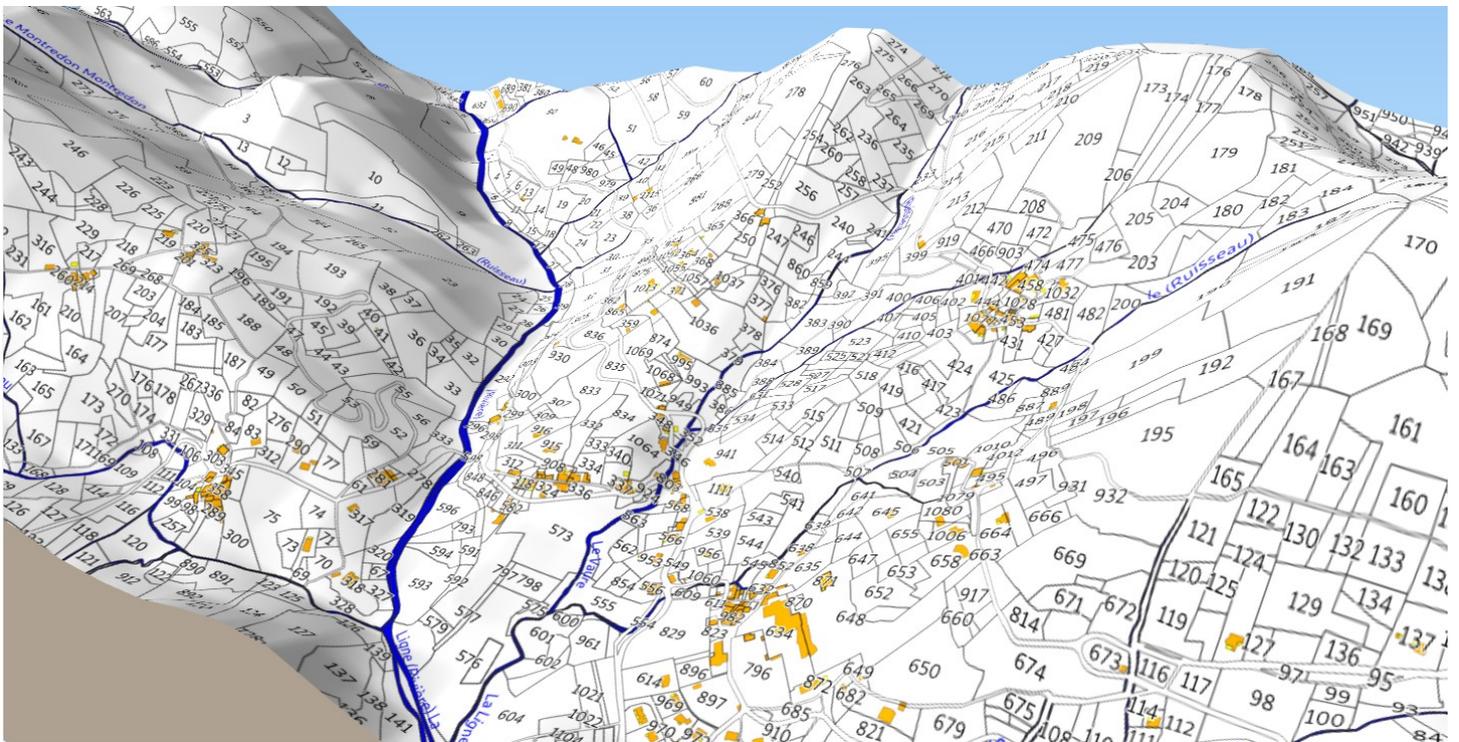


**Documentation  
projet ADULLACT edigeo  
Version 0.90**

**Mai 2020**



*Plan cadastral 3D en Ardèche*

contact : Jean-Claude CATY ([harnel.rasalhague@nospam@laposte.net](mailto:harnel.rasalhague@nospam@laposte.net))

# Table des matières

<b>Introduction.....</b>	<b>3</b>
I Système.....	3
II Données.....	3
III Exploitation.....	3
<b>Mise en place (le coin du développeur).....</b>	<b>4</b>
I Dossier git.....	4
II Dépendances.....	4
II.1 Installation des briques.....	4
II.2 Installation cmake.....	4
II.3 Installation postgresql.....	4
II.4 Installation de postgis.....	5
II.5 Installation de doxygen.....	5
III Compilation.....	5
III.1 Construction avec le logiciel cmake.....	5
III.2 Installation avec le logiciel cmake.....	5
<b>Installation.....</b>	<b>6</b>
I Dossier des données.....	6
II Installation des scripts bash.....	6
III Ajout des données des scripts.....	6
<b>Données ouvertes.....</b>	<b>8</b>
I Téléchargement des données.....	8
I.1 Données EDIGÉO.....	8
I.2 Données altimétriques de l'IGN.....	8
<b>Utilisation.....</b>	<b>10</b>
I Production EDIGÉO vers sql.....	10
I.1 Le programme edigeo2sql.....	10
<i>A Paramètres du programme.....</i>	<i>10</i>
I.2 Le script repartiteurLecture.bash.....	11
I.3 Le script lectureEdigeo.bash.....	11
II Chargement en base de données.....	12
II.1 Le script repartiteurSQL.bash.....	12
II.2 Le script complementsCadastre.sql.....	12

# INTRODUCTION

## *I Système*

Le projet edigeo est développé sous Linux. La version utilisée est Debian. Les données étant insérées dans une base de données cartographiques postgresql/postgis, il sera possible de les exploiter avec tout logiciel capable de s'interfacer avec le serveur de la base de données.

## *II Données*

Depuis quelques années, de multiples données géographiques sont ouvertes. Il en est ainsi des données cadastrales disponibles sous les formats *edigeo* et *dxf*. Le format *dxf* est plutôt un format de dessin aussi le projet utilise les données au format *edigeo*, format de données géographiques.

La norme *edigeo* est une norme française pour l'échange de données géographiques. Bien qu'elle ait été annulée en août 2017, le cadastre français exporte toujours ses informations dans ce format.

Le cadastre ne gère pas de données altimétrique. Il peut cependant être utile de combiner les données cadastrales avec des données altimétriques comme celles de la BDALTI de l'IGN. Le présent projet *edigeo* est donc capable d'affecter une altitude issue de la BDALTI à tout point planimétrique issu des données cadastrales. La BDALTI est délivrée dans le système de projection LAMBERT93. Le mariage « planimétrie – altimétrie » ne sera donc effectif qu'avec les données cadastrales exprimées dans le système LAMBERT93.

## *III Exploitation*

Le programme *edigeo* exporte les données au format SQL permettant d'être ensuite intégrées dans une base de données postgresql/postgis. Il sera donc possible d'utiliser le système d'information géographique libre QGIS pour toute gestion géographique. Le format *edigeo* indiquant la structure des données transmises (fichiers suffixés GEO, DIC et SCD), la passerelle permet de produire les instructions SQL de création d'une base dans le système de projection ad-hoc et dans un schéma « cadastre » de la base de données.

Par ailleurs, l'INSEE délivre également des informations sur les structures territoriales (Communes fusionnées, EPCI). Un script SQL permet d'exploiter ces éléments et, par requêtes géographiques, construit par fusion les objets géographiques correspondants (communes « périmées » par fusion des sections cadastrales, EPCI par fusion des communes) dans la base de données.

## MISE EN PLACE (le coin du développeur)

La mise en place du projet est constituée de plusieurs briques : Un dossier pour le code source, le compilateur C++, le logiciel cmake pour la construction du projet, l'installation de la librairie SFCGAL, l'installation du logiciel doxygen si vous souhaitez construire la documentation développeur à partir du code source et enfin le serveur de base de données postgresql et les extensions associées postgis et postgis\_sfcgal et enfin le logiciel QGIS pour l'exploitation des données dans un système d'information géographique.

### I Dossier git

Le code source du projet est géré avec le logiciel git. Vous devez donc installer le paquet git si ce n'est déjà fait. En ligne de commande, en mode administrateur (root) :

```
apt-get install git
```

Pour importer les fichiers du projet, vous devez tout d'abord créer votre espace de travail. Une solution est de créer un dossier *git* (ou autre) dans votre répertoire personnel. En ligne de commande, sous votre compte utilisateur :

```
cd
mkdir -p git
cd git
```

Si vous choisissez un dossier différent de `~/git`, vous devrez adapter les commandes qui suivent à votre choix.

Vous importez ensuite le projet edigeo :

```
git clone https://scm.adullact.net/anonscm/git/edigeo/edigeo.git
```

Un dossier `~/git/edigeo` est créé, contenant une copie des dossiers et fichiers nécessaires à la construction du projet.

### II Dépendances

La construction et l'utilisation du projet est dépendante de nombreuses briques logicielles. Il est donc nécessaire d'installer ces diverses briques pour une correcte exploitation.

#### II.1 Installation des briques

Commençons par installer l'ensemble de briques nécessaires. En mode administrateur (root) :

```
apt-get install cmake cmake-extras extra-cmake-modules libsfcgal-dev postgresql postgis
apt-get install doxygen graphviz
```

#### II.2 Installation cmake

cmake est un outil de construction de projet. Il est livré avec de nombreux modules. Ces modules sont situés dans le dossier `/usr/share/cmake/Modules`. Le présent projet nécessite la librairie libsfcgal, laquelle n'est pas référencée dans les modules. Vous pouvez l'ajouter aux modules présents en copiant le fichier <https://github.com/Oslandia/SFCGAL/blob/master/cmake/Modules/FindSFCGAL.cmake> dans le dossier `/usr/share/cmake-X.YY/Modules` où `X.YY` est la version cmake installée. Les droits administrateur (root) sont nécessaires pour cette copie :

```
cd /usr/share/cmake-3.13/Modules
wget https://github.com/Oslandia/SFCGAL/blob/master/cmake/Modules/FindSFCGAL.cmake
```

#### II.3 Installation postgresql

Après installation de la base de données postgresql, paramétrez un utilisateur correspondant à votre nom de connexion. Référez-vous au manuel postgresql pour paramétrer correctement votre base de données.

Exemple :

```
su
passwd postgres # renseignez ici un mot de passe pour l'utilisateur postgres
exit
```

```
su - postgres
createuser -dlrs jcc # Votre nom de connexion à la place de jcc
exit
test avec la commande psql -l
```

## II.4 Installation de postgis

postgis est l'extension cartographique de la base données postgresql.

## II.5 Installation de doxygen

Le logiciel doxygen permet de documenter à partir des commentaires du code source. Il utilise le logiciel *graphviz* pour générer divers schémas (graphes d'héritage, graphes de collaboration...). En vous positionnant dans le dossier `~/git/edigeo` (si vous avez pris l'option du § "Dossier git"), lancer les commandes :

```
cd ~/git/edigeo
doxygen Doxyfile
```

À l'heure de rédaction de cette documentation (mai 2020), la documentation de la librairie n'est pas terminée. La partie documentée est accessible avec un navigateur :

```
firefox ~/git/edigeo/libedigeo/doxygen/html/index.html
```

## III Compilation

### III.1 Construction avec le logiciel cmake

Afin de bien séparer les fichiers de code source et la construction avec cmake, il a été créé un dossier `~/git/edigeo/cmake` dans lequel aura lieu la construction. En contrepartie, il faudra lancer le script `~/git/edigeo/cmake/edigeo.bash` pour actualiser les sources à compiler. Ce script copie les fichiers nécessaires dans le dossier cmake. Ces fichiers auraient pu être liés (commande *ln*) mais des IDE comme codeblocks détruisent ces liens à l'enregistrement.

Pour construire la librairie libedigeo et le programme edigeo2sql, Placez-vous dans le dossier `~/git/edigeo/cmake` et lancez la construction :

```
cd ~/git/edigeo/cmake
./edigeo.bash
cmake .
make
```

### III.2 Installation avec le logiciel cmake

Pour installer la librairie *libedigeo.so*, ses fichiers "include" de développement et le programme *edigeo2sql* :

```
cd ~/git/edigeo/cmake
su - root
make install
```

# INSTALLATION

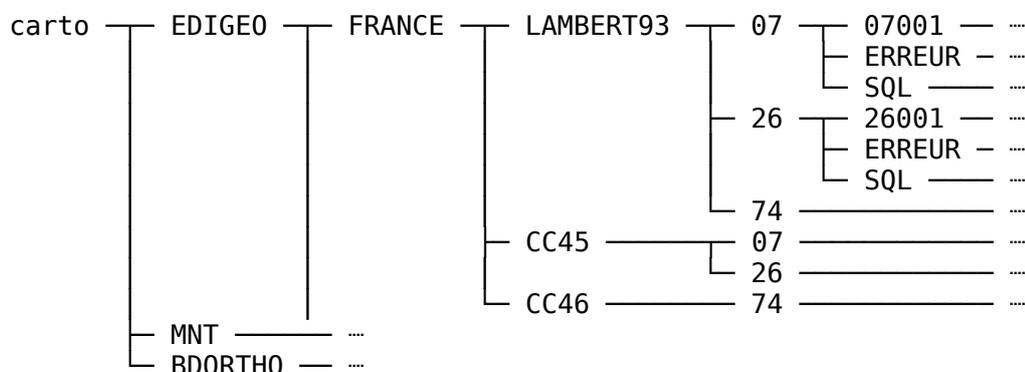
Le logiciel edigeo2sql traite un seul échange EDIGÉO et produit les fichiers sql et éventuellement erreur de cet échange. Pour traiter les milliers d'échanges constituant un département, des scripts sont bienvenus. Il en est de même pour le traitement des fichiers sql à insérer ensuite dans une base de données cartographiques. De plus, afin d'accélérer ces processus, il est utile de paralléliser ces différents processus.

C'est le rôle des scripts que nous allons maintenant installer à un endroit stratégique, proche des données.

## I Dossier des données

Comme pour le dossier `~/git/` mentionné au chapitre « Dossier git », nous allons préparer une arborescence pour les données. Sur mon ordinateur personnel, une partition est consacrée aux données cartographiques. Cette partition est montée dans le dossier `/home/media/carto/`.

Dans ce dossier `carto` se trouve un sous-dossier `EDIGEO` comportant lui-même des sous-dossiers `FRANCE/LAMBERT93/`, `FRANCE/CC45/` ... selon une arborescence ressemblant à :



Sous le dossier `FRANCE`, nous trouvons des dossiers correspondant aux différents systèmes de projection (`LAMBERT93`, `CC45`, `CC46`...). Sous chaque système de projection, nous trouvons les départements. D'autres dossiers seront créés au gré des traitements : `ERREUR` et `SQL`. Nous verrons plus tard comment placer les données téléchargées dans cette arborescence.

Selon vos besoins et les données que vous souhaitez traiter, vous construirez l'arborescence qui vous correspond.

## II Installation des scripts bash

Après avoir cloné le projet dans votre dossier `~/git/`, vous trouverez un dossier `~/git/edigeo/scripts`. Dans ce dossier `scripts`, figurent 4 scripts bash, un script sql et un fichier `"departementATraiter"`.

Les scripts bash et le script sql peuvent être copiés dans le dossier `FRANCE`.

## III Ajout des données des scripts

Le fichier `"departementATraiter"` peut être copié vers les dossiers des systèmes de projections que vous souhaitez traiter.

Il sera utilisé lors des insertions en base de données. Si vous préférez ne traiter que certains départements (sous-dossiers des projections) et pas tous, il vous suffit de « dénaturer » les dossiers que vous ne souhaitez pas voir traiter. Par exemple, si vous avez téléchargé les données des départements 07 et 26, que vous ne souhaitez traiter que le 07, vous dénaturer la ligne "26" de telle façon à ce qu'elle ne référence aucun sous-dossier de la projection. (en remplaçant la ligne dans laquelle figure "26" par "#26" : "26" référence un dossier mais pas "#26").

Le script sql “*complementsCadastre.sql*” complète les données cadastrales par des requêtes ajoutant les départements, les communes périmées (suite à fusion) et les EPCI. Les informations géométriques sont issues de requêtes géographiques, les informations littérales proviennent de fichiers de l’INSEE. Cependant, pour les communes périmées, toutes les communes n’ont pas été trouvées. Les libellés des anciennes communes de la table *cadastre.fusion* ne sont donc pas complets.

Si vous complétez ce fichier avec les anciennes communes de votre département, envoyez vos ajouts à mon adresse courriel spécifiée sur la première page de ce document.

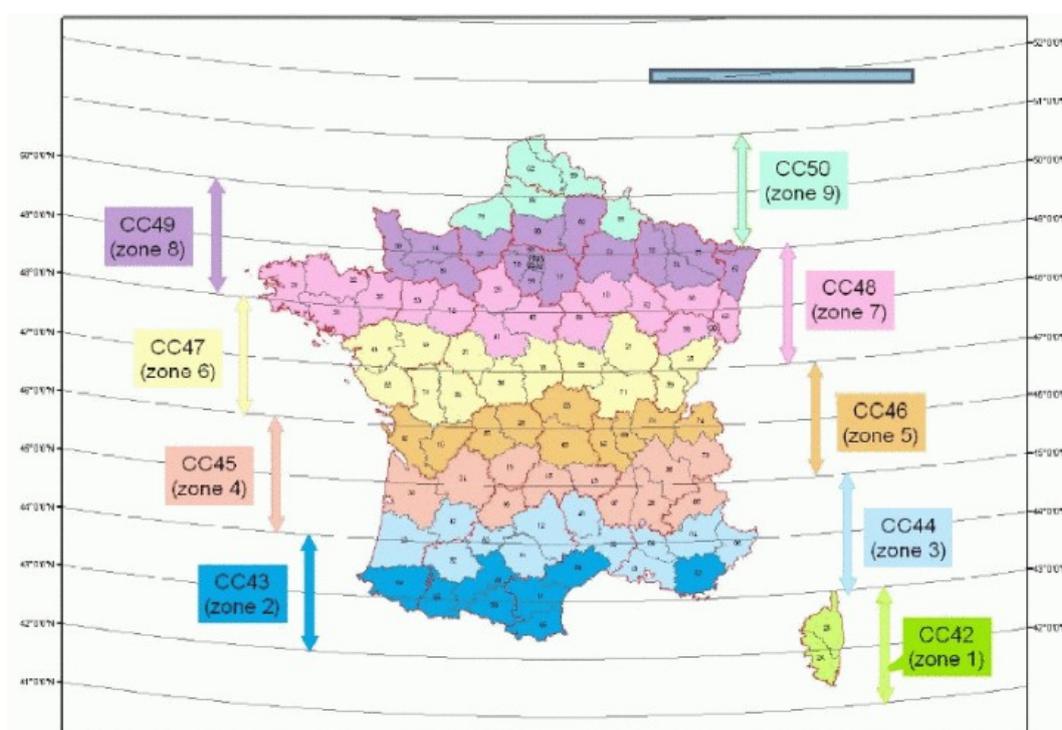
## I Téléchargement des données

### I.1 Données EDIGÉO

Le format EDIGÉO emporte avec lui des métadonnées. Notamment, le système de projection permettant de définir le code « EPSG »

Les données cadastrales au format EDIGEO sont disponibles sur internet à l'adresse <https://cadastre.data.gouv.fr/datasets/plan-cadastral-informatise>. En bas de cette page, vous trouverez 2 formats :

- Format EDIGÉO : Il s'agit des données cadastrales correspondant au système de projection LAMBERT 93 (Code EPSG 2154). Toute la France est disponible dans ce système de projection.
- Format EDIGÉO (CC 9 zones) : Ce format est également une projection LAMBERT. Pour minimiser les altérations linéaires, la France est découpée en 9 zones. Chaque département est entièrement compris dans une des 9 zones.



Projections « coniques conformes » en France métropolitaine

Pour plus d'informations concernant les systèmes de projection, voir sur wikipédia : [https://fr.wikipedia.org/wiki/Système\\_de\\_coordonnées\\_\(cartographie\)](https://fr.wikipedia.org/wiki/Système_de_coordonnées_(cartographie))

Afin de pouvoir exploiter les données EDIGÉO avec les scripts, il est conseillé de télécharger le département choisi dans le dossier correspondant au système de projection. Par exemple, Vous téléchargerez le département 07 dans le système CC45 (Lien <https://cadastre.data.gouv.fr/data/dgfip-pci-vecteur/latest/edigeo-cc/departements/dep07.zip>) dans le dossier FRANCE/CC45 (cf Dossier des données). Après décompression, vous renommez le dossier dep07 en 07 de façon à correspondre à l'arborescence présentée plus haut.

### I.2 Données altimétriques de l'IGN

Lors de la lecture des données EDIGÉO, le programme edigeo2sql recherche d'éventuelles données altimétriques issues de la BD-ALTI de l'IGN. Les coordonnées des données altimétriques étant exprimées dans le système LAMBERT93, cette fonctionnalité n'est valable que pour les échanges exprimés dans ce même

système.

Les données de la BD-ALTI sont disponibles à l'adresse suivante : <https://geoservices.ign.fr/documentation/diffusion/telechargement-donnees-libres.html#bd-alti>. Seuls les fichiers suffixés “.asc” peuvent être conservés pour la présente application et déplacés dans le dossier .../carto/MNT/.

Dans le fichier téléchargé (au format 7zip), les fichiers suffixés “.asc” se trouvent dans le sous-dossier /BDALTIV2\_2-0\_75M\_ASC\_LAMB93-IGN69\_FRANCE\_2018-01-15/BDALTIV2/1\_DONNEES\_LIVRAISON\_2018-01-00245/BDALTIV2\_MNT\_75M\_ASC\_LAMB93-IGN69\_FRANCE/.

Au démarrage, le programme edigeo2sql ne charge que les entêtes des fichiers “.asc” et détermine l'emprise correspondante. La suite du fichier ne sera effectivement lue qu'en cas de besoin pour estimer l'altitude à partir des coordonnées lues dans les échanges EDIGÉO.

## I Production EDIGÉO vers sql

Le programme edigeo2sql traite un seul échange. Si l'on souhaite traiter un département entier comprenant plusieurs milliers d'échanges (un échange par subdivision de section cadastrale), il est nécessaire d'encapsuler le traitement d'un échange dans un traitement plus global. Ce traitement global est l'apanage de 2 scripts bash au sein de la structure définie au paragraphe Dossier des données

Par ailleurs, afin de traiter plus rapidement ces échanges, le script « repartiteurLecture.bash » parcourt l'arborescence des données et lance des traitements en parallèle en fonction du nombre de cœurs du / ou des processeurs présents sur la machine. Ces traitements individuels sont réalisés via le script « lectureEdigeo.bash » Ce script va traiter l'ensemble des échanges EDIGÉO d'une commune.

### I.1 Le programme edigeo2sql

#### A Paramètres du programme

Un échange EDIGÉO est lu en précisant comment réaliser cette lecture. Les scripts *bash* joints adapteront ces paramètres au contexte. Voici la liste des paramètres, sous forme d'option sur la ligne de commande :

- Génération des métadonnées correspondant à la structure EDIGÉO (option “**-se <fichier sql>**”)

Cette option permet de générer un fichier sql pour préparer la base à recevoir les données dans le même format qu'EDIGÉO sans retraitement. Chaque table représente un descripteur.

L'option **-se** doit être suivie du nom du fichier sql qui reçoit le résultat.

- Génération des métadonnées correspondant à la structure du SCD (option “**-so <fichier sql>**”)

Un échange EDIGÉO définit la structure des données transmises. Ces informations sont une combinaison des fichiers GEO, DIC et SCD. Elles permettent de créer les tables d'accueil des objets transmis. Par exemple, la table des parcelles pour un échange cadastral.

L'option **-so** doit être suivie du nom du fichier sql qui reçoit le résultat.

- Traduction EDIGÉO vers SQL (Option “**-e <dossier des fichiers sql>**”)

Cette option est simplement une traduction EDIGÉO vers SQL. Les données EDIGÉO sont inscrites dans les tables définies par l'option **-se**.

L'option **-e** doit être suivie du dossier où sont stockés les fichiers sql résultat.

- Traduction EDIGÉO vers données SIG (Option “**-o <dossier des fichiers sql>**”)

génère les requêtes de création des objets correspondant à la structuration définie dans l'échange. Les données sont inscrites dans les tables définies par l'option **-so**

L'option **-o** doit être suivie du dossier où sont stockés les fichiers sql résultat.

- Ajout d'un préfixe attribué aux informations de l'échange (Option “**-p <préfixe>**”)

Ce préfixe caractérise un échange et permet d'obtenir un identifiant global pour les échanges cadastraux. Il complète les identifiants des descripteurs EDIGÉO. Pour les échanges cadastraux, ce préfixe est formé des codes département, code insee des communes, préfixe de section (ancien code insee de communes absorbées), section et subdivision de section.

L'option **-p** doit être suivie du préfixe associé à l'échange.

- pré-traitement cadastre (Option “**-c**”)

Ce prétraitement permet de réaliser des traitements comme la fusion `TEX_id` de certains objets (rivières, lieudits ...).

- Exclusion d'objets (Option "`-n <Type(s)>`")
- Permet de ne pas insérer le même objet plusieurs fois. Par exemple, les échanges EDIGÉO cadastraux correspondent chacun à une subdivision de section cadastrale. Tous les échanges d'une subdivision contiennent la section. Pour le premier échange traité, la section est considérée. Pour les échanges suivants dans la même section, la section est ignorée.

Nous avons le même problème avec l'ensemble des échanges d'une même commune.

L'option `-n` doit être suivie des objets à éviter (ex : `-n COMMUNE,SECTION`)

- Calcul d'altitudes approchées pour chaque coordonnée EDIGÉO (Option "`-m <dossier MNT>`")

Si vous disposez de MNT (Modèles Numériques de Terrain) au format *asc*, précisez le dossier où trouver ces MNT avec cette option. Une altitude pondérée approchée est calculée. Ultérieurement, une information de qualité viendra accompagner cette altitude (type *descripteur QAA* dans EDIGÉO).

## I.2 Le script repartiteurLecture.bash

Ce script sert principalement à répartir la lecture des échanges d'un ou plusieurs départements en processus parallèles en fonction du nombre de cœurs dans les processeurs de l'ordinateur hôte. Le script lit le nombre de cœurs (*c*), le multiplie par 1,5 et lance au maximum de  $1,5 * c$  processus de lecture de communes en parallèle. Dès que le nombre de processus est atteint, des temporisations successives permettent d'attendre la fin de processus de lecture pour pouvoir en lancer d'autres.

Avant de lancer la lecture effective des échanges de départements d'une projection, le script génère 2 fichiers SQL permettant de construire la base de données sous postgis. Ces fichiers sont générés via les options « `-se` » et « `-so` » du programme *edigeo2sql*. Ces 2 fichiers sont déposés dans le dossier de la projection et prennent en compte le code EPSG de la projection.

Ce script prend un paramètre facultatif. Il s'agit d'un nom de sous-dossier correspondant à un système de projection. Par exemple LAMBERT93, CC45, CC46 ... Par défaut, le paramètre sera LAMBERT93.

Exemple de lancement (scripts placés dans le dossier `/home/media/carto/EDIGEO/FRANCE`) :

```
cd /home/media/carto/EDIGEO/FRANCE
./repartiteurLecture.bash CC45
```

Ces commandes traiteront les départements du dossier `/home/media/carto/EDIGEO/FRANCE/CC45` listés dans le fichier `departementATraiter` situé dans le dossier de la projection.

Les fichiers SQL produits seront stockés sous `/home/media/carto/EDIGEO/FRANCE/CC45/SQL`

Les éventuelles erreurs de géométrie détectées seront stockées sous `/home/media/carto/EDIGEO/FRANCE/CC45/ERREUR`.

## I.3 Le script lectureEdigeo.bash

Le script « *lectureEdigeo.bash* » est lancé en arrière plan par le répartiteur « *repartiteurLecture.bash* ». Il permet de traiter l'ensemble des échanges EDIGÉO d'une commune. Plusieurs processus de lecture sont donc traités en parallèle. Il nécessite 3 paramètres :

- Le dossier « Projection ». Exemple : `/home/media/carto/EDIGEO/FRANCE/LAMBERT93`,
- Le sous-dossier département. Exemple : `07`,
- Le dossier de la commune. Exemple : `07001`.

Ce script prend en charge les objets à éviter de façon à ne pas multiplier les communes et sections. Ainsi, tous les échanges de la commune contiennent le même objet commune. Seul le premier échange traité pour cette commune générera l'objet « *commune* » dans les résultats SQL. Il en est de même pour les sections cadastrales.

## II Chargement en base de données

Comme pour la lecture des échanges EDIGEO, le chargement en base de données des fichiers SQL précédemment produit est géré par 2 scripts. Le premier gère la parallélisation des chargements, le second lit les fichiers SQL indépendamment les uns des autres. le script « repartiteurSQL.bash » parcourt la liste des scripts SQL et lance les chargements en parallèle en fonction du nombre maximum de connexions au serveur de la base de données (la moitié du maximum de connexions défini dans postgresql.conf). Chaque chargement individuel est le fait du script « lectureSQL.bash ».

### II.1 Le script repartiteurSQL.bash

Ce script sert principalement à répartir les lancements des scripts SQL en processus parallèles en fonction du nombre maximum de connexions autorisées par la base de données. Le script lit le nombre maximum de connexions autorisées (une centaine par défaut) et parallélise au maximum la moitié de ce maximum de processus de chargement dans la base. Dès que le nombre de processus est atteint, des temporisations successives permettent d'attendre la fin de processus de chargement pour pouvoir en lancer d'autres.

Ce script prend deux paramètres (le second étant facultatif). Le premier est le nom de la base. Elle sera créé si besoin. Les extensions *postgis* et *postgis\_sfcgal* y seront chargées. Le deuxième est le nom de sous-dossier correspondant à un système de projection. Par exemple LAMBERT93, CC45, CC46 ... Par défaut, le paramètre sera LAMBERT93.

Exemple de lancement (scripts placés dans le dossier /home/media/carto/EDIGEO/FRANCE) :

```
cd /home/media/carto/EDIGEO/FRANCE
./repartiteurSQL.bash 07L93 LAMBERT93
```

Ces commandes traiteront les scripts SQL précédemment produits du dossier /home/media/carto/EDIGEO/FRANCE/LAMBERT93/DD/SQL où DD est un département listé dans le fichier departementATraiter situé dans le dossier de la projection.

Les éventuelles erreurs détectées seront stockées sous /home/media/carto/EDIGEO/FRANCE/LAMBERT93/ERREUR.

### II.2 Le script complementsCadastre.sql

Ce script ajoute des tables avec géométrie via des requêtes pour obtenir différents objet absents des échanges EDIGÉO. Ce sont :

- les départements,
- Les anciennes communes ayant fait l'objet de fusion. De nombreux anciens noms ont été récupérés depuis le site de l'INSEE mais la liste n'est pas complète.
- Les EPCI

Pour lancer ce script, vous lancerez la commande en spécifiant la base sur laquelle procéder :

```
cd /home/media/carto/EDIGEO/FRANCE # à adapter selon votre architecture
psql -f complementsCadastre.sql <Base>
```

Par exemple, pour une base ardéchoise nommée 07L93 :

```
psql -f complementsCadastre.sql 07L93
```