 7, 7 1, 5 1))',-1));
INSERT INTO B VALUES
  (21, ST_GeometryFromText ('POLYGON ((5 2, 5 8, 7 8, 7 2, 5 2))',-1));
INSERT INTO C VALUES
  (31, ST_GeometryFromText ('POLYGON ((6 5, 6 6, 8 6, 8 5, 6 5))',-1));
-- pas de D

INSERT INTO A VALUES
  (12, ST_GeometryFromText ('POLYGON ((9 1, 9 7, 11 7, 11 1, 9 1))',-1));
INSERT INTO B VALUES
  (22, ST_GeometryFromText ('POLYGON ((9 2, 9 8, 11 8, 11 2, 9 2))',-1));
-- pas de C
INSERT INTO D VALUES
  (42, ST_GeometryFromText ('POLYGON ((10 3, 10 4, 12 4, 12 3, 10 3))',-1));

```

Résultats attendus :

a	b	interab
10	20	POLYGON((1 2,1 7,3 7,3 2,1 2))
11	21	POLYGON((5 2,5 7,7 7,7 2,5 2))
12	22	POLYGON((9 2,9 7,11 7,11 2,9 2))
(3 rows)		

a	b	c	intercinterab
10	20	30	POLYGON((2 5,2 6,3 6,3 5,2 5))
11	21	31	POLYGON((6 5,6 6,7 6,7 5,6 5))
(2 rows)			

a	b	d	interdinterab
10	20	40	POLYGON((2 3,2 4,3 4,3 3,2 3))
12	22	42	POLYGON((10 3,10 4,11 4,11 3,10 3))
(2 rows)			