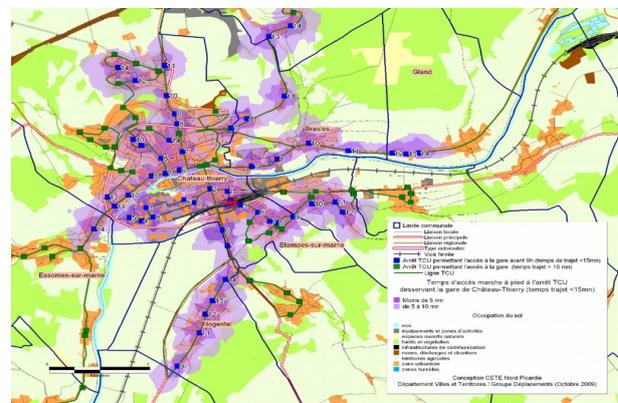
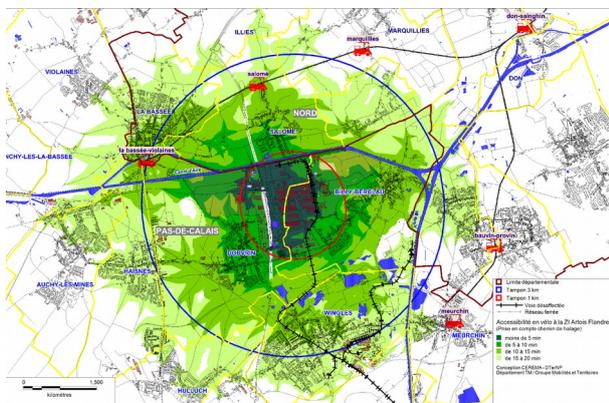
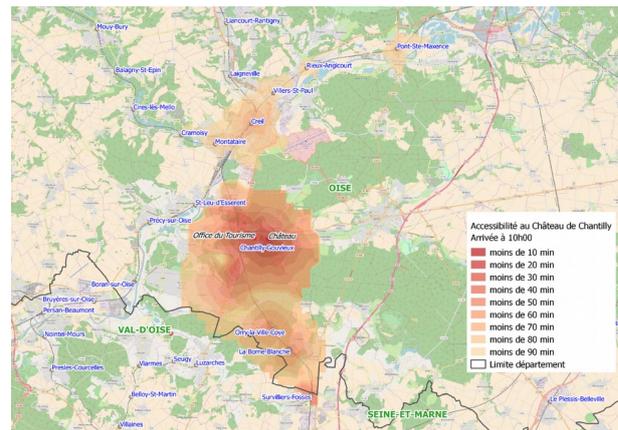
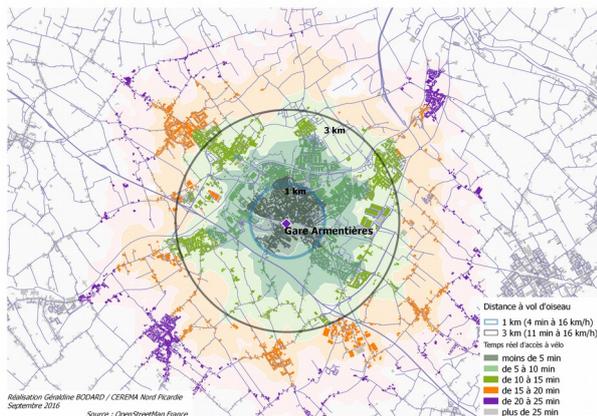


# Réaliser une carte accessibilité (marche, vélo ou TC) à partir d'un réseau multimodal





# Sommaire

AVERTISSEMENT.....	4
Partie 1 : Préparation des données.....	5
1.Récupérer un réseau Open street map (OSM).....	5
2.Nettoyer/Découper la table récupérée sous OSM.....	6
3.Ajouter et mettre à jour les colonnes dans la base voie découpée.....	7
4.Rendre les autoroutes intraversables.....	8
5.Préparation du sens 2.....	10
6.Générer les nœuds.....	11
7.Créer le réseau (vélo ou marche) .txt.....	12
8.Supprimer les nœuds isolés (nœuds qui sont inaccessibles / non connectés).....	13
9.Créer un réseau TC (générer réseau horaire pour carte accessibilité TC) - A partir d'un fichier GTFS non prêt pour musliw.....	14
10.Convertir réseau GTFS.....	15
11.Charger les arrêts (TC) dans Qgis.....	15
12.Générer les connecteurs.....	16
13.Création du fichier multimodal.....	17
Partie 2 : Réalisation de la carte accessibilité.....	18
1.Préparer la matrice pour musliw.....	18
2.Calcul paramètre.....	18
3.Calcul Musliw.....	19
4.Mettre à jour les champs ti et tj.....	19
5.Lancer l'interpolation linéaire.....	20
6.Créer isovaleurs (polygones).....	20
Partie 3 : Pour aller plus loin.....	22
1.Calcul population à l'intérieur des isochrones.....	22

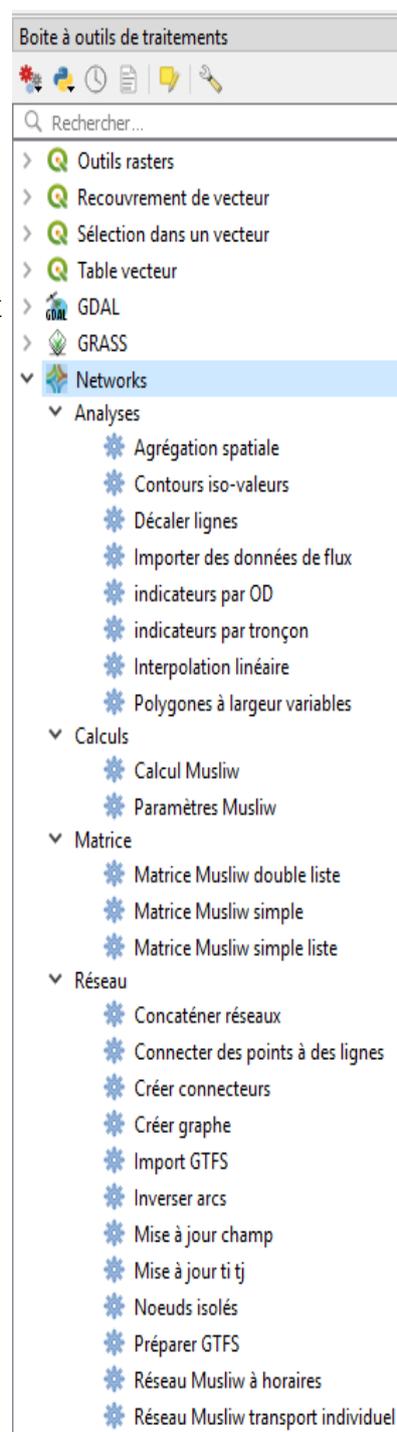
## AVERTISSEMENT

 **Ne pas mettre de blanc ni d'accent pour les noms de répertoire et/ou de tables.**  
**Exemple : Armentieres/voirie\_decoupe\_2sens**

La plupart des scripts utilisés pour :

- modifier les différentes couches voiries, nœuds, ...
- créer les différents réseaux (vélo, marche, transport)

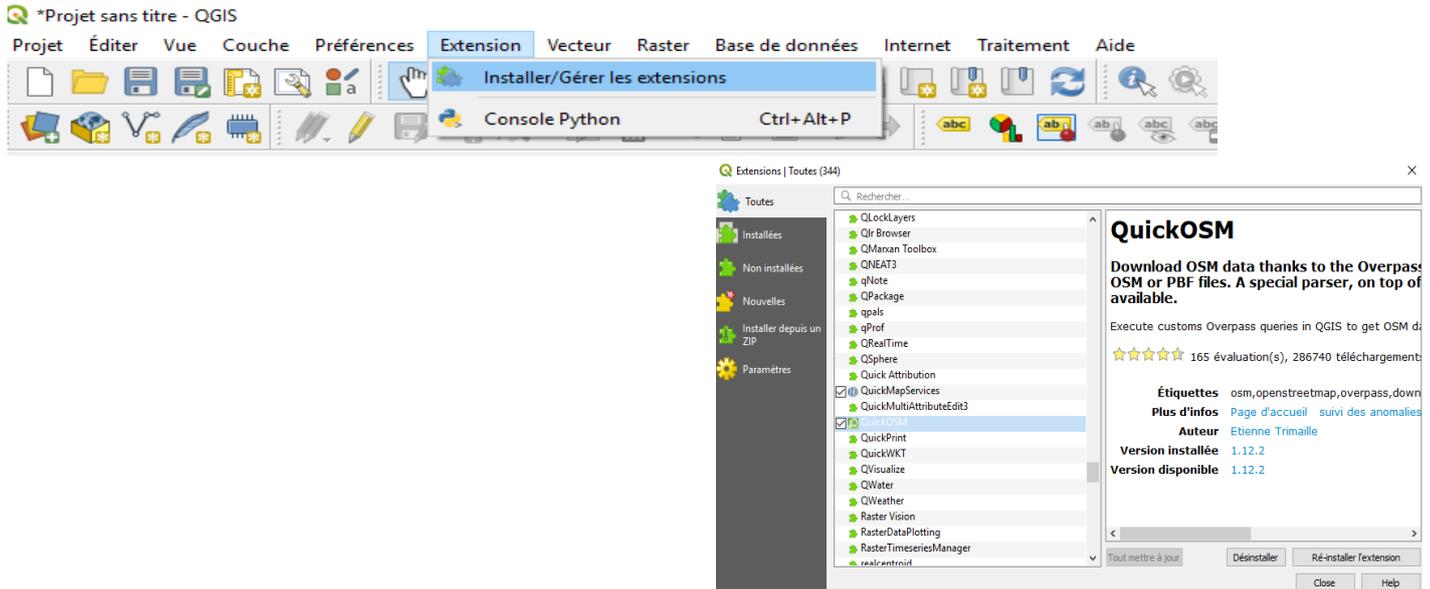
se trouvent dans la boîte à outils de traitements, sous l'onglet **Networks**.



# Partie 1 : Préparation des données

## 1. Récupérer un réseau Open street map (OSM)

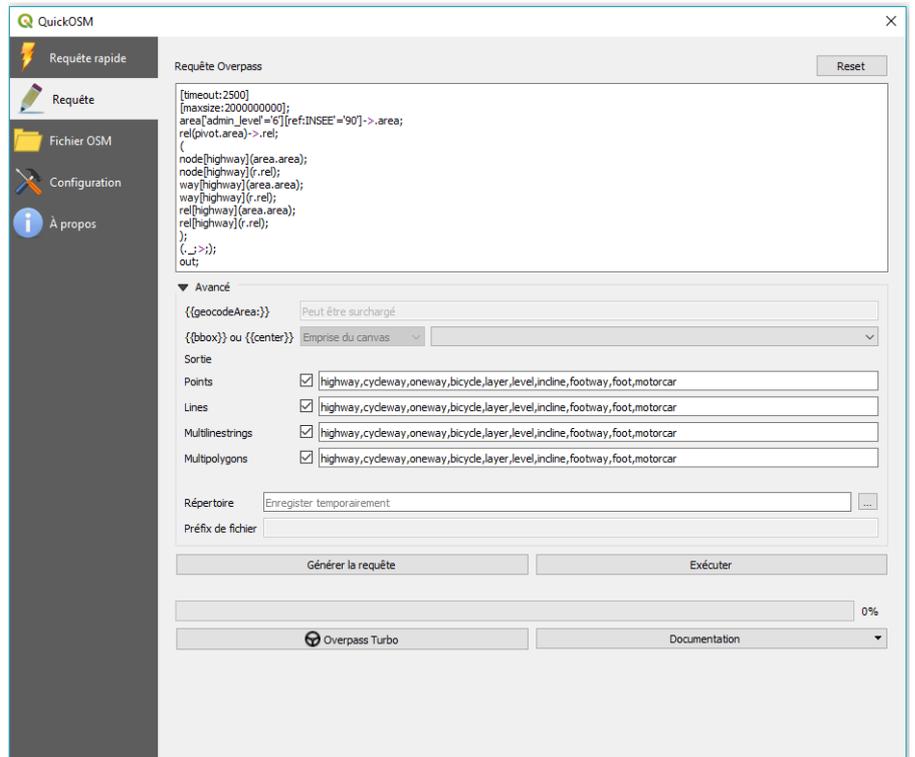
Onglet : **Extension** / Installer gérer les extensions : **Installer Quick OSM**



Saisir la requête suivante :

Exemple pour obtenir les voies du département 90.

```
[timeout:2500]
[maxsize:2000000000];
area[admin_level='6'][ref='90']->.area;
rel(pivot.area)->.rel;
(
node[highway](area.area);
node[highway](r.rel);
way[highway](area.area);
way[highway](r.rel);
rel[highway](area.area);
rel[highway](r.rel);
);
(._;>.);
out;
```



Ensuite :

Générer la requête

Exécuter

Fermer

Enregistrer la table « Osmquery » : **modifier le SCR (EPSG 2154)**

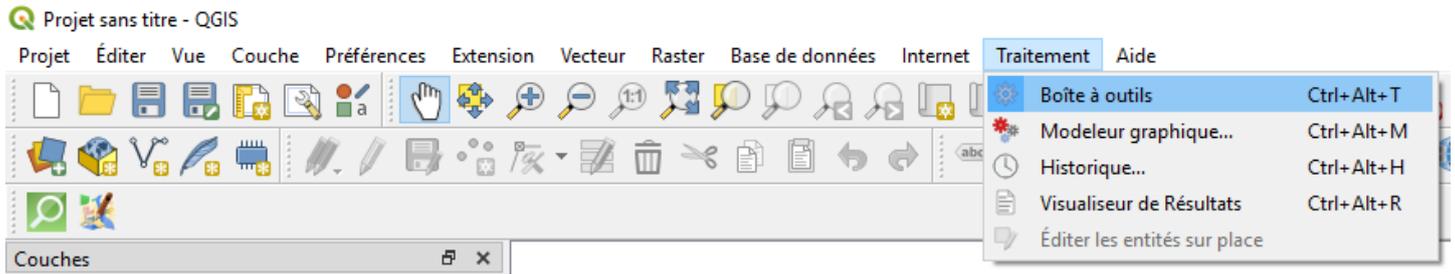
Exemple : DEPT90\_voirie

**Enregistrer la requête pour charger d'autres réseaux.**

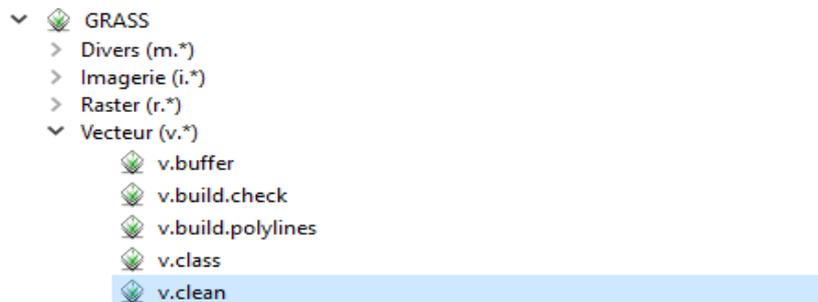
## 2. Nettoyer/Découper la table récupérée sous OSM

### Afficher la boîte à outils de traitement (si pas affichée)

Onglet : Traitement / Boîte à outils



### Afficher la Commande Grass / Vecteur (v.\*) / v.clean



### Commande Grass / Vecteur (v.\*) / v.clean :

Layer to clean : sélectionner la couche à nettoyer : **DEPT01\_Voirie**

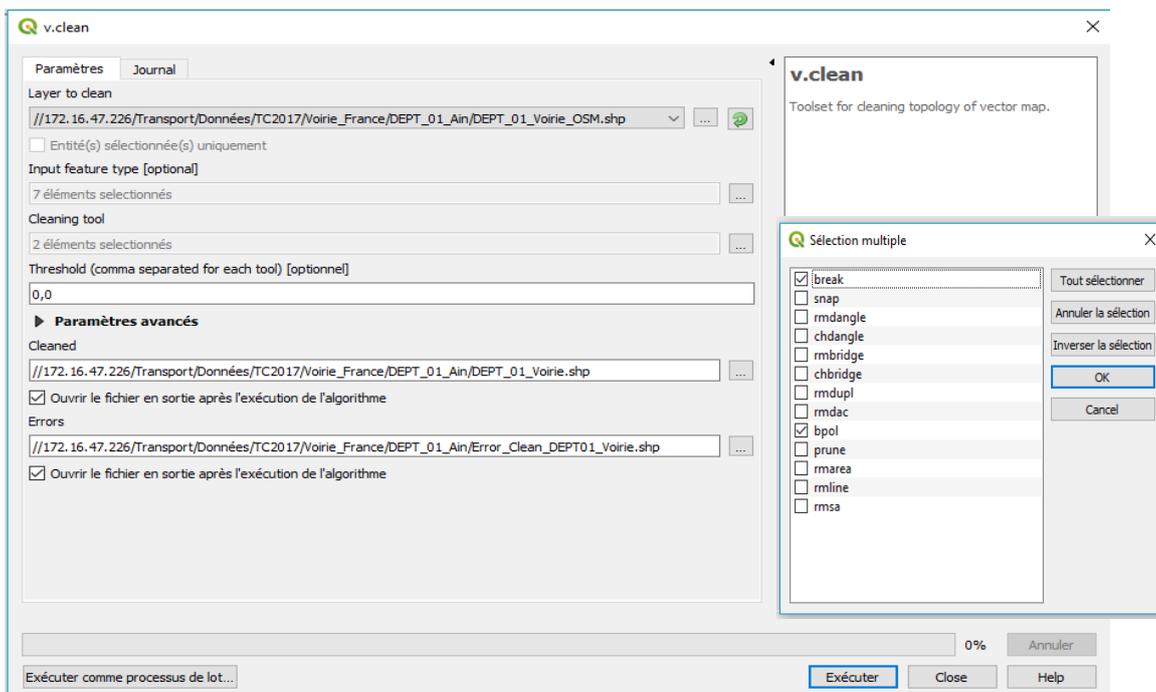
Cleaning tools : indiquer les outils : **break,bpol**

Threshold (comma separated for each tool) : Indiquer le séparateur : **0,0**

Cleaned : enregistrer le résultat dans un fichier : **Clean\_DEPT01\_Voirie**

Errors : enregistrer un fichier erreur : **Error\_clean\_DEPT01\_Voirie**

Exécuter



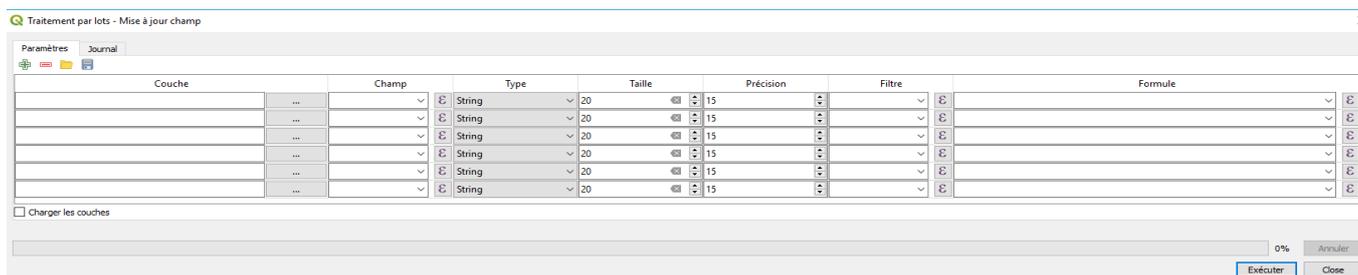
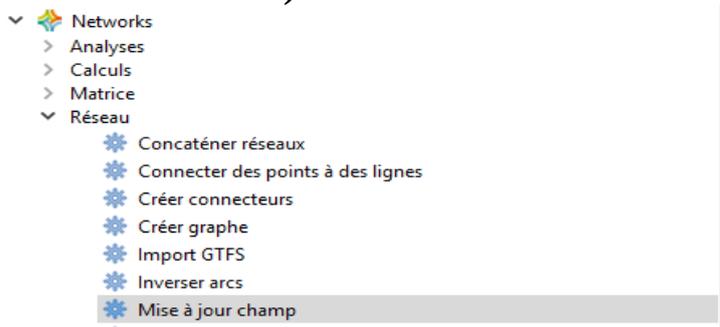
### Enregistrer la table cleaned (base voie découpée) ajoutée dans Qgis : modifier le SCR (EPSG 2154)

Exemple : Clean\_DEPT01\_voirie

### 3. Ajouter et mettre à jour les colonnes dans la base voie découpée

**SCRIPT : Mise à jour champ**

Clic droit sur script **Mise à jour champ** et sélectionner « Exécuter comme processus de lot » (penser à sauvegarder la requête pour l'utiliser à nouveau)



**Sélectionner la base voie à modifier :**

**Les colonnes à créer et à mettre à jour :**

Champ : **Sens**  
Type : **String**  
Taille : **1**  
Précision : **0**  
Filtre :  
Formule : **'1'**

Champ : **Longueur**  
Type : **Double**  
Taille : **15**  
Précision : **5**  
Filtre :  
Formule : **\$length**

Champ : **Diffusion**  
Type : **String**  
Taille : **1**  
Précision : **0**  
Filtre :  
Formule : **'3'** (autorise la diffusion dans les deux sens)

Champ : **Impasse**  
Type : **String**  
Taille : **1**  
Précision : **0**  
Filtre :

Formule : '0' (franchissable)

Champ : **Tps\_vélo** ou **Tps\_marche**

Type : **Double**

Taille : **15**

Précision : **5**

Filtre :

Formule :  $\$length*60/16000$  (vélo : 16 km/h) ou  $\$length*60/4000$  (marche : 4 km/h)

## Résultat :

Traitement par lots - Mise à jour champ

Couche	Champ	Type	Taille	Précision	Filtre	Formule
77/Voironne_France/DEPT_01_Ain/DEPT_01_Voironne_OSM.shp	Sens	String	1	0		'1'
77/Voironne_France/DEPT_01_Ain/DEPT_01_Voironne_OSM.shp	Longueur	Double	15	5		\$length
77/Voironne_France/DEPT_01_Ain/DEPT_01_Voironne_OSM.shp	Diffusion	String	1	0		'1'
77/Voironne_France/DEPT_01_Ain/DEPT_01_Voironne_OSM.shp	Impasse	String	1	0		'1'
77/Voironne_France/DEPT_01_Ain/DEPT_01_Voironne_OSM.shp	Tps_marche	Double	15	5		\$length*60/4000

Charger les couches

0% Annuler Exécuter Close

## Correspondances pour la colonne Type :

String = chaîne de texte

Double = décimal

DateTime = Date/Heure

La colonne « **filtre** » est à utiliser pour la mise à jour d'un champ existant.

## 4. Rendre les autoroutes intraversables

### SCRIPT : **Mise à jour Champ**

Clic droit sur script **Mise à jour Champ** et sélectionner **Exécuter** comme processus de lot (penser à sauvegarder la requête pour l'utiliser à nouveau)

- Networks
  - Analyses
  - Calculs
  - Matrice
  - Réseau
    - Concaténer réseaux
    - Connecter des points à des lignes
    - Créer connecteurs
    - Créer graphe
    - Import GTFS
    - Inverser arcs
    - Mise à jour champ
      - Exécuter...
      - Exécuter comme processus de lot...
      - Éditer les styles de rendu pour les sorties ...
    - Mise à jour ti tj
    - Noeuds isolés
    - Préparer GTFS
    - Réseau Musliw à horaires
    - Réseau Musliw transport individuel

Traitement par lots - Mise à jour champ

Couche	Champ	Type	Taille	Précision	Filtre	Formule
		String	20	15		

Charger les couches

0% Annuler Exécuter Close

**Les colonnes à mettre à jour :**

Champ : **Sens**

Type : **laisser le champ affiché par défaut**

Taille : **laisser le champ affiché par défaut**

Précision : **laisser le champ affiché par défaut**

Filtre : **highway in ('motorway','motorway\_link','trunk','trunk\_link')**

Formule : '0'

Champ : **Diffusion**

Type : **laisser le champ affiché par défaut**

Taille : **laisser le champ affiché par défaut**

Précision : **laisser le champ affiché par défaut**

Filtre : **highway in ('motorway','motorway\_link','trunk','trunk\_link')**

Formule : '0'

Champ : **Impasse**

Type : **laisser le champ affiché par défaut**

Taille : **laisser le champ affiché par défaut**

Précision : **laisser le champ affiché par défaut**

Filtre : **highway in ('motorway','motorway\_link','trunk','trunk\_link') and "layer" is Null**

Formule : '3' (autoriser des 2 côtés sauf autoroute)

Champ : **Impasse**

Type : **laisser le champ affiché par défaut**

Taille : **laisser le champ affiché par défaut**

Précision : **laisser le champ affiché par défaut**

Filtre : **highway in ('motorway','motorway\_link','trunk','trunk\_link') and "layer" in ('0')**

Formule : '3'

Champ : **Sens**

Type : **laisser le champ affiché par défaut**

Taille : **laisser le champ affiché par défaut**

Précision : **laisser le champ affiché par défaut**

Filtre : **foot in ('no')**

Formule : '0'

Champ : **Diffusion**

Type : **laisser le champ affiché par défaut**

Taille : **laisser le champ affiché par défaut**

Précision : **laisser le champ affiché par défaut**

Filtre : **foot in ('no')**

Formule : '0'

Champ : **Impasse**

Type : **laisser le champ affiché par défaut**

Taille : **laisser le champ affiché par défaut**

Précision : **laisser le champ affiché par défaut**

Filtre : **foot in ('no')**

Formule : '3'

## Résultat :

Qgis Traitement par lots - Mise à jour champ

Couche	Champ	Type	Taille	Précision	Filtre	Formule
DEPT_01_Clean_Voirie_2sens	abc Sens	String	20	15	highway in ('motorway','motorway_link','trunk','trunk_link')	'0'
DEPT_01_Clean_Voirie_2sens	abc Diffusion	String	20	15	highway in ('motorway','motorway_link','trunk','trunk_link')	'0'
DEPT_01_Clean_Voirie_2sens	abc Impasse	String	20	15	highway in ('motorway','motorway_link','trunk','trunk_link') and 'layer' is Null	'3'
DEPT_01_Clean_Voirie_2sens	abc Impasse	String	20	15	highway in ('motorway','motorway_link','trunk','trunk_link') and 'layer' in ('0')	'3'
DEPT_01_Clean_Voirie_2sens	abc Sens	String	20	15	foot in ('no')	'0'
DEPT_01_Clean_Voirie_2sens	abc Diffusion	String	20	15	foot in ('no')	'0'
DEPT_01_Clean_Voirie_2sens	abc Impasse	String	20	15	foot in ('no')	'3'

0% Annuler

Exécuter Close

## 5. Préparation du sens 2

### SCRIPT : **Inverser arcs**

Réseau : **Ouvrir la couche réseau à inverser**

Ajouter au réseau (si cocher : le sens 2 s'ajoute directement à la couche réseau)

Enregistrer le résultat

Ouvrir ou pas la couche dans Qgis.

- Networks
  - Analyses
  - Calculs
  - Matrice
  - Réseau
    - Concaténer réseaux
    - Connecter des points à des lignes
    - Créer connecteurs
    - Créer graphe
    - Import GTFS
    - Inverser arcs**

### Résultat :

Inverser arcs

Paramètres Journal

Réseau

DEPT\_01\_Clean\_Voirie\_2sens [EPSG:2154]

Ajouter au réseau?

Réseau inversé

[Créer une couche temporaire]

Ouvrir le fichier en sortie après l'exécution de l'algorithme

0% Annuler

Exécuter comme processus de lot... Exécuter Close

**Inverser arcs**

Inverse la géométrie des arcs et met à jour i,j et ij pour préserver la cohérence du graphe

Paramètres:

réseau : couche réseau (objets linéaires)

réseau inversé: Nom du réseau inversé

ajouter au réseau: si True ajoute le réseau inversé au réseau initial

### Uniquement pour réseau vélo « à adapter selon le territoire »

oneway= 'yes' and highway in ('primary','secondary','tertiary','primary link','secondary link','tertiary link') and cycleway is Null  
Mettre à jour champ Sens : '0'

A partir du boulier :



Mise à jour de la colonne « incline » pour inversion des inclinaisons (champs à adapter selon les types d'inclinaison dans la couche)

```

Case when « incline »= 'up' then 'down'
when « incline »= 'down' then 'up'
when « incline »= '-8%' then '8%'
when « incline »= '10%' then '-10%'
when « incline »= '8%' then '-8%'
when « incline »= '-10%' then '10%'
else Null
End
  
```

## 6. Générer les nœuds

### SCRIPT Créer graphe

Réseau : **Charger la couche réseau**

Node id : **Chaîne géographique (unique)**

Préfixe : **m** (pour différencier les nœuds des différents modes de transport)

Table des nœuds : **Donner un nom à la couche des nœuds**

Exécuter

- Networks
  - > Analyses
  - > Calculs
  - > Matrice
  - Réseau
    - Concaténer réseaux
    - Connecter des points à des lignes
    - Créer connecteurs
    - Créer graphe**

### Résultat :

**Créer graphe**

Crée un graphe à partir d'une couche d'objets linéaires (créer une couche de nœuds, crée et met à jour les champs i, j et ij avec les identifiants de nœuds) en tenant compte du sens de circulation. Génération des identifiants de nœuds soit par un compteur ou une chaîne géographique (unique) qui permet de fusionner des réseaux contigus

Paramètres:

réseau : Couche réseau (objets linéaires)

préfixe: préfixe pour les identifiants de nœuds (ex: 'MAP'; et n° 12563 => MAP12563)

node\_id: format des identifiants de nœuds: compteur ou chaîne géographique (unique)

sens: sens de circulation ('0':interdit,'1': sens unique sens de a géométrie, '2': sens unique sens inverse de la géométrie, '3': double sens)

fichier nœuds: couche des nœuds (extrémités des arcs)

0% Annuler

Exécuter Close

### Uniquement pour réseau vélo « à adapter selon le territoire »

Mise à jour du champ temps vélo à partir du boulier (*champs à adapter selon les types d'inclinaison dans la couche*)

```
case when « incline » = 'up' then $length*60/8000
when « incline » = 'down' then $length*60/24000
when « incline » = '-8 %' then $length*60/32000
when « incline » = '10 %' then $length*60/4000
when « incline » = '8 %' then $length*60/4000
when « incline » = '-10 %' then $length*60/32000
else NULL
END
```

## 7. Créer le réseau (vélo ou marche) .txt

### SCRIPT Réseau Musliw transport individuel

Permet de générer un réseau transport individuel au format Musliw à partir d'une couche Qgis

Ouvrir la couche réseau

Lancer script ti

Réseau routier : **nom de la couche voirie**

Sens : **Sens**

Temps : **Tps\_marche**

Longueur : **Longueur**

i-node : **i**

j-node : **j**

Id période : **ne rien modifier**

Id plage horaire : **ne rien modifier**

Heure début : **ne rien modifier**

Heure fin : **ne rien modifier**

Calendrier : **ne rien modifier**

Texte arc : **highway**

Mode : **m** (si marche)

Réseau musliw : **Enregistrer réseau (avec extension .txt)**

**Exécuter**

- Networks
  - Analyses
  - Calculs
  - Matrice
  - Réseau
    - Concaténer réseaux
    - Connecter des points à des lignes
    - Créer connecteurs
    - Créer graphe
    - Import GTFS
    - Inverser arcs
    - Mise à jour champ
    - Mise à jour ti tj
    - Noeuds isolés
    - Préparer GTFS
    - Réseau Musliw à horaires
    - Réseau Musliw transport individuel**

### Résultats

Réseau Musliw transport individuel

Paramètres Journal

Réseau routier

DEPT\_D1\_Clean\_Voirie\_2sens [EPSG:2154]

Entité(s) sélectionnée(s) uniquement

Fenêtre (xmin, xmax, ymin, ymax)

Sens

abc Sens

Temps

1..2 Tps\_marche

Longueur

1..2 Longueur

i-node

abc j

j-node

abc j

Id Période

-1

Id plage horaire

-1

Heure début

-1

Heure fin

-1

Calendrier

-1

Texte arc

abc highway

Mode

m

Réseau Musliw

[Enregistrer dans un fichier temporaire]

Enregistrer dans un fichier temporaire

Enregistrer vers un fichier...

Changer l'encodage du fichier (System)...

Exécuter Close

**Réseau Musliw transport individuel**

Génère un réseau Musliw de mode individuel (voiture, marche, vélo,...) à partir d'une couche réseau (objets linéaires)

Paramètres:

réseau routier: couche réseau routier (objets linéaires)

fenêtre: aire de génération (peut être une sous partie du réseau global)

sens: sens de circulation ('0' interdit, '1' sens unique de la géométrie, '2' sens unique sens inverse de la géométrie, '3' double sens)

temps: temps de parcours sur l'arc

longueur: longueur de l'arc

i-node: identifiant noeud i

j-node: identifiant noeud j

id période: identifiant de la période

id plage horaire: identifiant de la plage horaire

heure début: heure de début de la plage horaire

heure fin: heure fin de la plage horaire

calendrier: calendrier de la plage horaire (ex: 'OOOONN' tous les jours sauf samedi et dimanche)

texte arc: libellé arc

mode: identifiant de mode pour le calcul de plus courts chemins multimodal

réseau Musliw: Nom du réseau Musliw généré (txt file)

## 8. Supprimer les nœuds isolés (nœuds qui sont inaccessibles / non connectés)

### SCRIPT **Nœuds isolés**

**Nœuds : ouvrir la couche nœud**

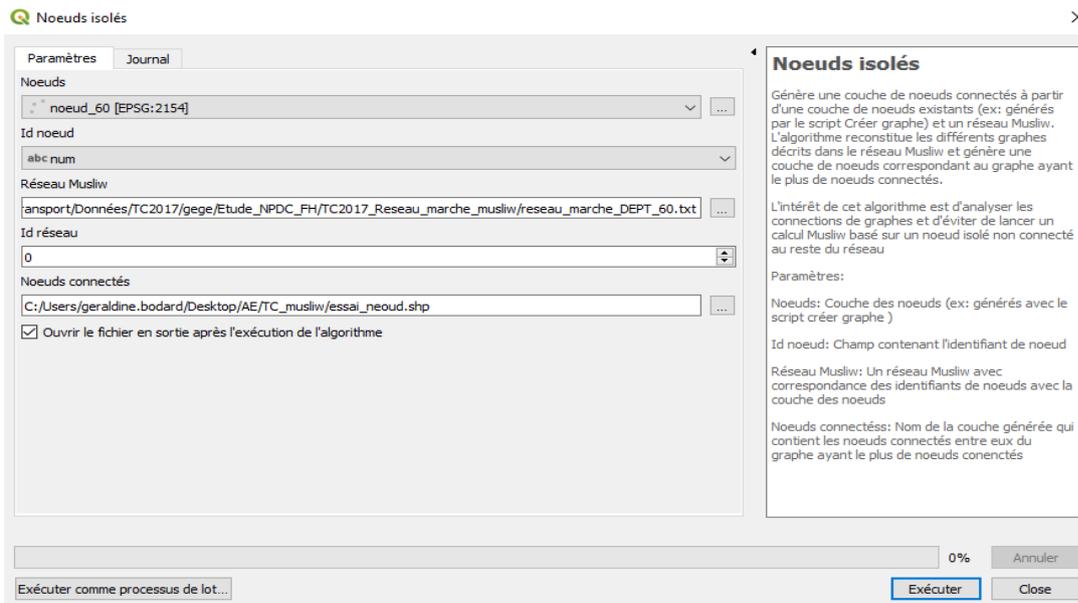
Id nœud : **num**

Réseau Musliw : **ouvrir réseau marche .txt**

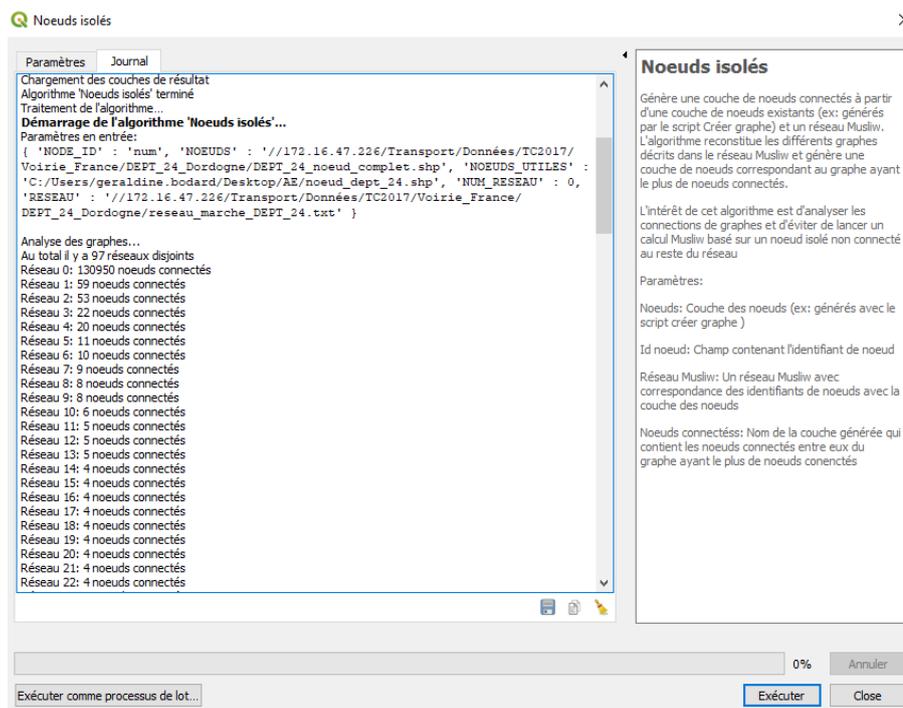
Id réseau : 0 dans un premier temps, ensuite 1 voire 2, etc. selon le résultat de l'analyse des graphes

Nœuds connectés : **enregistrer le fichier**

- Networks
- > Analyses
- > Calculs
- > Matrice
- Réseau
  - Concaténer réseaux
  - Connecter des points à des lignes
  - Créer connecteurs
  - Créer graphe
  - Import GTFS
  - Inverser arcs
  - Mise à jour champ
  - Mise à jour ti tj
  - Noeuds isolés**



Dans cet exemple, il y a pour le réseau 0, 130 950 nœuds connectés.



## 9. Créer un réseau TC (générer réseau horaire pour carte accessibilité TC) - A partir d'un fichier GTFS non prêt pour musliw

### SCRIPT Préparer GTFS

Clic droit sur le script / Exécuter par lot

Sélectionner le **répertoire source** (endroit où sont stockés les GTFS)

Id réseau : **AE** (exemple AE pour Aéroport Lille)

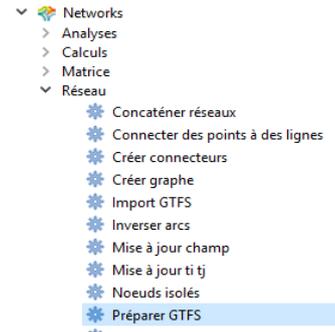
Extraire stop\_id ? (UIC) : **Non**

Formule d'extraction : **laisser la valeur par défaut parce que UIC = non**

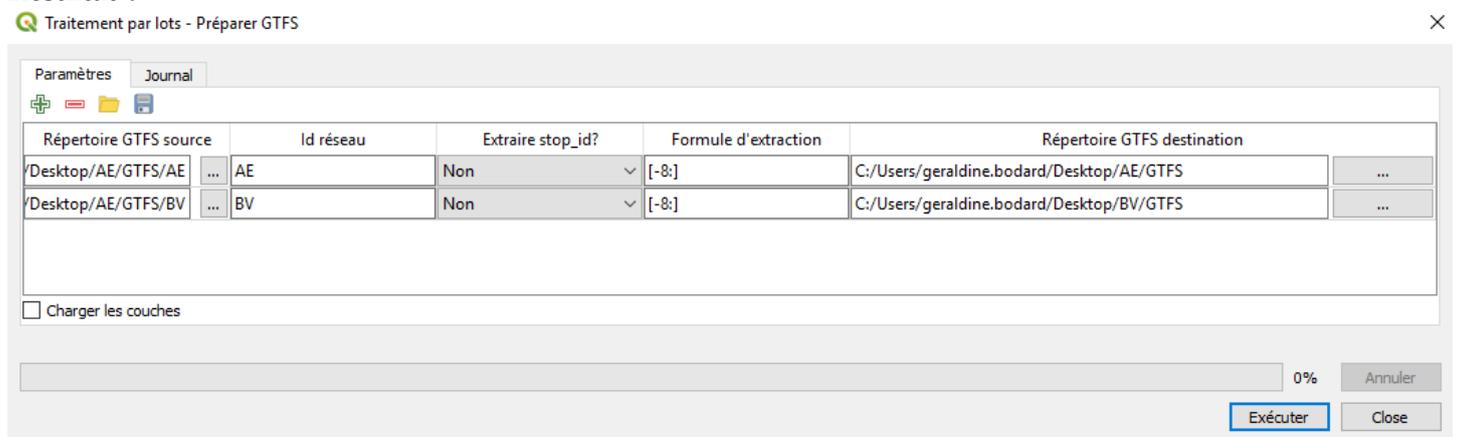
Répertoire GTFS destination : Enregistrer le résultat dans le **répertoire**

**résultat**. Exemple répertoire « **GTFS** »

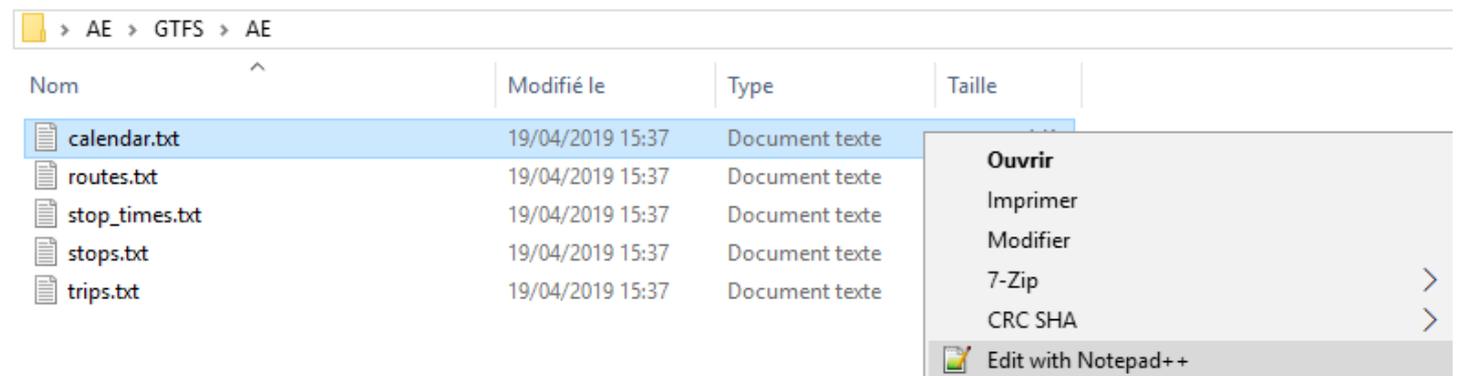
**Exécuter**



### Résultat :



**Dans le répertoire résultat** (sauvegarde des fichiers GTFS créés), éditer le fichier (avec par exemple Notepad) **calendar.txt de chaque réseau** afin de sélectionner une semaine type (hors périodes de vacances). Exemple : du 10/06/2019 au 16/06/2019



## 10. Convertir réseau GTFS

### SCRIPT Réseau Musliw à horaires

Répertoire GTFS : ouvrir le répertoire où se trouvent les fichiers GTFS

Début calendrier : **Indiquer date de début de semaine (démarré le lundi)**

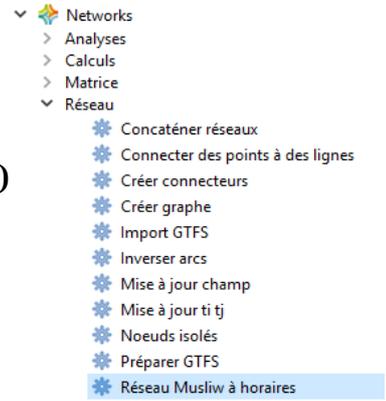
Fin calendrier (semaine type) **Indiquer date de fin de semaine (termine**

**le dimanche)**

Réseau musliw à horaires : Enregistrer le nom du réseau (**avec**

**extension .txt) – Exemple AE.txt**

**Exécuter**



## 11. Charger les arrêts (TC) dans Qgis

### SCRIPT Import GTFS

Clic droit / Exécuter par lot (penser à sauvegarder la requête)

Répertoire GTFS : **Sélectionner le répertoire GTFS**

Début calendrier : **10/06/2019**

Fin calendrier : **16/06/2019** (si semaine choisie lundi 10 juin 2019 au dimanche 16 juin 2019)

Heure début : **laisser les valeurs par défaut**

Heure fin : **laisser les valeurs par défaut**

Nom des tables : **AE** (remettre le nom du réseau exemple AE pour aéroport de Lille)

CRS : **laisser les valeurs par défaut (EPSG 2154)**

Encodage : « **utf8** » **laisser cette valeur par défaut. Si jamais ne fonctionne pas avec cette variable, remplacer « utf8 » par « cp1252 »**

Répertoire destination : **créer un répertoire de sauvegarde** (exemple shp (les fichiers de sorties sont des .shp))

**Exécuter**

**Résultat :**



## 12. Générer les connecteurs

Afficher les nœuds précédemment générés  
Afficher le réseau voirie

### SCRIPT Créer connecteurs

Permet à partir de deux couches (arrêts de transport en commun et nœuds d'un réseau routier) de générer un fichier Musliw de connecteurs. **Le connecteur créé est visible.**

Connecter les arrêts aux nœuds piéton les plus proches

Clic droit / Exécuter par lots

arrêt : **arrêt TC (stops.shp)**

stop id : **ident**

texte nœud i : **t (TC)**

mode nœud i : **t (TC)**

nœud : **table nœud piéton**

node id : **num**

texte nœud j : **m (marche)**

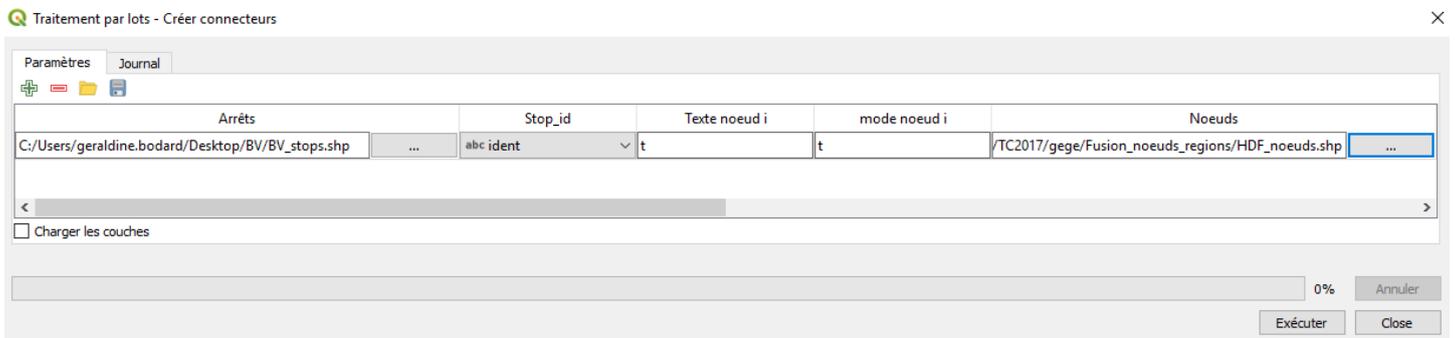
mode nœud j : **m (marche)**

rayon recherche : **1000**

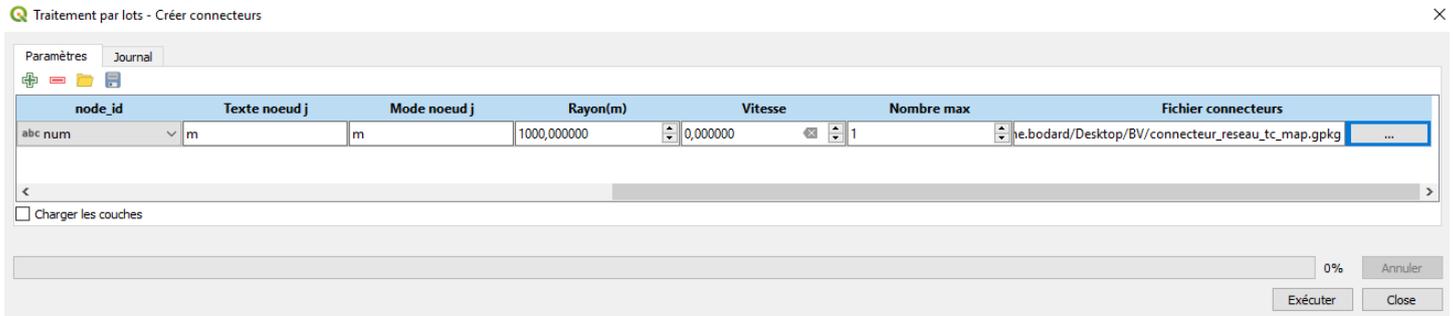
vitesse : **0**

Nombre max : **1**

Enregistrer le fichier : `connecteur_reseau_tc_map`



### (suite du script)



## 13. Création du fichier multimodal

### SCRIPT Concaténer réseaux

Concaténer les différents réseaux dont l'extension est .txt

Réseau marche  
Connecteurs  
Arrêts piétons  
Horaires TC  
...

Mettre l'ensemble des fichiers à concaténer dans un répertoire (**exemple : répertoire « concatener »**)  
source : **sélectionner le répertoire où se trouvent les fichiers .txt à concaténer**  
réseau multimodal : **enregistrer réseau multimodal (avec extension .txt)**  
Exécuter

### Résultat :



# Partie 2 : Réalisation de la carte accessibilité

## 1. Préparer la matrice pour musliw

### SCRIPT **Matrice Musliw simple**

Nœuds : **fichier nœuds**

Id nœud : **num**

Point de départ : **sélectionner directement dans Qgis le nœud de départ**

Point d'arrivée : **sélectionner directement dans Qgis le nœud d'arrivée**

Demande : **nombre de passagers**

Jour : **1**

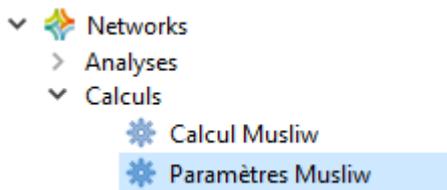
Horaire : **horaire format 00:00:00**

Départ/Arrivée : **à choisir**

Mode écriture : **écriture ou Ajout (lot)**

Fichier matrice : **enregistrer la matrice**

## 2. Calcul paramètre



### SCRIPT **Paramètre Musliw**

Facteur d'échelle temps individuel :  
**0,25 pour le vélo**

Temps détaillé : **Sans les arcs à horaires** (fichier temps)

### 3. Calcul Musliw

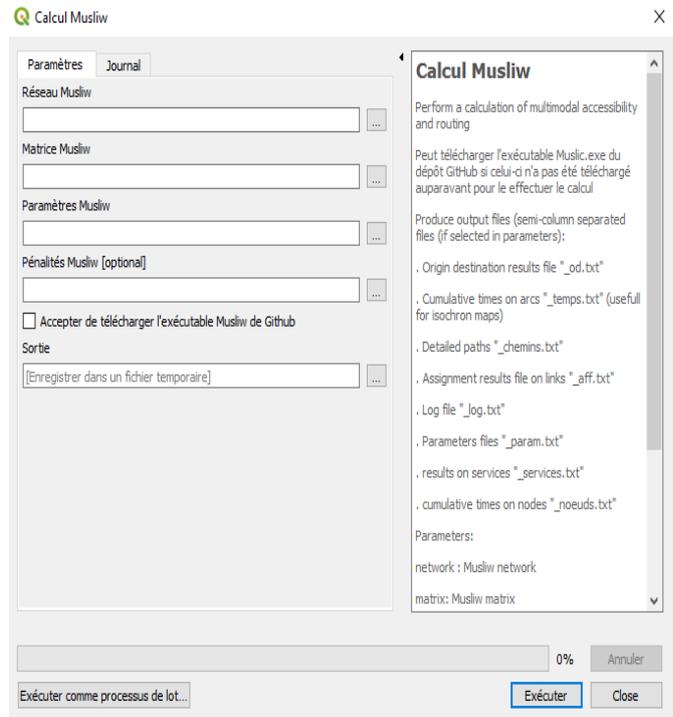
Réseau Musliw : **Réseau marche txt**

Matrice Musliw : **Matrice**

Paramètre Musliw : **fichier paramètre**

Pénalités Musliw (optional) : **fichier pénalités (si pénalités)**

Accepter de télécharger l'exécutable Musliw : **Cocher la case si Musliw n'a pas encore été télécharger**



### 4. Mettre à jour les champs ti et tj

SCRIPT **Mise à jour ti tj**

Réseau: **reseau\_voirie.shp**

Fichier temps de parcours : **sélectionner le fichier temps générer par musliw**

Temps musliw : **temps**

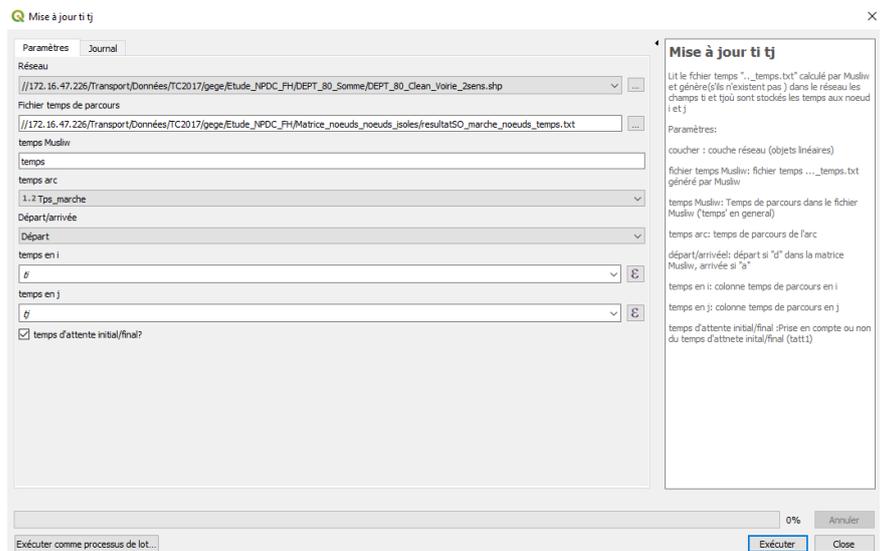
Temps arc : **Tps\_marche**

Départ/arrivée : **celui de la matrice musliw**

Temps en i : **ti si ce champ existe (mise à jour) ou possibilité de créer un nouveau champ, dans ce cas écrire « ti\_velo » par exemple.**

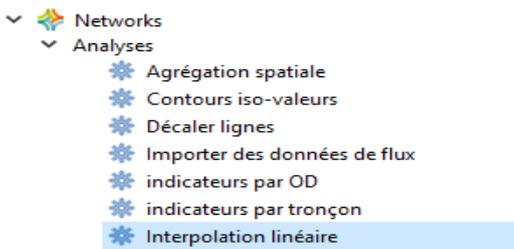
Temps en j : **tj (idem Temps en i)**

Cocher d'attente initial/final pour ne pas prendre en compte le temps restant depuis l'arrêt si par exemple arrivée à 8h20 alors que dans la matrice l'arrivée est prévue à 8h30. Dans ce cas il ne prend pas en compte les 10 min restantes.

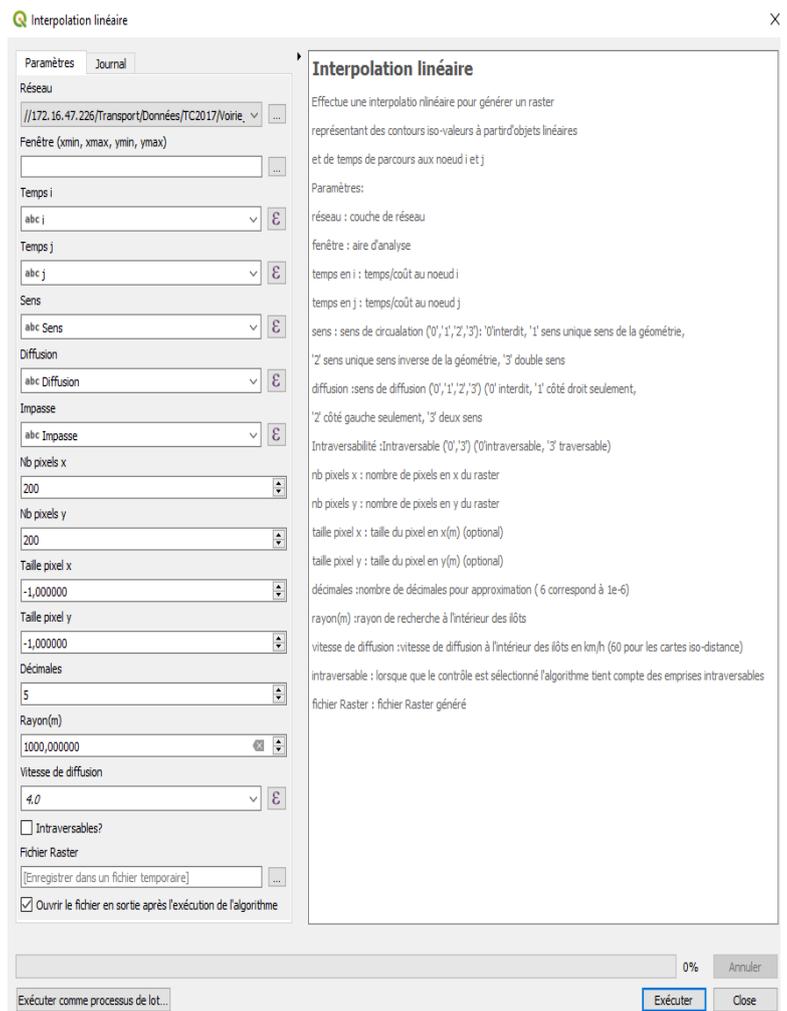


## 5. Lancer l'interpolation linéaire

### SCRIPT Interpolation linéaire



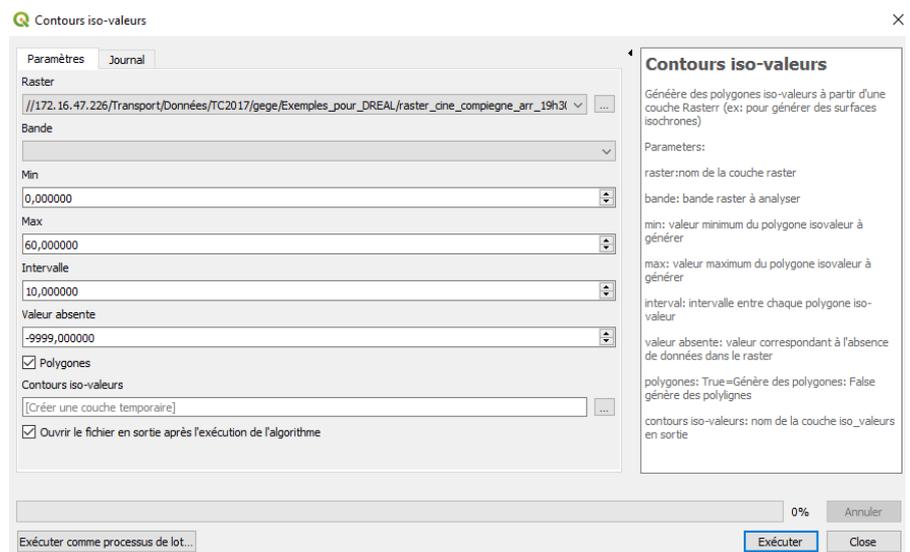
Réseau : **réseau\_voirie\_shp**  
Fenêtre : **permet de zoomer sur la couche**  
Temps i : **i**  
Temps j : **j**  
Sens : **Sens**  
Diffusion : **Diffusion**  
Impasse : **Impasse**  
Nb pixel x : **200**  
Nb pixel y : **200**  
Taille pixel x : **laisser les valeurs par défaut**  
Taille pixel y : **laisser les valeurs par défaut**  
Décimales : **5**  
Rayon(m) : **1000**  
Vitesse diffusion : **4** (4 km/h map) vitesse à adapter selon le mode (16 km/h vélo) ou pour une carte affichage en distance : **60**  
Intraversables : **ne pas cocher**  
Fichier Raster : enregistrer le raster  
**Exécuter**



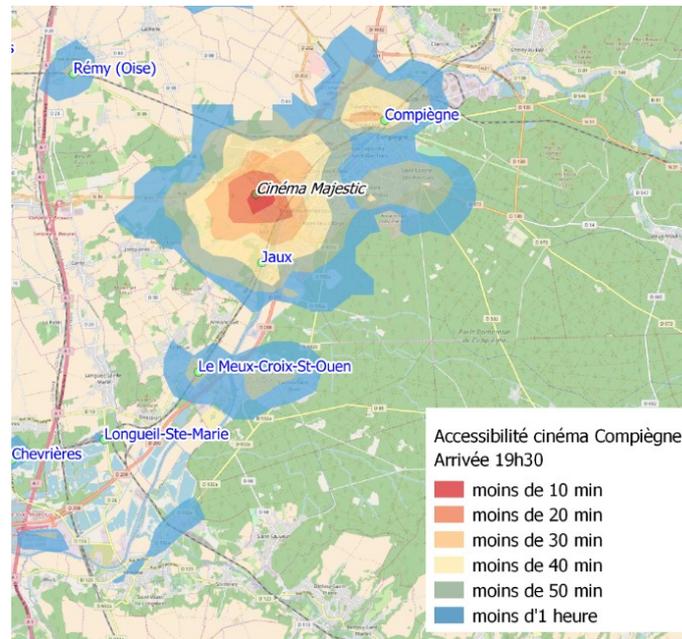
## 6. Créer isovaleurs (polygones)

### SCRIPT Contour iso-valeurs

Raster : **ouvrir le raster**  
Bande :  
Min : **0**  
Max : **60** selon durée choisie  
Intervalle : **10** selon le seuil choisi  
Valeur absente : **laisser la valeur par défaut**  
Polygones : **cocher la case**  
Contours iso-valeurs : **enregistrer l'isochrone**  
Ouvrir le fichier en sortie après l'exécution de l'algorithme : **cocher la case**  
**Exécuter**



## Exemple isochrone



# Partie 3 : Pour aller plus loin

## 1. Calcul population à l'intérieur des isochrones

### SCRIPT Agrégation spatiale

Ouvrir la couche isochrone  
Ouvrir la couche des communes avec les données « population »

Polygones : **afficher l'isochrone**

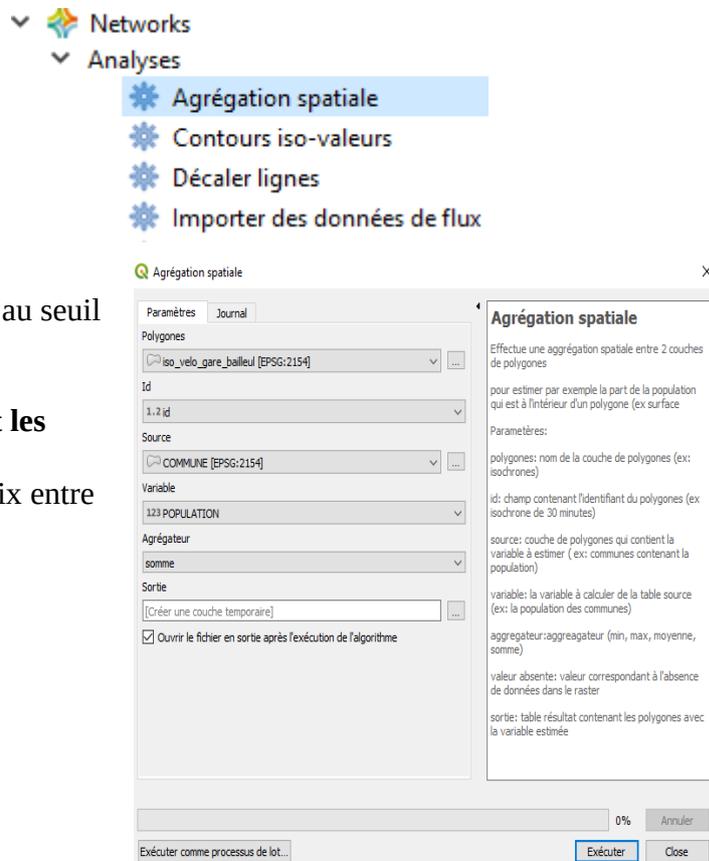
Id : **identifiant du polygone** (correspond au seuil défini dans le polygone)

Source : **table des communes**

Variable : **nom du champ où se trouvent les données « population »**

Agrégateur : **somme** (pour sommer). Choix entre moyenne, min, max

Sortie : **enregistrer le résultat**



### Résultats

Sortie :: Total des entités: 4, filtrées: 4, sélectionnées: 0

	id	POPULATION
1	0,00000	621,01899
2	5,00000	2381,86852
3	10,00000	5078,68296
4	15,00000	5542,47301

Montrer toutes les entités