

# geom

## Description

Géométrie de la localisation

- Code : **geom**
- Libellé : **Géométrie de la localisation**
- Objectif du partage : Ce champ permet d'avoir une précision géographique plus importante que celle apporté par le rattachement concernant la localisation du relevé.
- Format : geometrie
- Exemple(s) : 01010000206A080000664BC8877E402D417A855A7B0F705841
- Lien avec le SINP : -

## Format standard

FSD	Obligation de renseignement
DATA	Obligatoire
TAXA	-

## Discussion

Nouveau champ non validé en GT

## Vocabulaire contrôlé

*pas de vocabulaire contrôlé*

## Spécificité

Le champ geom est une géométrie Postgis. C'est un type de champ particulier qui ne fonctionne pas spécifiquement avec un XY car il a implicitement un système de coordonnées (ou géodésique tout du moins) et un format spécifique.

## Import de XY

Il est possible de renseigner ce champ à partir de point X et Y et d'un système de projection. Pour se faire, les éléments suivants peuvent être réalisés :

1. Construire une table temporaire pour y importer les X et Y

```
CREATE TABLE public.releve_xy
(
  cd_jdd character varying NOT NULL,
  cd_releve character varying NOT NULL,
  cd_releve_perm character varying,
  coor_x float NOT NULL,
  coor_y float NOT NULL,
  geom geometry,
  cd_obj_geo character varying,
  CONSTRAINT releve_localisation_pkey PRIMARY KEY (cd_jdd, cd_releve),
  CONSTRAINT geom_srid CHECK (st_srid(geom) = 2154)
);
```

2. Importer les relevé et XY dans cette table :

```
COPY public.releve_xy FROM 'C:/....' CSV HEADER ENCODING 'UTF8' DELIMITER
';';
--- Remplacer C:/.... par lchemin vers le fichier;
--- Attention aux option à la fin de cette ligne (HEADER = un en-tête est
présente par exemple)
```

ATTENTION : les X et Y ne doivent pas utiliser de virgule (,) mais des points (.) en tant que séparateur

3. la fonction ST\_MakePoint va construire la géométrie et la fonction ST\_SetSRID va définir le système de projection (2154 pour RGF93) :

```
UPDATE public.releve_xy SET geom = ST_SetSRID(ST_MakePoint(coor_x,
coor_y),2154);
```

4. Tu peux ensuite renseigner la table localisation à partir de cette table :

```
INSERT INTO libSchema.releve_localisation SELECT cd_jdd, cd_releve,
cd_releve_perm, geom, cd_obj_geo FROM public.releve_xy;
--- Remplacer libSchema par le nom de votre schema
```

## Import de données en WKT

Il est possible de renseigner ce champ à partir de données au format WKT (Well-known text - par exemple : POLYGON <sup>1)</sup>) et d'un système de projection. Pour se faire, les éléments suivants peuvent être réalisés :

1. Construire une table temporaire (attention au srid)

```
CREATE TABLE public.releve_wkt
(
  cd_jdd character varying NOT NULL,
  cd_releve character varying NOT NULL,
  cd_releve_perm character varying,
  coor_wkt character varying NOT NULL,
```

```
geom geometry,  
cd_obj_geo character varying,  
CONSTRAINT releve_wkt_pkey PRIMARY KEY (cd_jdd, cd_releve),  
CONSTRAINT geom_srid CHECK (st_srid(geom) = **3948**)  
);
```

2. Importer les données dans cette table :

```
COPY public.releve_wkt FROM 'C:/....' CSV HEADER ENCODING 'UTF8' DELIMITER  
' ';  
--- Remplacer C:/.... par le chemin vers le fichier;  
--- Attention aux option à la fin de cette ligne (HEADER = un en-tête est  
présente par exemple)
```

ATTENTION : les X et Y ne doivent pas utiliser de virgule (,) mais des points (.) en tant que séparateur

3. la fonction ST\_GeomFromText va construire la géométrie (attention au srid):

```
UPDATE public.releve_wkt SET geom = ST_GeomFromText(coor_wkt,**3948**);
```

4. Tu peux ensuite renseigner la table localisation à partir de cette table :

```
INSERT INTO libSchema.releve_localisation SELECT cd_jdd, cd_releve,  
cd_releve_perm, geom, cd_obj_geo FROM public.releve_wkt;  
--- Remplacer libSchema par le nom de votre schema
```

1)

30 10, 40 40, 20 40, 10 20, 30 10

From:  
<https://wiki.fcbn.fr/> - Wiki - FCBN

Permanent link:  
<https://wiki.fcbn.fr/doku.php?id=ddd:geom&rev=1475162045>

Last update: **2022/03/07 12:00**

