



**RÉPUBLIQUE
FRANÇAISE**

*Liberté
Égalité
Fraternité*



Documentation des Données foncières

Extrait de la documentation en ligne



DOCUMENTATION



**MINISTÈRE
DE LA TRANSITION
ÉCOLOGIQUE
ET DE LA COHÉSION
DES TERRITOIRES**

*Liberté
Égalité
Fraternité*



**RÉPUBLIQUE
FRANÇAISE**

*Liberté
Égalité
Fraternité*



Le Cerema est un établissement public sous la tutelle du ministère de la Transition écologique, présent partout en métropole et dans les Outre-mer grâce à ses 26 implantations et ses 2 400 agents. Détenteur d'une expertise nationale mutualisée, le Cerema accompagne l'État et les collectivités territoriales pour la transition écologique, l'adaptation au changement climatique et la cohésion des territoires par l'élaboration coopérative, le déploiement et l'évaluation de politiques publiques d'aménagement et de transport. Doté d'un fort potentiel d'innovation et de recherche incarné notamment par son institut Carnot Clim'adapt, le Cerema agit dans 6 domaines d'activités : Expertise & ingénierie territoriale, Bâtiment, Mobilités, Infrastructures de transport, Environnement & Risques, Mer & Littoral.

Site web : www.cerema.fr

Avertissement

Ce document est extrait de la documentation en ligne, disponible sur le site <https://doc-datafoncier.cerema.fr/>, qui est mise à jour en continu. Ce document n'est donc valable qu'à un instant t, et le lecteur est invité à aller consulter la documentation en ligne pour bénéficier des dernières nouveautés.

Ce guide est un guide technique renseignant sur le contenu, les limites et la manière d'utiliser la base de données « Fichiers fonciers ». Pour d'autres questions (modalités d'accès, études réalisées...), le lecteur est invité à consulter le site <https://datafoncier.cerema.fr>

Cette documentation, évolutive, peut cependant encore contenir des erreurs ou coquilles. Si vous relevez une erreur dans celle-ci, n'hésitez pas à nous en faire part par mail via la boîte datafoncier@cerema.fr

Fiche

Table des matières

1 Cas pratiques pour dénombrer

1.1 Cas pratique : Dénombrer les mutations dans PostgreSQL/PostGIS

1.1.1 Dénombrer sur un département

1.1.2 Dénombrer sur une ou plusieurs communes

1.2 Cas pratique : Dénombrer les locaux dans PostgreSQL/PostGIS

1.2.1 Dénombrer sur un département

1.2.2 Dénombrer sur une commune

1.2.3 Dénombrer selon la forme physique du local

1.2.4 Dénombrer les appartements de moins de 3 pièces principales

1.2.5 Dénombrer les appartements T3 anciens

1.2.6 Dénombrer les mutations d'un local déterminé

1.3 Cas pratique : Dénombrer les parcelles dans PostgreSQL/PostGIS

1.3.1 Dénombrer sur un département

1.3.2 Dénombrer sur une commune

1.3.3 Dénombrer les mutations qui concernent une parcelle déterminée

1 Cas pratiques pour dénombrer

1.1 Cas pratique : Dénombrer les mutations dans PostgreSQL/PostGIS

1.1.1 Dénombrer sur un département

Pour obtenir le nombre de mutations qui ont eu lieu sur le département du Nord (59) :

```
SELECT count(*) as nombre_de_mutations
FROM dvf.mutation
WHERE coddep = '59';
```

1.1.2 Dénombrer sur une ou plusieurs communes

Pour obtenir le nombre de mutations qui concernent la commune de Lille (code insee : 59350) :

```
SELECT count(*) as nombre_de_mutations
FROM dvf.mutation
-- il est préférable de préciser le coddep pour réduire le temps de réponse
WHERE coddep = '59' AND '59350' = ANY(I_codinsee);
```

Pour obtenir le nombre de mutations qui concernent exclusivement la commune de Lille (code insee : 59350) :

```
SELECT count(*) as nombre_de_mutations
FROM dvf.mutation
-- il est préférable de préciser le coddep pour réduire le temps de réponse
WHERE coddep = '59' AND nbcomm = 1 AND I_codinsee[1] = '59350';
```

Pour obtenir le nombre de mutations qui concernent 2 communes (Lille - code insee : 59350 et Villeneuve d'Ascq - code insee : 59009) :

```
SELECT count(*) as nombre_de_mutations
FROM dvf.mutation
-- il est préférable de préciser le coddep pour réduire le temps de réponse
WHERE coddep = '59' AND I_codinsee && ARRAY['59350', '59009']::VARCHAR[];
```

1.2 Cas pratique : Dénombrer les locaux dans PostgreSQL/PostGIS

1.2.1 Dénumbrer sur un département

Pour obtenir le nombre de locaux ayant muté sur le département du Nord (59) :

```
-- via la table mutation
SELECT sum(nblocmut) as nombre_de_locaux
FROM dvf.mutation
WHERE coddep = '59';
```

```
-- via la table local (mode avancé)
SELECT count(*) as nombre_de_locaux
FROM
(
  SELECT DISTINCT idmutation, idloc
  FROM dvf.local
  WHERE coddep = '59'
)t;
```

1.2.2 Dénumbrer sur une commune

Pour obtenir le nombre de locaux mutés dont les mutations concernent exclusivement la commune de Lille (code insee : 59350) :

```
SELECT sum(nblocmut) as nombre_de_locaux
FROM dvf.mutation
-- il est préférable de préciser le coddep pour réduire le temps de réponse
WHERE coddep = '59' AND nbcomm = 1 AND I_codinsee[1] = '59350';
```

Dans le cas précédent, les mutations concernant 2 ou plusieurs communes dont Lille sont exclues. Pour obtenir le nombre de locaux ayant muté sur la commune de Lille (code insee : 59350) sans prendre en compte les locaux des autres communes :

```
SELECT count(*)
FROM
(
  SELECT DISTINCT idmutation, idloc
  FROM dvf.local
  WHERE substring(idpar, 1, 5) = '59350'
)t;
```

1.2.3 Dénumbrer selon la forme physique du local

Pour obtenir la répartition des locaux ayant muté selon leur forme physique sur le département du Nord (59) :

```
-- via la table mutation
SELECT
  sum(nblocmut) as nombre_de_locaux,
```

```

sum(nblocmai) as nombre_de_maisons,
sum(nblocapt) as nombre_d_appartements,
sum(nblocmai + nblocapt) as nombre_de_logements,
sum(nblocdep) as nombre_de_dependance,
sum(nblocact) as nombre_de_locaux_activites,
sum(ffnbacksec) as nombre_d_activites_secondaires,
sum(ffnbackter) as nombre_d_activites_tertiaires
FROM dvf.mutation
WHERE coddep = '59';

```

-- via la table local (mode avancé)

```

SELECT
count(*) as nombre_de_locaux,
sum(CASE WHEN codtyploc = 1 THEN 1 ELSE 0 END) AS nombre_de_maisons,
sum(CASE WHEN codtyploc = 2 THEN 1 ELSE 0 END) as nombre_d_appartements,
sum(CASE WHEN codtyploc IN (1, 2) THEN 1 ELSE 0 END) as nombre_de_logements,
sum(CASE WHEN codtyploc = 3 THEN 1 ELSE 0 END) as nombre_de_dependance,
sum(CASE WHEN codtyploc = 4 THEN 1 ELSE 0 END) as nombre_de_locaux_activites,
sum(CASE WHEN ffctyploc = '4' AND ffcnatloc IN ('U', 'US', 'UN', 'UE', 'UG') THEN 1 ELSE 0 END) as
nombre_d_activites_secondaires,
sum(CASE WHEN ffctyploc = '4' AND ffcnatloc IN ('CA', 'CM', 'CB', 'CH', 'ME', 'SM') THEN 1 ELSE 0 END) as
nombre_d_activites_tertiaires
FROM
(
SELECT DISTINCT ON (idmutation, idloc) *
FROM dvf.local
WHERE coddep = '59'
)t;

```

1.2.4 Dénombrer les appartements de moins de 3 pièces principales

Pour obtenir le nombre d'appartements de moins de 3 pièces principales (strictement) ayant muté sur le département du Nord (59) :

-- via la table mutation

```

SELECT sum(nbapt1pp + nbapt2pp) as nombre_appartement_moins_de_3_pieces
FROM dvf.mutation
WHERE coddep = '59';

```

-- via la table local

```

SELECT
count(*) AS nombre_appartement_moins_de_3_pieces
FROM
(
SELECT DISTINCT ON (idmutation, idloc) *
FROM dvf.local
WHERE coddep = '59'
)t
WHERE codtyploc = 2 AND nbpprinc IN (0, 1, 2);

```

1.2.5 Dénombrer les appartements T3 anciens

Pour obtenir le nombre d'appartements anciens de type T3 ayant muté sur le département du Nord (59) :

```
SELECT
  count(*) AS nombre_T3_ancien
FROM
  (
    SELECT DISTINCT ON (idmutation, idloc) *
    FROM dvf.local
    WHERE coddep = '59'
  )t
WHERE codtyploc = 2 AND nbpprinc = 3 AND anciennete = 'ancien';
```

1.2.6 Dénombrer les mutations d'un local déterminé

Pour lister toutes les mutations qui concernent le local ayant l'identifiant 595400649993 :

```
SELECT *
FROM dvf.mutation
WHERE coddep='59' AND '595400649993' = ANY(I_idlocmut);
```

Pour compter toutes les mutations qui concernent le local ayant l'identifiant 595400649993 :

```
SELECT count(*)
FROM dvf.mutation
WHERE coddep='59' AND '595400649993' = ANY(I_idlocmut);
```

1.3 Cas pratique : Dénombrer les parcelles dans PostgreSQL/PostGIS

1.3.1 Dénombrer sur un département

Pour obtenir le nombre de parcelles ayant muté sur le département du Nord (59) :

```
-- via la table mutation
SELECT sum(nbparmut) as nombre_de_parcelles_mutees
FROM dvf.mutation
WHERE coddep = '59';
```

```
-- via la table disposition_parcelle (mode avancé)
SELECT count(*) as nombre_de_parcelles_mutees
FROM
```



```
(
SELECT DISTINCT idmutation, idpar
FROM dvf.disposition_parcelle
WHERE coddep = '59' AND parcvendue IS TRUE
);
```

Pour obtenir le nombre de parcelles concernées par une mutation sur le département du Nord (59) :

```
-- via la table mutation
SELECT sum(nbpar) as nombre_de_parcelles_concernees
FROM dvf.mutation
WHERE coddep = '59';
```

```
-- via la table disposition_parcelle (mode avancé)
SELECT count(*) as nombre_de_parcelles_concernees
FROM
(
SELECT DISTINCT idmutation, idpar
FROM dvf.disposition_parcelle
WHERE coddep = '59'
);
```

1.3.2 Dénombrer sur une commune

Pour obtenir le nombre de parcelles mutées dont les mutations concernent exclusivement la commune de Lille (code insee : 59350) :

```
SELECT sum(nbparmut) AS nombre_de_parcelles_mutees
FROM dvf.mutation
-- il est préférable de préciser le coddep pour réduire le temps de réponse
WHERE coddep = '59' AND nbcomm = 1 AND I_codinsee[1] = '59350';
```

Pour obtenir le nombre de parcelles mutées sur la commune de Lille (code insee : 59350) :

```
SELECT count(*) AS nombre_de_parcelles_mutees
FROM
(
SELECT DISTINCT idmutation, idpar
FROM dvf.disposition_parcelle
WHERE coddep || codcomm = '59350' AND parcvendue IS TRUE
);
```

1.3.3 Dénombrer les mutations qui concernent une parcelle déterminée

Pour lister toutes les mutations qui concernent la parcelle ayant l'identifiant 59183540AL0328 :

```
SELECT *
```

```
FROM dvf.mutation  
WHERE coddep='59' AND '59183540AL0328' = ANY(I_idpar);
```

Pour compter toutes les mutations qui concernent la parcelle ayant l'identifiant 59183540AL0328 :

```
SELECT count(*)  
FROM dvf.mutation  
WHERE coddep='59' AND '59183540AL0328' = ANY(I_idpar);
```