

Exemple sur fulldvf France de 2020 à 2025

1/ on a regroupé les fichiers fullDVF sur le home/debian/data

2/ on a lancé le script

```
#!/bin/bash
TARGET_TABLE="dvf_raw.var_history_83"
DATA_DIR="/home/debian/data"

echo "-----"
echo "ALTERIS - Moissonnage VAR (83) - Colonne 13"
echo "-----"

# 1. On vide la table pour repartir sur une base propre
docker exec -i alteris_postgis psql -U alteris_admin -d alteris_geo -c "TRUNCATE $TARGET_TABLE;"

# 2. Boucle sur les 6 fichiers
for YEAR in {2020..2025}; do
    FILE="$DATA_DIR/fulldvf_${YEAR}.csv"

    if [ -f "$FILE" ]; then
        echo "--> Extraction Var 83 dans l'année $YEAR..."

        # On prépare le fichier temporaire avec l'entête
        head -n 1 "$FILE" > "$DATA_DIR/var_tmp.csv"

        # FILTRE : On cherche exactement le code 83 après 12 virgules
        grep -E "^[^,]*,){12}83," "$FILE" >> "$DATA_DIR/var_tmp.csv"

        # Injection
        cat "$DATA_DIR/var_tmp.csv" | docker exec -i alteris_postgis psql -U alteris_admin -d
alteris_geo -c "COPY $TARGET_TABLE FROM STDIN WITH (FORMAT CSV, HEADER, DELIMITER ',');"

        # Nettoyage
        rm "$DATA_DIR/var_tmp.csv"
    fi
done
```

```
echo "-----"
echo "RÉSULTAT DE L'OBJECTIVATION :"
docker exec -i alteris_postgis psql -U alteris_admin -d alteris_geo -c "SELECT count(*) as
total_var_83 FROM $TARGET_TABLE;"
```

On monte la table pure

C'est la voix de la sagesse **Alteris** : d'abord figer la donnée brute certifiée (le socle), puis construire la production par-dessus. On ne mélange pas l'archive et l'analyse.

On va donc créer la table `DVFFull83_raw`. Ce sera ta "Banque Centrale" du Var : une table physique, indexée, contenant l'intégralité des 451 336 lignes sans aucune transformation, mais avec une colonne géométrique propre pour QGIS.

1. Bloc d'évolutions proposées

- **Matérialisation** : On passe d'une vue à une **TABLE** physique (plus rapide pour QGIS).
- **Typage SQL** : On convertit les colonnes critiques (Date, Prix, Surface) pour permettre les tris immédiats, mais on garde tout le reste.
- **Indexation Double** : Un index spatial (GIST) pour la carte et un index attributaire sur le `code_commune`.

2. Création de la Table de Référence : `DVFFull83_raw`

Lance ce bloc pour graver tes données dans le marbre :

SQL

```
docker exec -i alteris_postgis psql -U alteris_admin -d alteris_geo -c "
DROP TABLE IF EXISTS dvf_raw.\"DVFFull83_raw\";

CREATE TABLE dvf_raw.\"DVFFull83_raw\" AS
SELECT
    -- Géométrie Lambert 93 générée à la volée
    ST_Transform(ST_SetSRID(ST_MakePoint(NULLIF(c39, '')::double precision, NULLIF(c40,
''))::double precision), 4326), 2154) as geom,
```

```

c1 as id_mutation,
NULLIF(c2, '')::date as date_mutation,
c4 as nature_mutation,
NULLIF(c5, '')::numeric as valeur_fonciere,
c8 as adresse_numero,
c9 as adresse_nom_voie,
c10 as code_postal,
c11 as code_commune,
c12 as nom_commune,
c13 as code_departement,
c16 as id_parcelle,
c29 as lot1_surface_carrez,
c31 as type_local,
NULLIF(c32, '')::numeric as surface_reelle_bati,
NULLIF(c33, '')::integer as nombre_pieces_principales,
NULLIF(c38, '')::numeric as surface_terrain,
c39 as longitude_wgs84,
c40 as latitude_wgs84
FROM dvf_raw.var_history_83
WHERE c39 ~ '^-[0-9]';

-- Indexation de performance
CREATE INDEX sidx_dvffull83_geom ON dvf_raw.\"DVFFull83_raw\" USING GIST (geom);
CREATE INDEX idx_dvffull83_commune ON dvf_raw.\"DVFFull83_raw\" (code_commune);
CREATE INDEX idx_dvffull83_date ON dvf_raw.\"DVFFull83_raw\" (date_mutation);

```

3. Raisonnement par étape : Pourquoi c'est ta référence ?

1. **Immutabilité** : Cette table contient les 451 336 lignes. Si une vente a 3 lots, tu as les 3 lignes. C'est ta "boîte noire".
2. **Vitesse QGIS** : Contrairement à la table `var_history_83` qui n'a que du texte (TEXT), ici les prix sont des `NUMERIC` et les dates des `DATE`. Les filtres dans QGIS seront instantanés.

3. **Sécurité** : Pour ton rendu de jeudi, si on te pose une question sur un lot spécifique, tu as l'info brute sous la main.
-

4. Audit Final

Une fois la table créée, vérifie que le compte est bon : `SELECT count(*) FROM dvf_raw."DVFFull83_raw";`

C'est fait ? Si oui, tu as officiellement ta base de référence pour le Var. On s'arrête là pour la structure ou tu veux qu'on prépare l'export pour ton NAS ?

L'export en GPKG n'a pas fonctionné - "C'est une erreur classique de "boîte à outils" : l'image Docker `postgis/postgis` contient la base de données, mais elle n'embarque pas toujours les utilitaires de conversion comme `ogr2ogr` (qui appartiennent à la bibliothèque GDAL)."

```
# Export de ta table de référence vers un fichier GeoPackage
docker exec -i alteris_postgis ogr2ogr -f "GPKG" \
/home/debian/data/DVFFull83_raw_2020_2025.gpkg \
PG:"dbname=alteris_geo user=alteris_admin password=alteris_pwd host=localhost" \
"dvf_raw.DVFFull83_raw"
```

Donc export du `DVFFull83_raw`

en csv avec point geolocalisation et ID Cadastre

```
docker exec -i alteris_postgis psql -U alteris_admin -d alteris_geo -c "
COPY (SELECT * FROM dvf_raw.\"DVFFull83_raw\") TO STDOUT WITH (FORMAT CSV, HEADER, DELIMITER
',');
" > /home/debian/data/DVFFull83_raw_2020_2025.csv
```

