RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

Cerema
CLIMAT & TERRITOIRES DE DEMAIN

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

Cerema
CLIMAT & TERRITOIRES DE DEMAIN

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 boite datafoncier@cerema.fr

Fiche

Table des matières

1 Format SQL de DV3F

1.1 ❓Version SQL de DV3F

1.2 ❓ Type d'utilisation

1.3 ❓ Avantages

1.4 ❓ Limites

1.5 ❓ Comment récupérer les données ?

1.6 ❓ Description des fichiers obtenus

1.7 ❓ Comment utiliser les données ?

1.7.1 ❓ Étape 1 – Décompresser l'archive

1.7.2 ❓ Étape 2 – Restaurer les fichiers SQL sous PostgreSQL

1.7.3 Définition du SRID des colonnes géométriques

1.7.4 ❓ Étape 3 – Interroger les données

1.8 ℹ Besoin d'aide ?

1 Format SQL de DV3F

1.1 ❓Version SQL de DV3F

Depuis sa première version, la base **DV3F** est proposée au **format SQL** pour PostgreSQL/PostGIS, afin de permettre une **restauration directe en base de données**.

> ⚠ Ce format restitue **l'ensemble de la structure relationnelle DV3F** (tables, clés, relations, index), avec une **restauration par département**. Depuis la version 2025-1, il **n'y a plus recours à l'héritage** dans le modèle (fonctionnalité peu maintenue sous PostgreSQL, compatibilité limitée).

1.2 ❓ Type d'utilisation

Ce format est particulièrement adapté à des usages avancés et structurés :

- Intégration dans une **base relationnelle** PostgreSQL/PostGIS
- Requêtes SQL complexes sur les transactions et leurs relations (locaux, parcelles,acheteurs/vendeurs, etc.)
- Analyses croisées exploitant **l'ensemble du modèle DV3F** et d'autres sources
- Intégration dans des chaînes de traitement ou plateformes de données

1.3 ❓ Avantages

- **Structure relationnelle complète** : toutes les tables DV3F sont disponibles
- **Cohérence garantie** : intégrité du modèle grâce aux clés et contraintes définies dans `dv3f_init.sql`
- **Performant** : possibilité d'interroger de gros volumes de données via SQL

1.4 ❓ Limites

- **Installation nécessaire** : requiert un serveur PostgreSQL (local ou distant) pour l'intégration

- **Volume important** : les fichiers départementaux peuvent être lourds à charger
- **Utilisateur averti** : des compétences minimales en SQL sont nécessaires pour interroger la donnée

1.5 ❓ Comment récupérer les données ?

- ❓ **Rendez-vous sur le Portail Données foncières**

  - ❓ *Nouveau sur le portail* ? Suivez le parcours guidé pour créer un compte "bénéficiaire"
  - ❓ *Déjà inscrit* ? Connectez-vous à votre espace

- ❓ **Faites une demande de téléchargement**

  - Cliquez sur le bouton **"Télécharger une donnée"**

- ❓ **Remplissez le formulaire**

  - Indiquez la donnée souhaitée **DV3F**
  - Précisez le **format SQL** et le **périmètre géographique**
  - Une **invitation Box.com** vous parviendra sous **24 à 48h**

- ❓ **Téléchargez l'archive**

  - Connectez-vous à l'espace sécurisé pour récupérer le fichier compressé (.7z)

1.6 ❓ Description des fichiers obtenus

Selon le volume de données demandé, vous recevrez plusieurs fichiers découpés (taille < 800 Mo) : `DV3F_AAAA_V_SQL_NNNN_EDAA1_part.7z.001` , `.002` , etc.

1.7 ❓ Comment utiliser les données ?

1.7.1 ❓ Étape 1 – Décompresser l'archive

Utilisez un outil de décompression adapté extraire les données `.7z` à partir d'outils libres comme *7-Zip* ou *PeaZip* → voir tutoriel 7-zip

⚠ Pour les fichiers découpés (`.7z.001`), commencez toujours la décompression à partir du **fichier numéro 1** en s'assurant que tous les fichiers numérotés sont présents dans le même dossier.

Une fois l'archive décompressée (voir Etape 1 ci-après), vous obtenez trois dossiers :

❓ `1_DONNEES_LIVRAISON` → contient les fichiers de données `sql`

❓ `4_METADONNEES_LIVRAISON` → contient le fichier de métadonnées et les documents juridiques (acte d'engagement, CGU)

❓ `5_SUPPLEMENTS_LIVRAISON` → généralement vide (réservé à des compléments éventuels)

1.7.2 ❓ Étape 2 – Restaurer les fichiers SQL sous PostgreSQL

L'archive contient :

- `dv3f_init.sql` → script d'initialisation (création du schéma et des tables, contraintes, index, etc.)
- `dv3f_dXX.sql` → un fichier par département (`XX = code département`), contenant les requêtes d'**insertions des données**
- `dv3f_annexe.sql` → script de restauration des tables annexes DV3F (dictionnaires, nomenclatures, typologies, etc.)

> ⚠ Pour restaurer les fichiers SQL, il faut préalablement avoir accès à une instance PostgreSQL avec l'extension PostGIS.

1. Créez une base vide dans votre instance PostgreSQL si nécessaire:

```
CREATE DATABASE dv3f_db;
```

1. Activez l'extension Postgis dans votre base de donnée :

```
psql -d dv3f_db -c "CREATE EXTENSION postgis;"
```

1. Exécutez le script d'initialisation :

```
psql -d dv3f_db -f dv3f_init.sql
```

1. Chargez les données départementales (répétez pour chaque fichier `dv3f_dXX.sql`) :

```
psql -d dv3f_db -f dv3f_d75.sql
psql -d dv3f_db -f dv3f_d13.sql
...
```

1. Chargez les tables annexes :

```
psql -d dv3f_db -f dv3f_annexe.sql
```

1.7.3 Définition du SRID des colonnes géométriques

A partir du millésime 2026-1 de DV3F, les tables auparavant départementales ont été agrégées à l'échelle nationale.

Cette agrégation couvre la France hexagonale ainsi que les départements d'outre-mer.

En conséquence, les colonnes géométriques ne sont plus typées.

Vous trouverez ci-dessous une série de requêtes permettant de rétablir le typage des géométries.

L'exemple est donné pour la France hexagonale avec le SRID 2154 ; celui-ci devra être adapté dans le cas d'un département d'outre-mer.

Table mutation

```
ALTER TABLE dv3f_2026_1.mutation ALTER COLUMN geompar TYPE geometry(MultiPolygon, 2154) USING ST_SetSRID(ST_Multi(geompar), 2154);
ALTER TABLE dv3f_2026_1.mutation ALTER COLUMN geomparmut TYPE geometry(MultiPolygon, 2154) USING ST_SetSRID(ST_Multi(geomparmut), 2154);
ALTER TABLE dv3f_2026_1.mutation ALTER COLUMN geomlocmut TYPE geometry(MultiPoint, 2154) USING ST_SetSRID(ST_Multi(geomlocmut), 2154);
```

Table disposition_parcelle

```
ALTER TABLE dv3f_2026_1.disposition_parcelle ALTER COLUMN geompar TYPE geometry(MultiPolygon, 2154) USING ST_SetSRID(ST_Multi(geompar), 2154);
ALTER TABLE dv3f_2026_1.disposition_parcelle ALTER COLUMN geomloc TYPE geometry(MultiPoint, 2154) USING ST_SetSRID(ST_Multi(geomloc), 2154);
```

Table local

```
ALTER TABLE dv3f_2026_1.local ALTER COLUMN geomloc TYPE geometry(MultiPoint, 2154) USING ST_SetSRID(ST_Multi(geomloc), 2154);
```

1.7.4 ❓ Étape 3 – Interroger les données

Vous pouvez ensuite exécuter vos requêtes SQL, par exemple :

```
-- Nombre de mutations par année
SELECT anneemut, COUNT(*)
FROM dv3f_2025_1.mutation
GROUP BY anneemut
ORDER BY anneemut;
```

1.8 ℹ Besoin d'aide ?

Pour toute difficulté, vous pouvez contacter l'équipe Datafoncier