



**HAL**  
open science

# Introduction aux données OpenStreetMap (Structuration, interrogation, extraction et édition)

Boris Mericskay

► **To cite this version:**

Boris Mericskay. Introduction aux données OpenStreetMap (Structuration, interrogation, extraction et édition). Master. France. 2017. cel-01660629

**HAL Id: cel-01660629**

**<https://hal.science/cel-01660629>**

Submitted on 11 Dec 2017

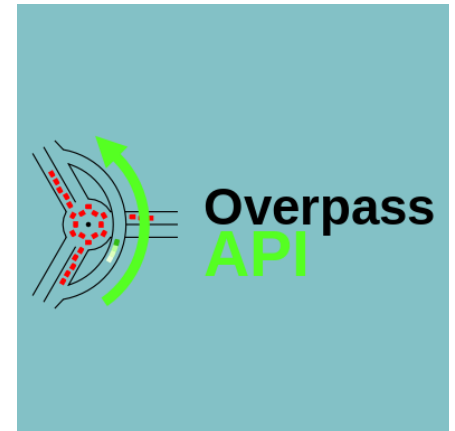
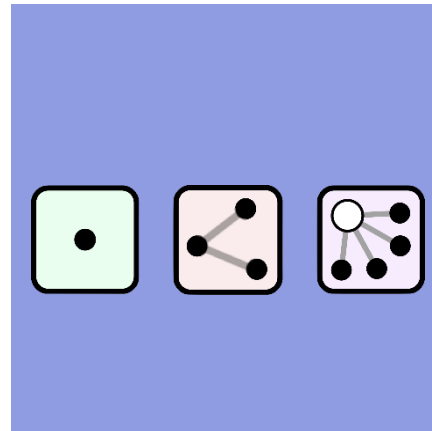
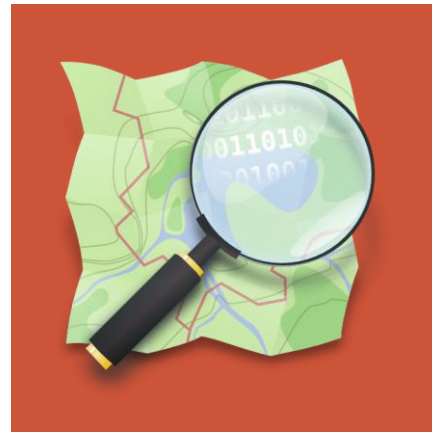
**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**UNIVERSITÉ  
RENNES 2**

**Introduction aux  
données OpenStreetMap  
(Structuration,  
interrogation, extraction  
et édition)**



**# M2 SIGAT / TELENVI  
Automne 2017**

@Boris Mericskay



# + Manipulation de données avec OSM



# Données OSM



Quelles sont les données disponibles sur OSM ?

Comment récupérer des données issues de OSM ?

- Selon une empreise
- Selon des critères qualitatifs

Comprendre comment sont structurées ces données

Savoir comment les réutiliser...

...et les faire évoluer



# + Les données OSM



Dans un SIG, les données cartographiques sont représentées de trois façons différentes

- Points, lignes et polygones
- Les données attachées à ces objets sont généralement stockées dans une base de données liée à la base géographique.

Dans OpenStreetMap, ces trois concepts sont modélisés différemment :

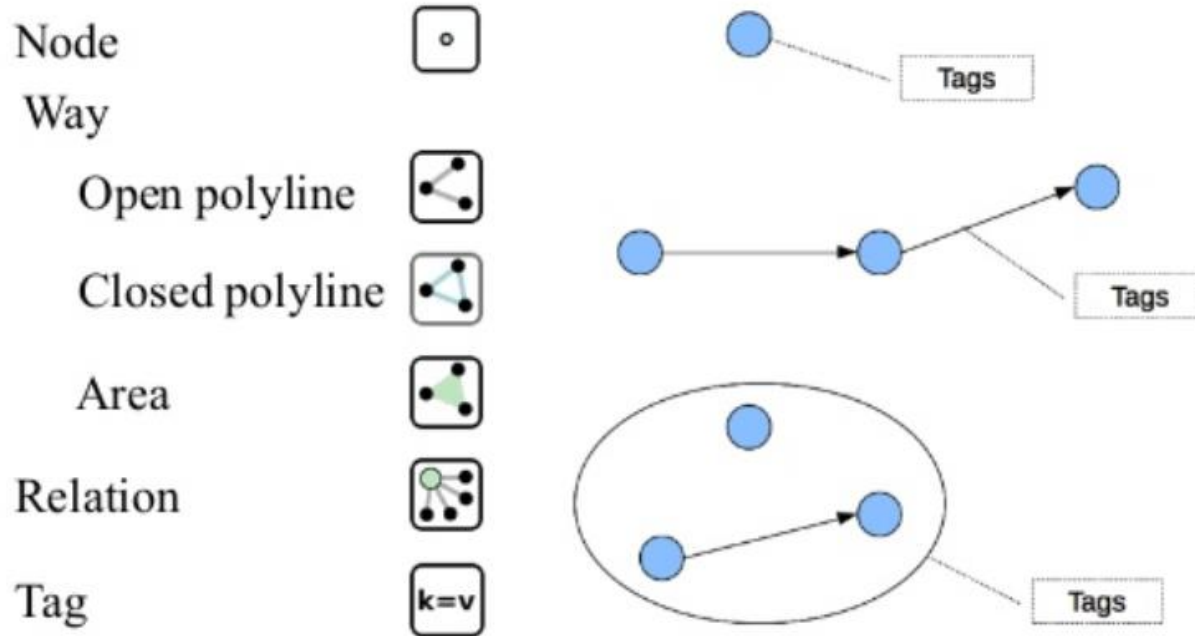
- **Noeuds** (*nodes*)
- **Lignes** (*ways*)
- **Relations**
- complétés par des **attributs** (*tags*) décrivant chaque objet



# + Les données OSM

Clip slide

## OSM Data Model

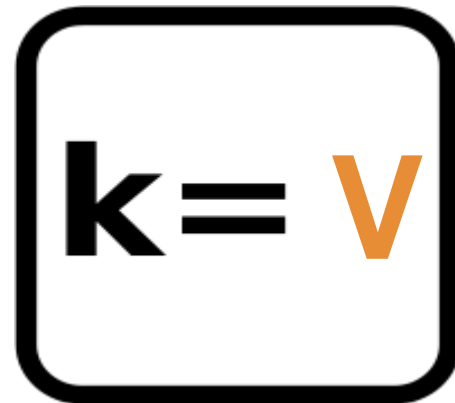


# + Les données OSM




Par exemple pour indiquer qu'un trait correspond à une route secondaire, en sens unique, de vitesse maximale 90 km/h et munie d'une bande cyclable, on utilisera :

- [highway=secondary](#)
- [oneway=yes](#)
- [maxspeed=90](#)
- [cycleway=lane](#)



# + OSM place search

 Cette extension est expérimentale

## OSM place search

Location search helper

Based on OSM data, Nominatim online tool  
([http://wiki.openstreetmap.org/wiki/Nominatim\\_usage\\_policy](http://wiki.openstreetmap.org/wiki/Nominatim_usage_policy))

★★★★☆ 44 évaluation(s), 48085 téléchargements

Étiquettes: [location search helper osm](#)  
Plus d'infos: [Page d'accueil](#) [bug tracker](#) [code repository](#)

Auteur: [Xavier Culos \(Agence de l'eau Adour Garonne\)](#)

Version installée : 0.9.3 (dans  
C:\Users\mericskay\_b\.qgis2\python\plugins\nominatim)  
Version disponible : 0.9.3 (dans Dépôt officiel des extensions  
QGIS)

Journal des modifications :  
0.9.3 : Change from Mapquest API to OSM Api (Mapquest  
require key)  
0.9.2 : Fix wrong panel management. Thanks to  
<https://github.com/Andre-J>  
0.9.1 : Few minor changes (readme, mask dependence)  
0.9 : QGIS 2.4 more compatible, use mask plugin if exists  
0.8.3 : fix encoding pb (<http://hub.qgis.org/issues/10920>,  
thanks to Augustin Roche)

Tout mettre à jour    Désinstaller    Ré-installer l'extension    Fermer    Aide

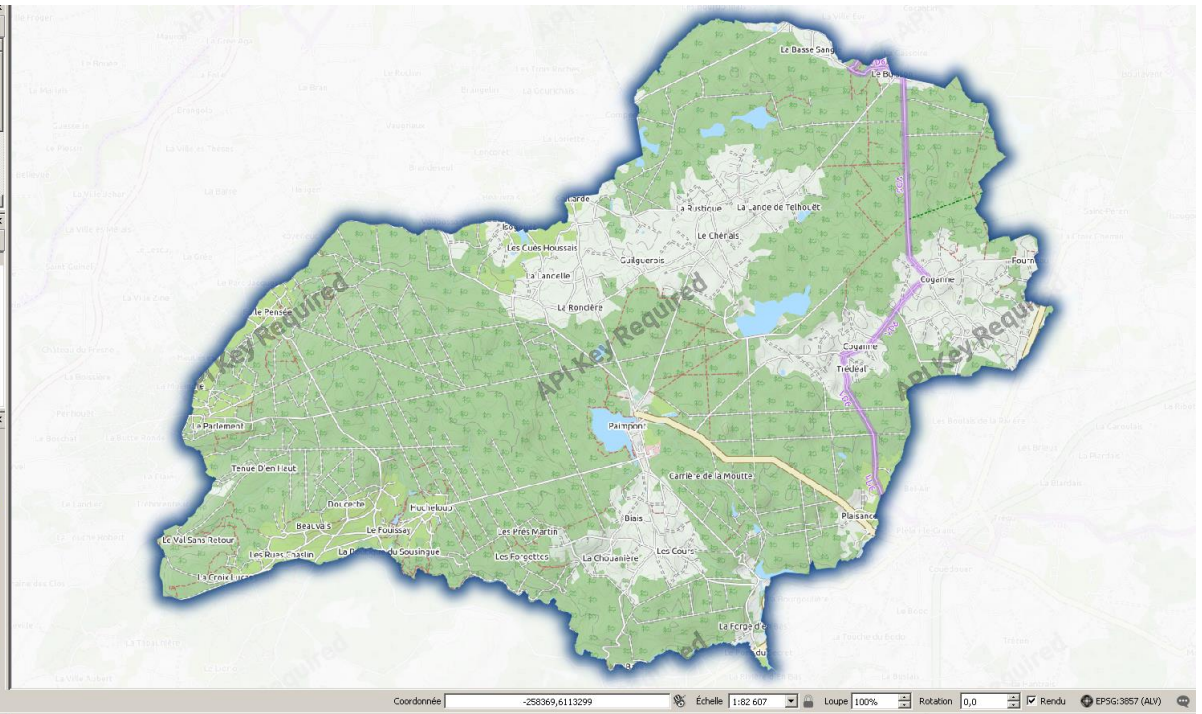
Recherche de lieux: OSM

Nom contenant...  >>

Limiter la recherche à la zone visible

Libellé
Paimpont, Rennes, Ille-et-Vilaine, Bretagne, France métropolitaine, 35330
Paimpont, Goven, Redon, Ille-et-Vilaine, Bretagne, France métropolitaine, :
Paimpont, Esplanade de Brocclande, La Gelle, Paimpont, Rennes, Ille-et-V

Zoomer







# OSM Info

## OSMInfo

Get full information about a point from OpenStreetMap database.

For a selected point, extract all information about nearby and enclosing features from OpenStreetMap database using Overpass API.

★★★★★ 14 évaluation(s), 29107 téléchargements

Catégorie: Vector

Étiquettes: osm,openstreetmap,overpass

Plus d'infos: [Page d'accueil](#) [bug tracker](#) [code repository](#)

Auteur: [NextGIS](#)

Version installée : 0.6.1 (dans

C:\Users\mericskay\_b\.qgis2\python\plugins\osminfo)

Version disponible : 0.6.1 (dans Dépôt officiel des extensions QGIS)

Journal des modifications :

0.6.1:

\* Save selected object into new memory layer.

0.6:

\* Account for language of the interface when showing results

\* Sort outputs by area for nicer hierarchy

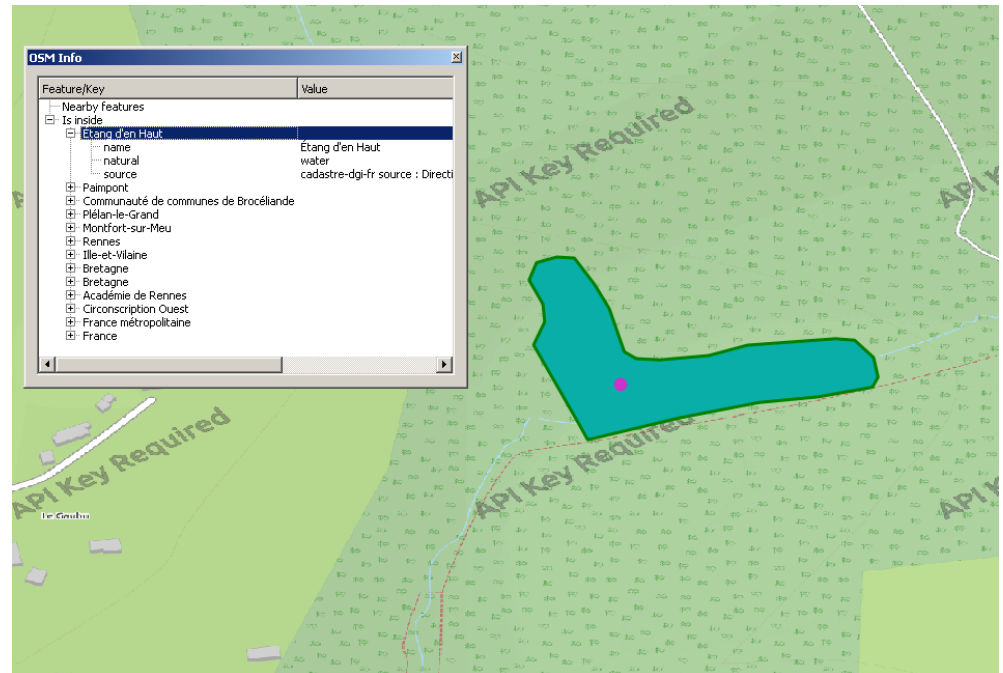
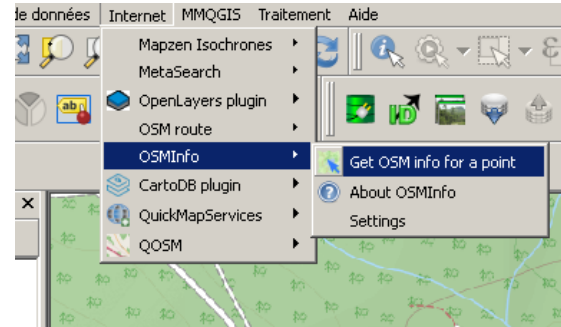
Tout mettre à jour

Désinstaller

Ré-installer l'extension

Fermer

Aide





# Isochrones

## Mapzen Isochrone Generator

**Mapzen Isochrones**

This plugin generates travel-time isochrones via the Mapzen Isochrone service API

QGIS Plugin to generate travel-time isochrones via the Mapzen Isochrone service API. This plugin requires a valid API key from Mapzen. This can be obtained at <https://mapzen.com/>. This plugin was created for personal use. I am not affiliated with Mapzen.

★★★★★ 10 évaluation(s), 3009 téléchargements

Catégorie: Web  
Étiquettes: analysis, distance, network analysis, openlayers, opentreetmap, osm, webservice, web, isochron, isochrone, traveltime, travel-time, travel time, drivetime, drive time, drive-time, accessibility, mapzen, API  
Plus d'infos: [Page d'accueil](#) [bug\\_tracker](#) [code\\_repository](#)

Auteur: [Ethan Monk](#)

Version installée : 0.1 (dans C:\Users\mericskay\_bv\qgis2\python\plugins\qgisMapzenIsochrones-master)  
Version disponible : 0.1 (dans Dépôt officiel des extensions QGIS)

Tout mettre à jour    Désinstaller    Ré-installer l'extension  
Fermer    Aide

**Mapzen Isochrones**

### Mapzen Isochrone Generator

Created by Ethan Monk  
[ethangmonk@gmail.com](mailto:ethangmonk@gmail.com)  
[emonk@trademarkproperty.com](mailto:emonk@trademarkproperty.com)

Powered by: © [Mapzen](#), [OpenStreetMap](#), and [others](#).

Mapzen API Key:  [Get Mapzen API Key](#)

Costing Model:

Travel-Time (Minutes):

Layer Name:

Input Latitude and Longitude (ESPG: 4326)

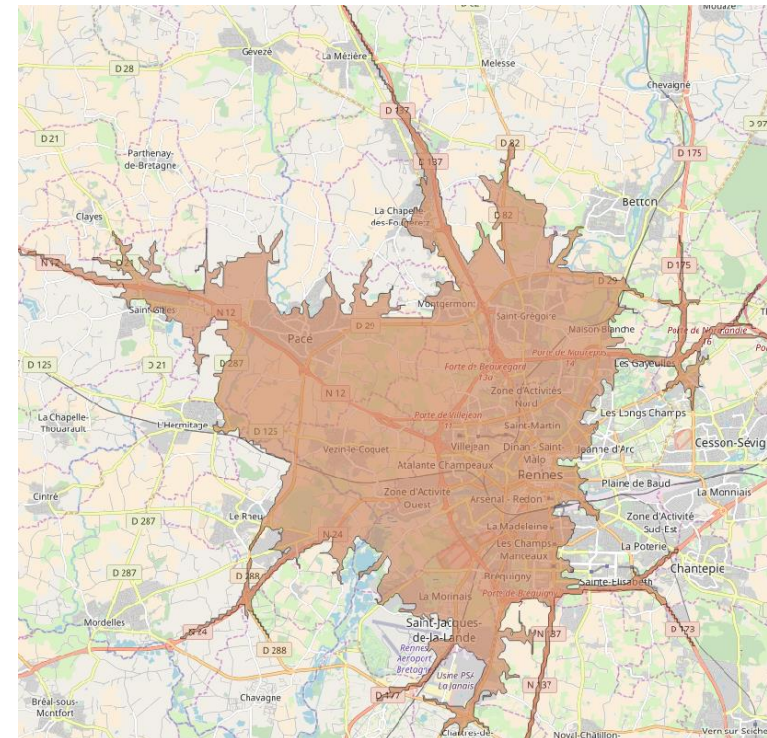
Latitude (Y):   
Longitude (X):

**IN DEVELOPMENT**

Select point-of-origin on map

Use point layer as point(s)-of-origin

OK    Annuler





# Isochrones

## OSM Tools

**OSM Tools**  
powered by [OpenRouteService](#)  
brought to you by [Nils Nolde](#)

API key  
 [Get Key!](#)

Routing Accessibility Areas

Travel mode:

Isochrone unit:

Maximum isochrone:  mins

Interval:  mins

Long: -1.702, Lat:48.119  
None  
None  
France

Use Layer

Point Layer

### OSM Tools

OpenRouteService routing and accessibility areas for QGIS

OSM Tools provides access to most of the functions of OpenRouteService.org, based on OpenStreetMap. The tool set includes most routing features and calculation of accessibility areas, both either interactive in the map canvas or from point files. Extensive attributes are set for output files, incl. duration, length and start/end locations.

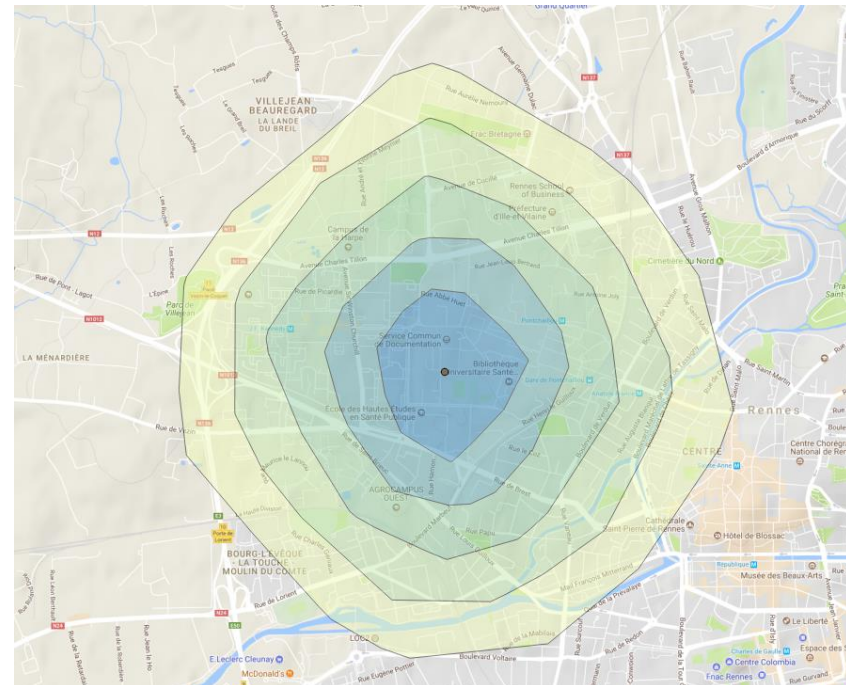
★★★★★ 23 évaluation(s), 23681 téléchargements

Catégorie: Plugins  
Étiquettes: routing, OSM, openstreetmap, openrouteservice, service area, accessibility area, route  
Plus d'infos: [Page d'accueil](#) [bug tracker](#) [code repository](#)

Auteur: [Nils Nolde](#)

Version installée : 0.3.4 (dans C:\Users\mericakay\_B\gis2\python\plugins\OSMTools)  
Version disponible : 0.3.4 (dans Dépôt officiel des extensions QGIS)

Journal des modifications :  
2017/10/07 v.0.3.4 Fixed zoomToLayer bug (#35)  
2017/09/24 v.0.3.3 Handled deprecated mapRenderer() function  
2017/09/23 v.0.3.2 Added avoid\_type feature for routing, auto-





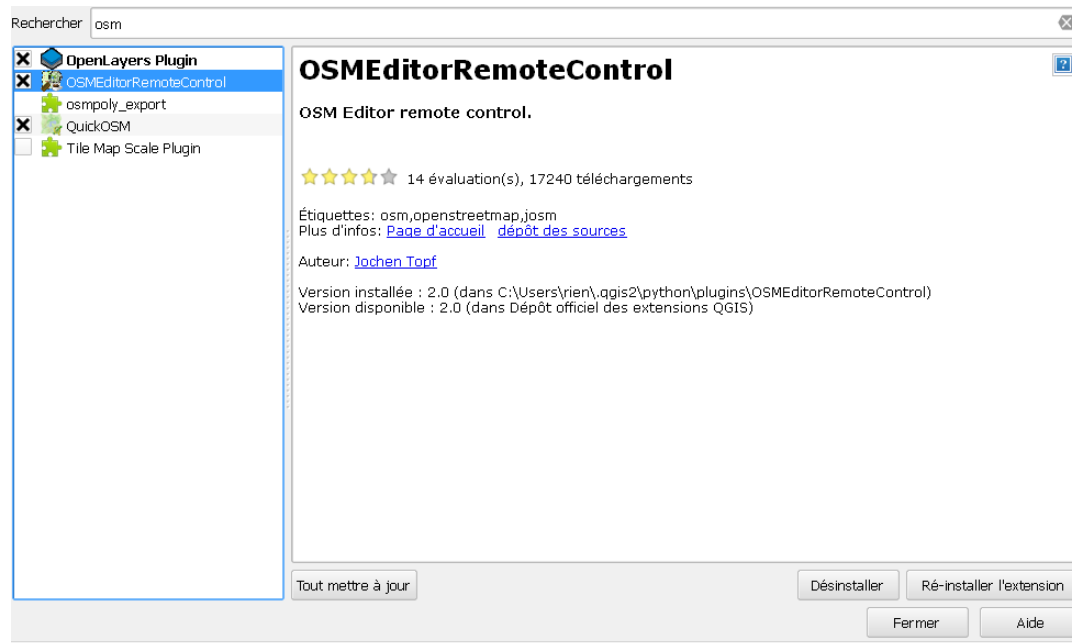
# OSMEditorRemoteControl



*Extension de QGIS*

→ Récupérer des données issue de l'emprise de la carte

→ Récupérer des données issue de l'emprise d'une couche





# OSMEditorRemoteControl



Menu Vecteur → OpenstreetMap → Télécharger données

Télécharger des données Open... ? X

Emprise

Depuis le canevas de la carte

Depuis la couche selection

Manuel

6.12517e+06

-188509 -185759

6.12368e+06

Fichier en sortie

...

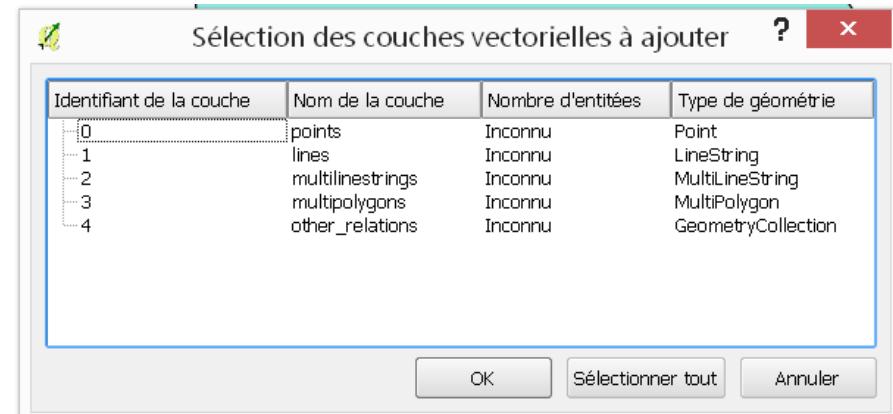
OK Fermer



# OSMEditorRemoteControl



- Importer la base de données .osm
- La base de données contient 5 tables
  - Points
  - Lines
  - Multilinestrings
  - Multipolygons
  - Other\_relations

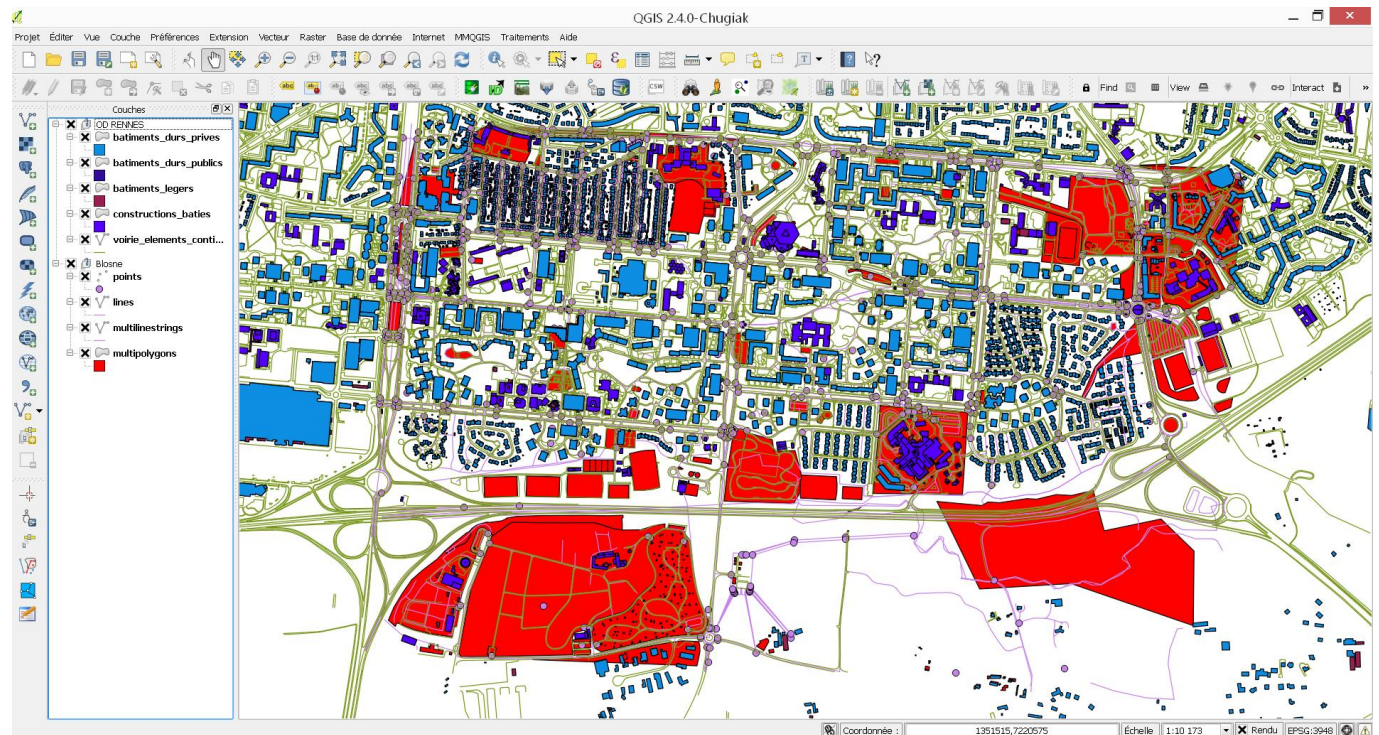


- Les tables attributaires contiennent plusieurs champs correspondant à certains des tags présents dans la base OSM (qualification des données ex. type de route, type de commerce, type de sol, etc.)



# Exercice 1

- Comparer les données d'OSM avec celle issues de l'IGN sur le **quartier du Blosne (Rennes)**



+

Quick OSM



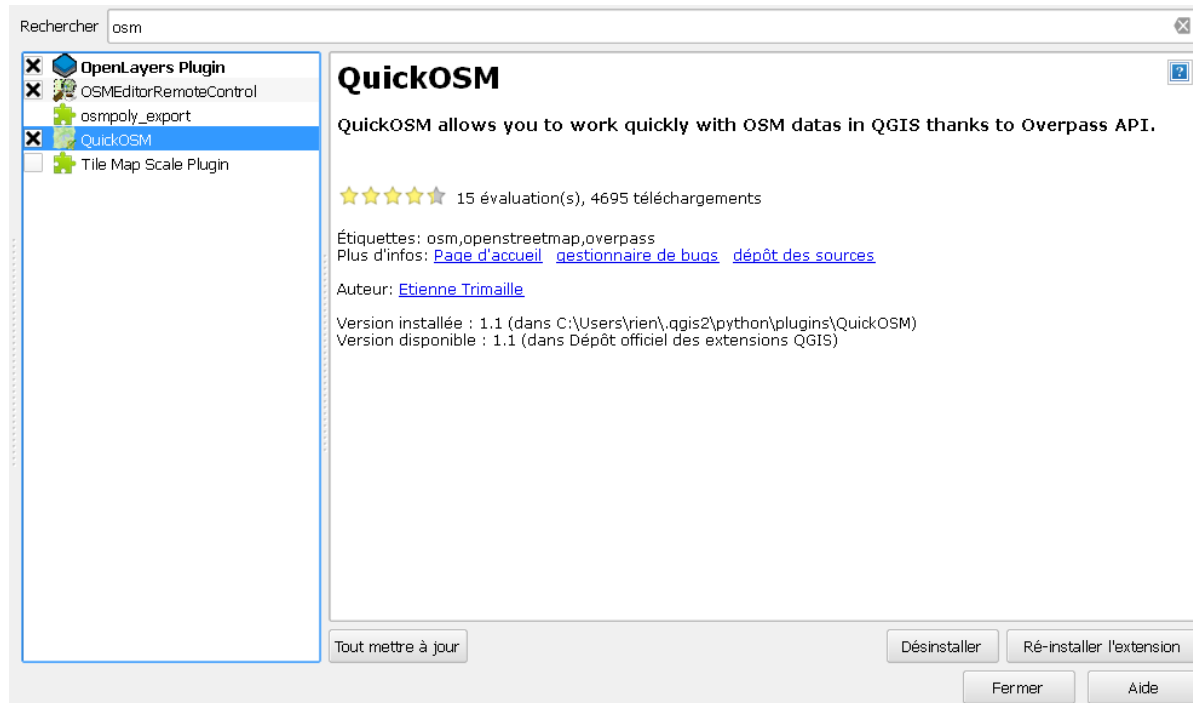




# Quick OSM



- Extension QGIS qui permet de récupérer des données selon des requêtes (avec des critères) et selon une entrée géographique comme une ville par exemple
- Basée sur l'API Overpass

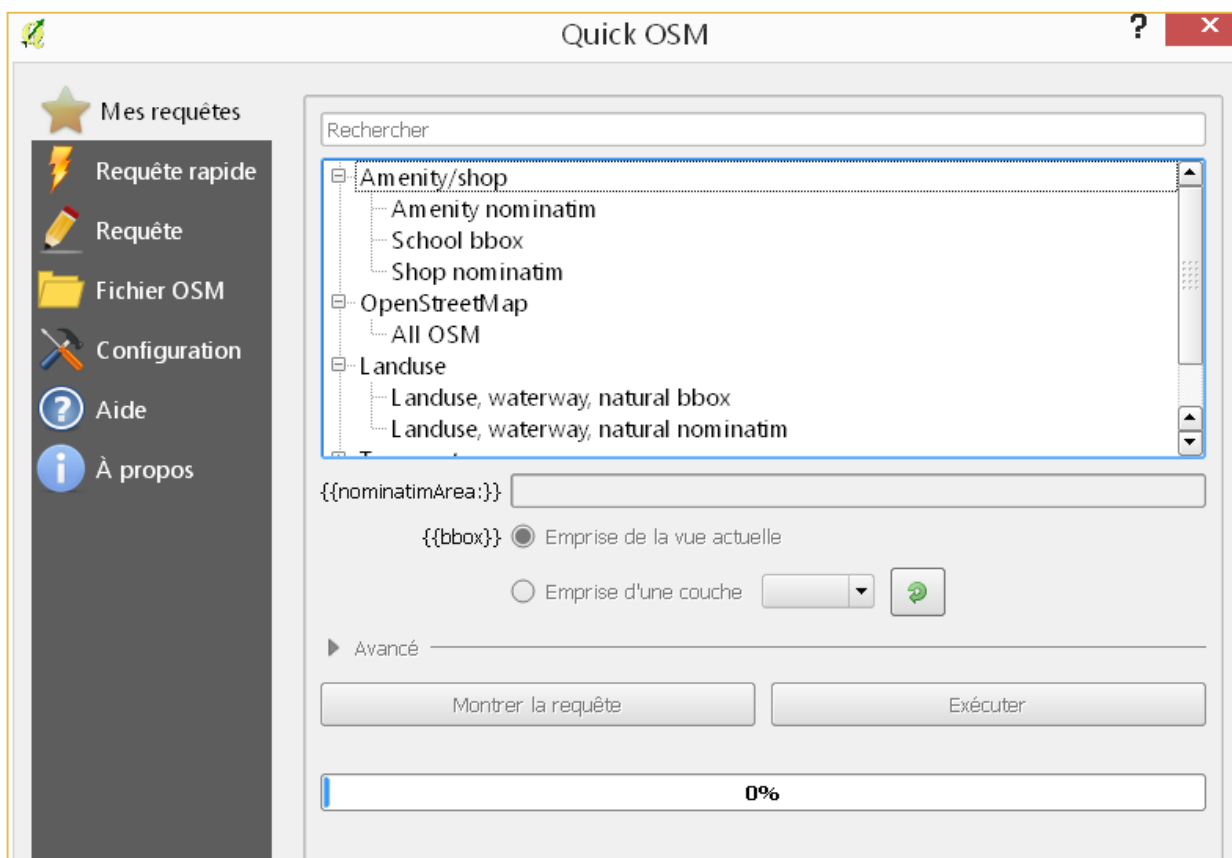




# Quick OSM



"Mes requêtes" = requêtes pré-enregistrées



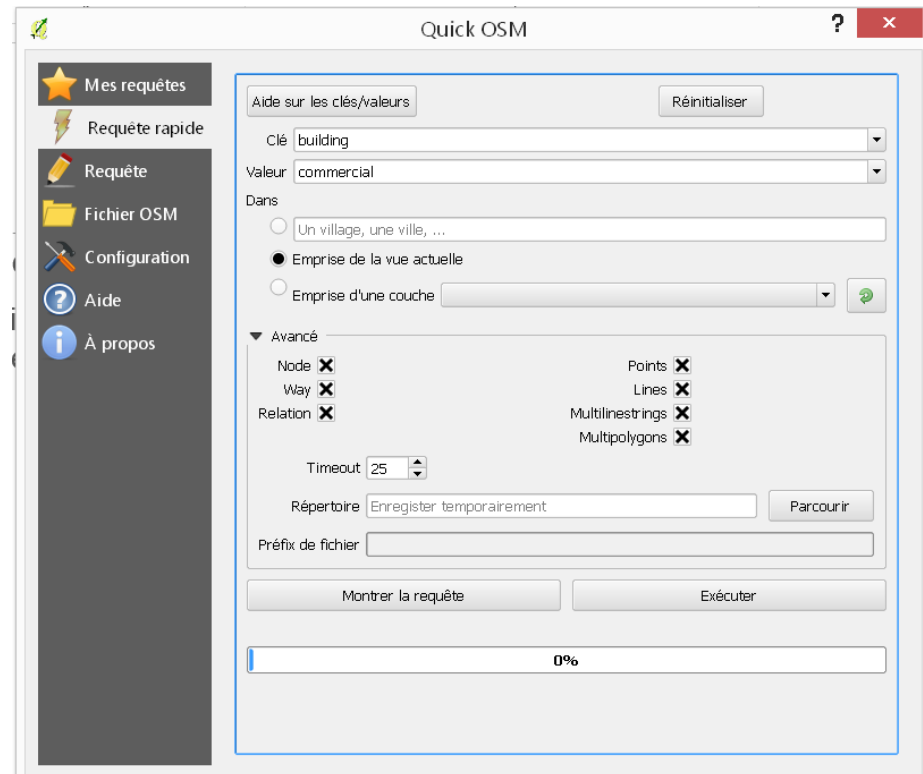


# Quick OSM



Requête rapide permet de requêter une clé et une valeur

- Possibilité de filtrer les informations selon une entrée thématique (type de route, type de bâtiment, type de magasin, etc.)

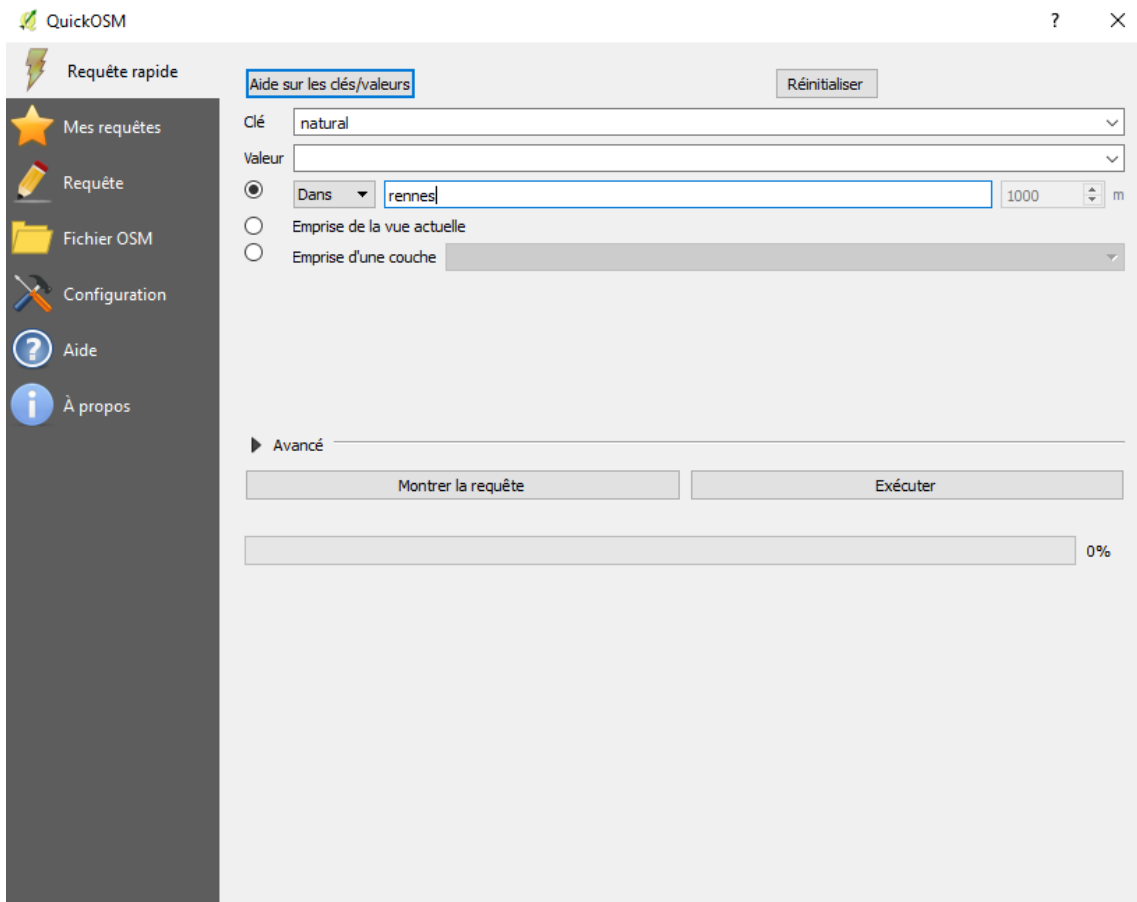




# QuickOSM



Télécharger les données avec la clef « natural » à Rennes





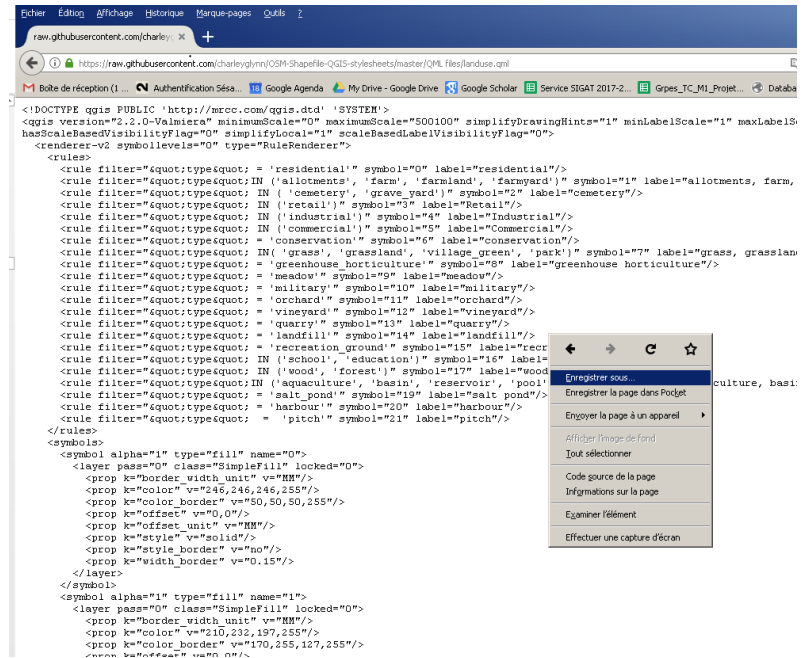
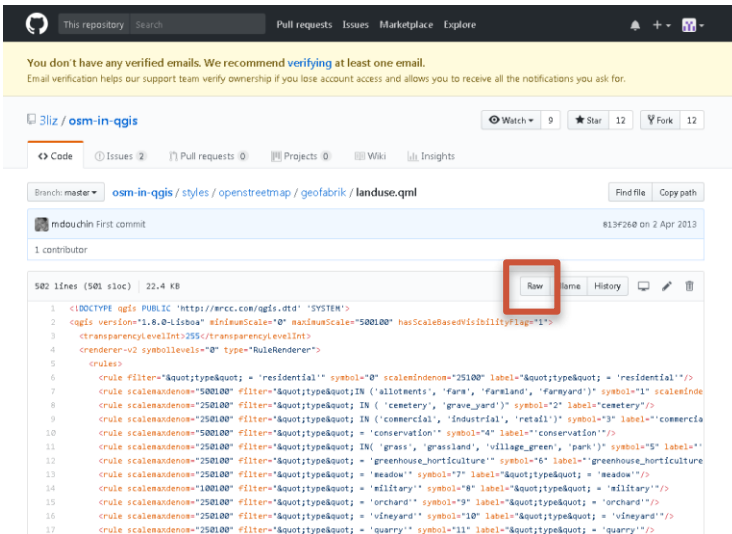
# Exercice 2

Appliquer un fichier de style pour mettre en forme la couche *landuse* issue d'OSM

- Télécharger un modèle .qml

<https://github.com/3liz/osm-in-qgis/blob/master/styles/openstreetmap/geofabrik/landuse.qml>

- Ouvrir la version brute (cliquer sur *Raw*)
- Clic droit → Enregistrer sous

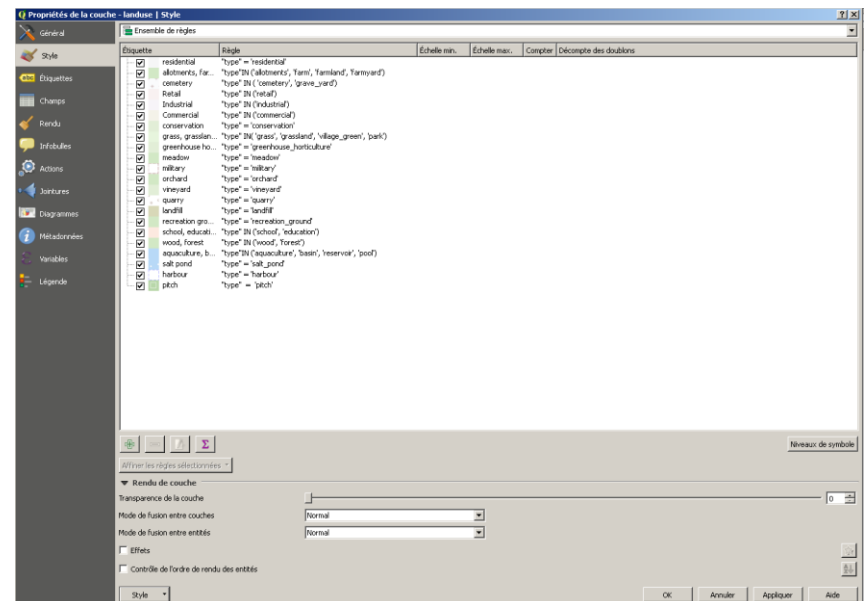
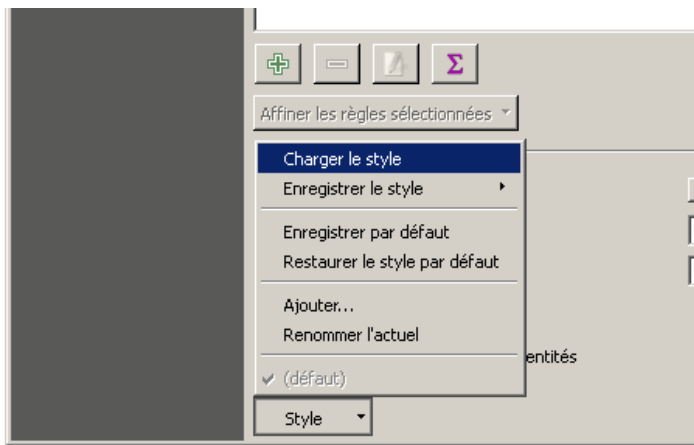




# Exercice 2

Appliquer un fichier de style pour mettre en forme la couche *landuse* issue d'OSM

- Dans les propriétés de style de la couche, charger le modèle .qml

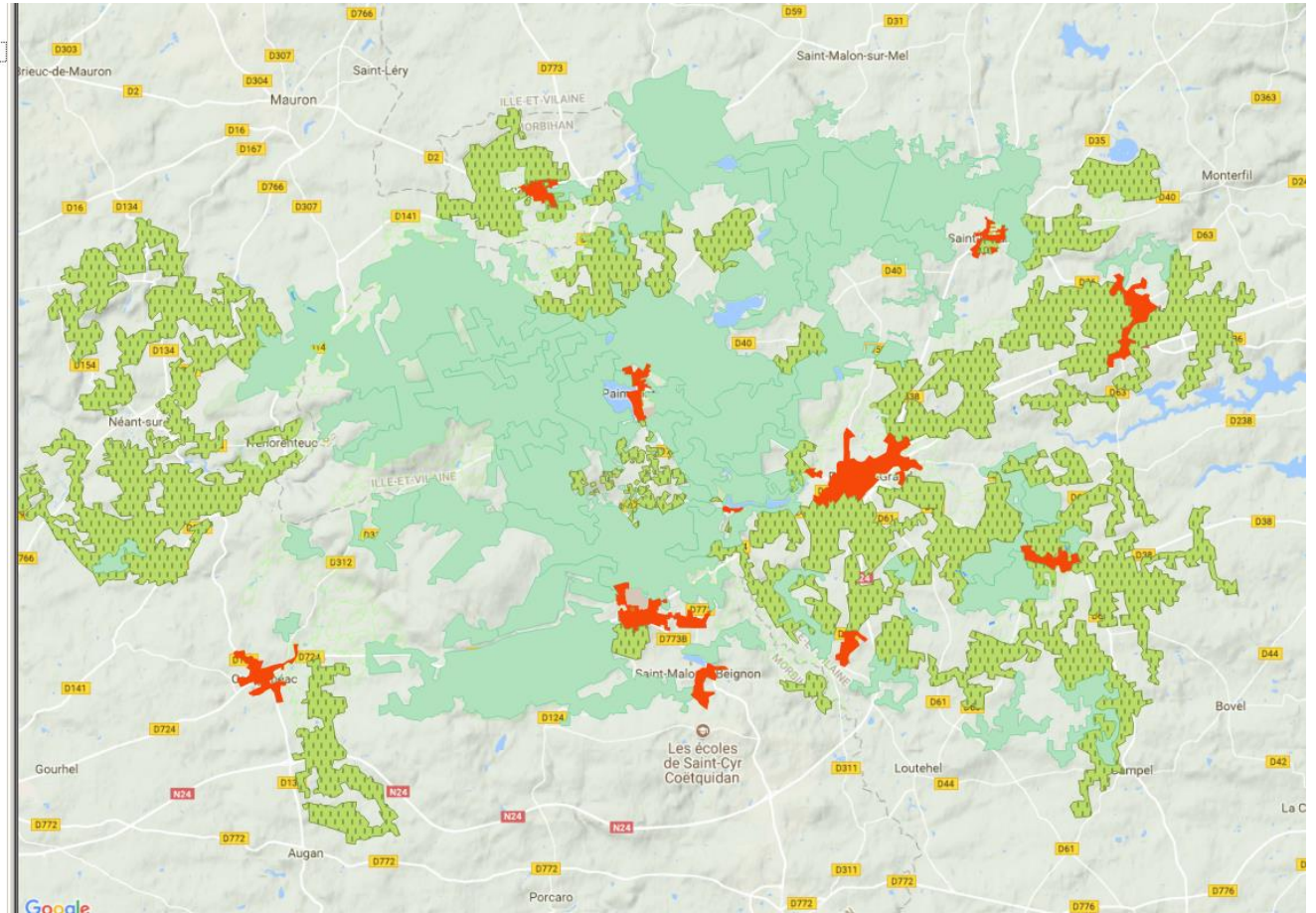




# Exercice 2



- landuse**
- "type" = 'residential'
- 'allotments', 'farm', 'farmland', 'farmyard'
- cemetery
- 'commercial', 'industrial', 'retail'
- 'conservation'
- 'grass', 'grassland', 'village\_green', 'park'
- 'greenhouse\_horticulture'
- "type" = 'meadow'
- "type" = 'military'
- "type" = 'orchard'
- "type" = 'vineyard'
- "type" = 'quarry'
- "type" = 'landfill'
- 'recreation\_ground'
- 'school', 'education'
- 'wood', 'forest'
- aquaculture, basin, reservoir, pool
- "type" = 'salt\_pond'
- "type" = 'harbour'
- "type" = 'railway'
- 'brownfield', 'construction', 'greenfield'
- "type" = 'garages'
- ZONE\_OCCUPATION\_SOL**
- Google Physical**





# Exercice 2



Comparer avec les données issues de CORINE LAND COVER

<http://www.statistiques.developpement-durable.gouv.fr/clc/fichiers/>

## Téléchargement des données CORINE Land Cover

Avant de procéder au téléchargement, veuillez indiquer pour quel(s) secteur(s) économique(s) vous avez l'intention d'utiliser ces données :

Secteurs économiques : Agriculture

Votre demande concerne :

Métropole  La Réunion  Mayotte  Guadeloupe  Martinique  Guyane

Pour chacune des zones souhaitées, veuillez compléter l'onglet correspondant afin de choisir les produits que vous souhaitez télécharger.

Métropole La Réunion Mayotte Guadeloupe Martinique Guyane

Métropole entière :  Oui  Non

Régions :

Départements : Ile-et-Vilaine

Campagne	Produits	Formats
2012	<input checked="" type="checkbox"/> CORINE Land Cover 2012	<input checked="" type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID <input type="checkbox"/> GeoTiff
2012	<input type="checkbox"/> CORINE Land Cover Changements 2006-2012	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID
2012	<input type="checkbox"/> CORINE Land Cover 2006 révisée	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID <input type="checkbox"/> GeoTiff
2006	<input type="checkbox"/> CORINE Land Cover 2006	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID <input type="checkbox"/> GeoTiff
2006	<input type="checkbox"/> CORINE Land Cover Changements 2000-2006	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID
2006	<input type="checkbox"/> CORINE Land Cover 2000 révisée	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID <input type="checkbox"/> GeoTiff
2000	<input type="checkbox"/> CORINE Land Cover 2000	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID <input type="checkbox"/> GeoTiff
2000	<input type="checkbox"/> CORINE Land Cover Changements 1990-2000	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID
1990	<input type="checkbox"/> CORINE Land Cover 1990	<input type="checkbox"/> Shapefile <input type="checkbox"/> MIF/MID <input type="checkbox"/> GeoTiff

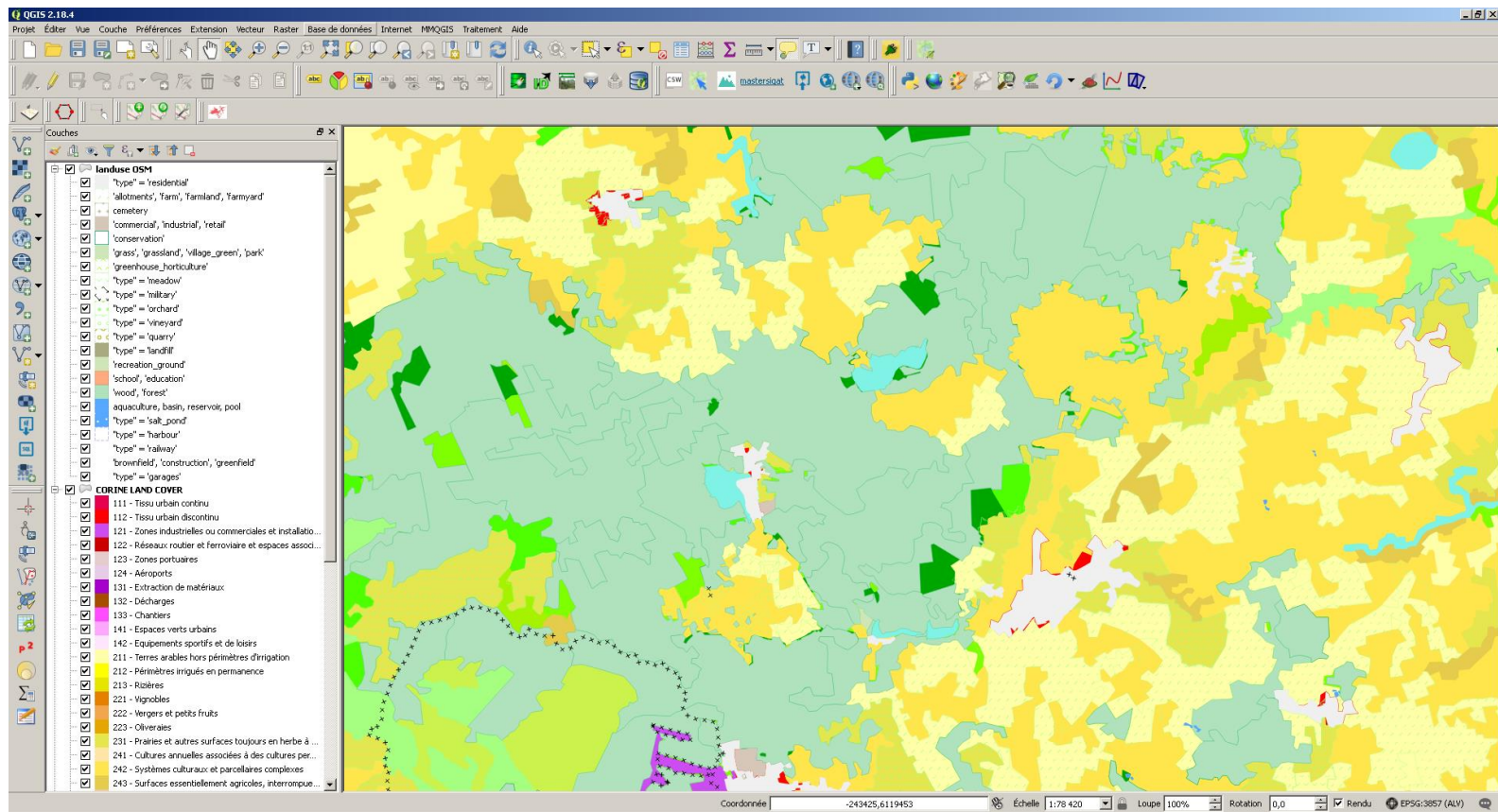




# Exercice 2



Appliquer le fichier de style SLD fourni et comparer les 3 jeux de données

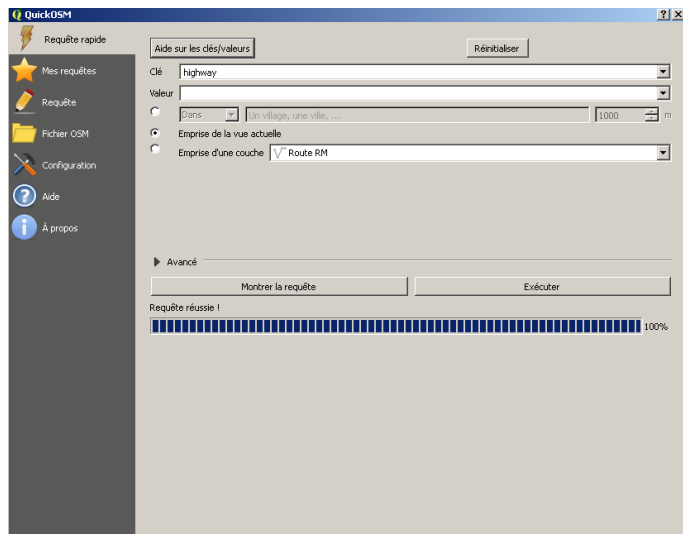




# Exercice 3



Comparer le réseau routier de la BDTOPO et les données routières issues d'OSM (key=highway) à Villejean





# Exercice 3



Comparer également avec la base de données de Rennes Métropole

<https://data.rennesmetropole.fr/explore/dataset/troncons-de-voies-du-referentiel-voies-et-adresses-de-rennes-metropole/>

## Tronçons de voies du référentiel voies et adresses de Rennes Métropole

📄 Informations    📊 Tableau    🗺️ Carte    📄 Analyse    📄 Export    🔌 API    💬 Commentaires (0)

Rennes Métropole dispose depuis 2013 d'un Référentiel Voies et Adresses (RVA) sur les communes de l'agglomération.

Lignes représentant chaque tronçon des voies.

Un tronçon appartient à au moins une voie.

Un tronçon appartient à une ou deux communes.

Un tronçon représente, le plus souvent, le centre de la chaussée.

Les tronçons de voies sont topologiques : les extrémités d'un tronçon correspondent à des intersections ou des jonctions, sauf dans le cas d'un chevauchement (cf paragraphe suivant).

Les tronçons gèrent les cas de chevauchement grâce à l'attribut « Niveau ». Cet attribut permet donc de différencier le tronçon passant dessus et le tronçon passant dessous.

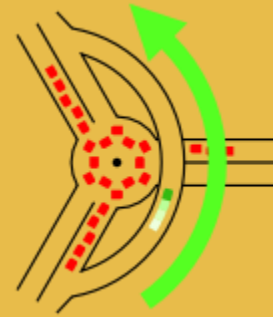


# Exercice 3



The screenshot displays a GIS application window with the following components:

- explorateur:** A file explorer on the left side showing a directory structure with folders for drives (C:/ to Z:/) and various data sources like DB2, MSSQL, Oracle, PostGIS, Spatialite, ArcGISFeatureServer, ArcGISMapServer, OWS, Tile Server (XYZ), WCS, WFS, and WMS.
- couches:** A layer management panel on the left showing three checked layers: **ROUTE IGN** (represented by a red line), **Routes\_OSM** (represented by a blue line), and **Route RM** (represented by a cyan line).
- Map:** The main area showing a street network map with the three route layers overlaid. The ROUTE IGN layer is red, Routes\_OSM is blue, and Route RM is cyan.
- Footer:** A status bar at the bottom containing the text "de légende supprimées.", "Coordonnées -1.71825,48.11454", "Échelle 1:3 056", "Loupe 100%", "Rotation 0,0", "Rendu" (checked), and "EPSG:4326 (ALV)".



**Overpass**  
**API**

+

API Overpass



# API Overpass



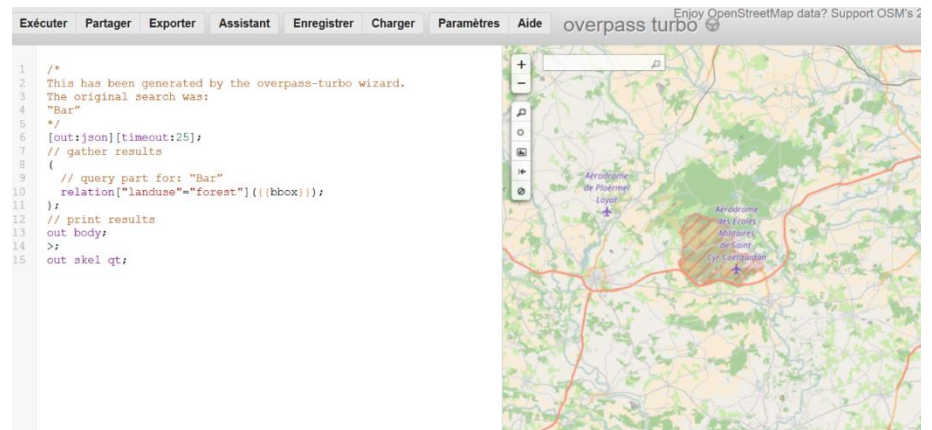
**API Overpass** permet d'interroger la base de données OSM

- Possibilité d'interrogation nombreuses
- Extraction des données massives et personnalisées

**Overpass turbo** est un outil internet d'exploration de données pour OpenStreetMap

- Ce site permet d'exécuter toutes sortes de requêtes de l' API Overpass et présente le résultat sur une carte interactive.

<https://overpass-turbo.eu/>



# + API Overpass



## Routes

- Documentation

<http://wiki.openstreetmap.org/wiki/Key:highway>

Key = **highway**

Value =

- motorway
- Trunk
- Primary
- Secondary
- ...

Key	Value	Element	Comment	Rendering	Photo
Roads					
These are the principal tags for the road network. They range from the most to least important.					
highway	motorway		A restricted access major divided highway, normally with 2 or more running lanes plus emergency hard shoulder. Equivalent to the Freeway, Autobahn, etc.		
highway	trunk		The most important roads in a country's system that aren't motorways. (Need not necessarily be a divided highway.)		
highway	primary		The next most important roads in a country's system. (Often link larger towns.)		
highway	secondary		The next most important roads in a country's system. (Often link towns.)		
highway	tertiary		The next most important roads in a country's system. (Often link smaller towns and villages)		
			The least most important through roads in a country's		



# API Overpass



## Extraire les routes selon une valeur (hierarchie)

- Les routes principales (trunk)

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

Type Key Value

```
2 This has been generated by the overpass-turbo wizard.  
3 The original search was:  
4  
5  
6  
7  
8 (   
9 // query part for: "route"  
10 way["highway"="trunk"] ({{bbox}});  
11 );  
12 // print results  
13 out body;  
14 >;  
15 out skel qt;
```

way highway trunk

Emprise ({{bbox}})





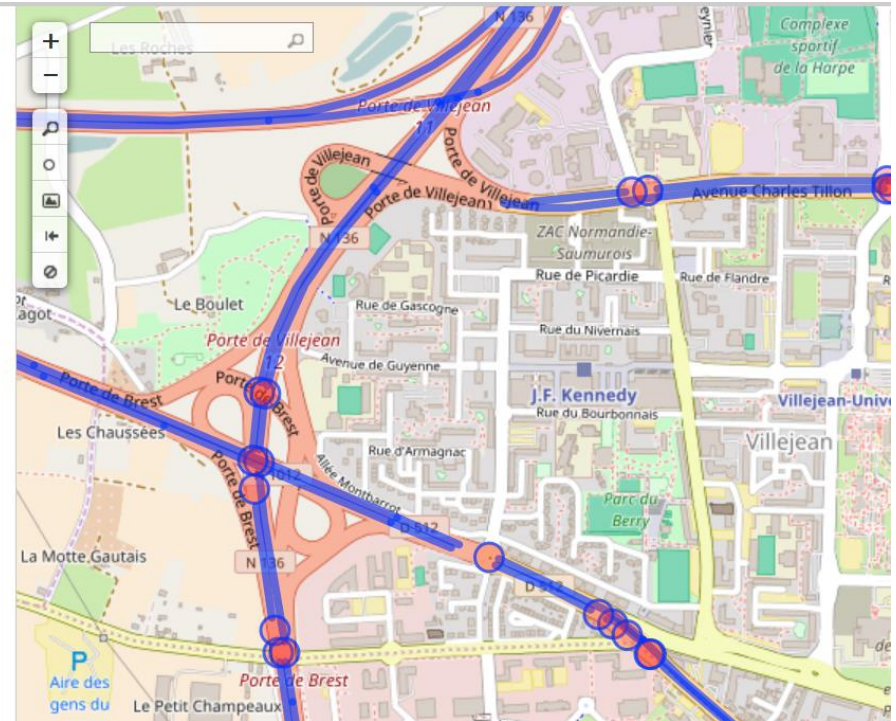
# API Overpass



## Combiner 2 critères

- Les grandes routes principales (*trunk*) et les routes principales (*primary*)

```
1  /*
2  This has been generated by the overpass-turbo wizard.
3  The original search was:
4  "route"
5  */
6  [out:json][timeout:25];
7  // gather results
8
9  // query part for: "route"
10 way["highway"="trunk"]>{{bbox}};
11 way["highway"="primary"]>{{bbox}};
12
13 );
14 // print results
15 out body;
16 >;
17 out skel qt;
```





# API Overpass



## Ajouter un critère de vitesse

- Les routes limitées à 30kmh (*maxspeed*)
- <http://wiki.openstreetmap.org/wiki/Key:maxspeed>

Enjoy OpenStreetMap data? Support OSM's 2016 donation d

overpass turbo

```
1 /*
2 This has been generated by the overpass-turbo wizard.
3 The original search was:
4 "route"
5 */
6 [out:json][timeout:25];
7 // gather results
8
9 // query part for: "route"
10 way["highway"]["maxspeed"="30"]>{{bbox}};
11
12 };
13 // print results
14 out body;
15 >;
16 out skel qt;
```

Chargé - nœuds: 400, cf Affiché - points d'intérêt - POIs: 0, li



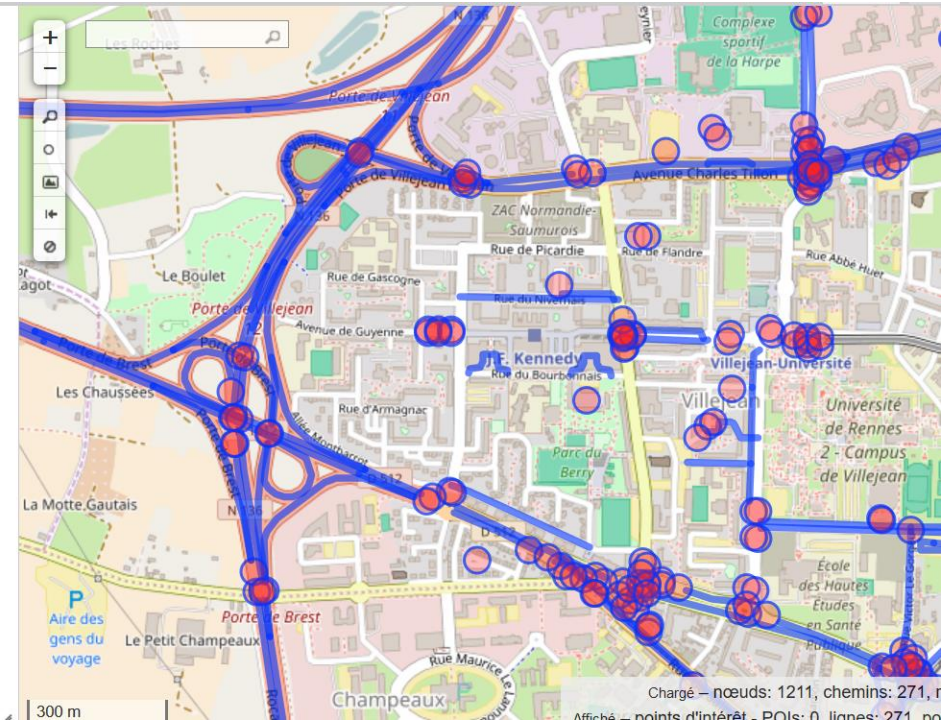
# API Overpass



## Ajouter un critère de direction

- Les routes en sens unique (*oneway*)
- <http://wiki.openstreetmap.org/wiki/Key:oneway>

```
1 /*
2 This has been generated by the overpass-turbo wizard.
3 The original search was:
4 "route"
5 */
6 [out:json][timeout:25];
7 // gather results
8
9 // query part for: "route"
10 way["highway"]["oneway"="yes"]>{{bbox}};
11
12 }}
13 // print results
14 out body;
15 >;
16 out skel qt;
```



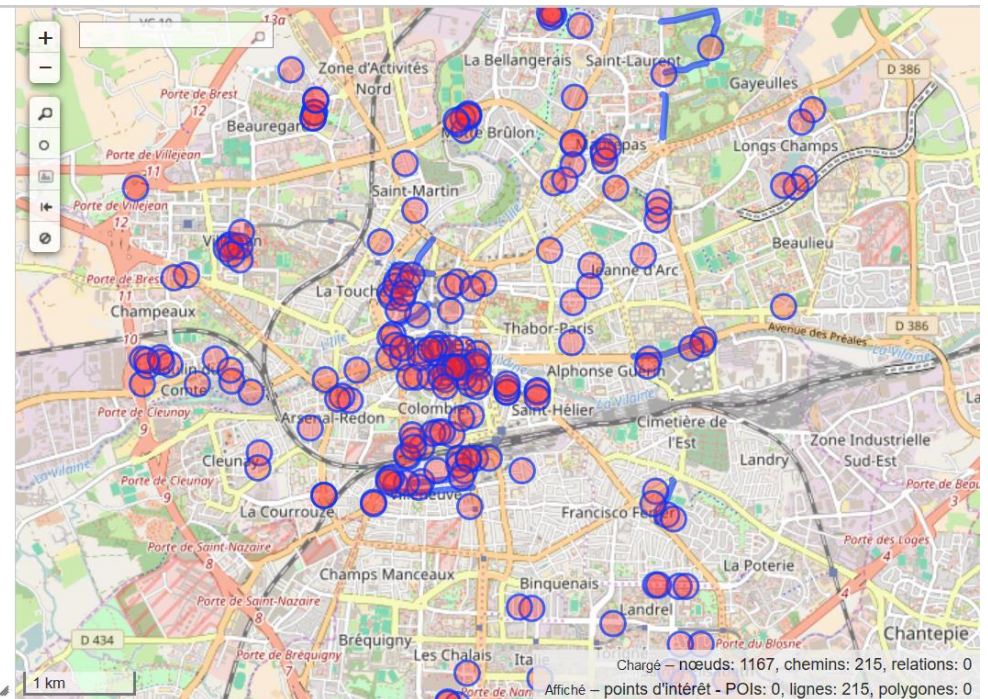


# API Overpass

## Choisir la zone d'interrogation (emprise de la requête)

- Récupérer les **routes** à **sens uniques** et **limitées à 30km/h** à **Rennes**

```
1  /*
2  This has been generated by the overpass-turbo wizard.
3  The original search was:
4  "highway=trunk in Rennes"
5  */
6  [out:json][timeout:25];
7  // fetch area "Rennes" to search in
8  {{geocodeArea:Rennes}}->.searchArea;
9  // gather results
10 (
11   // query part for: "highway=trunk"
12   way["highway"]["maxspeed"="30"]["oneway"="yes"]
13   (area.searchArea);
14 );
15 // print results
16 out body;
17 >;
18 out skel qt;
```





# API Overpass



## Amenity

- Documentation

<http://wiki.openstreetmap.org/wiki/Key:amenity>

Key = **highway**

Value =

- bar
- recycling
- bench
- school
- ...

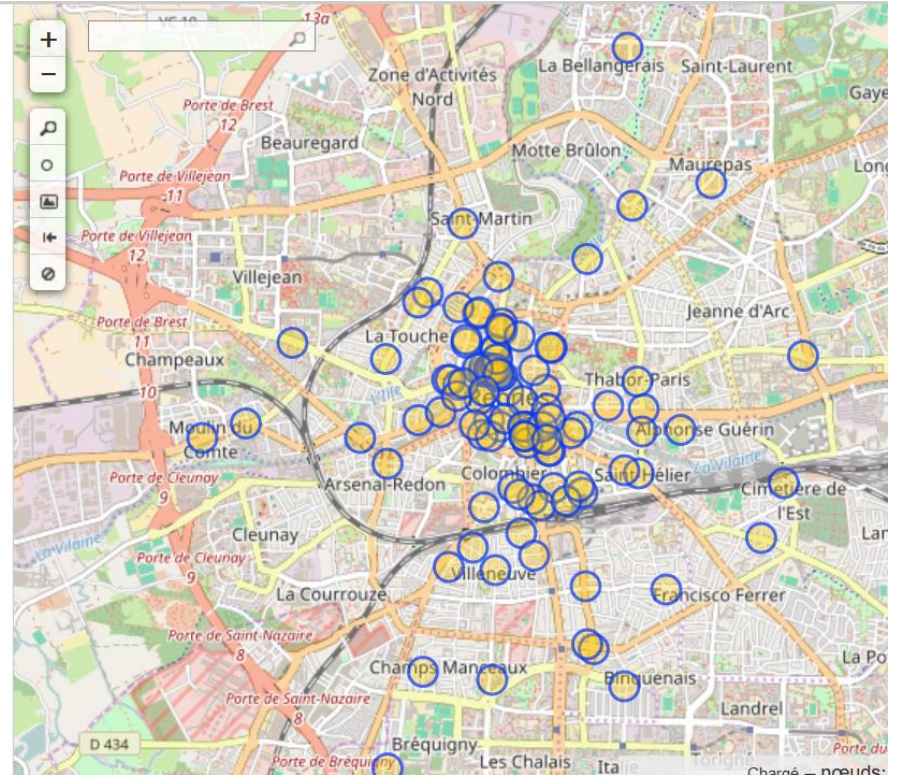
Key	Value	Element	Comment	Rendering	Photo
<b>Sustenance</b>					
amenity	bar		Bar is a purpose-built commercial establishment that sells alcoholic drinks to be consumed on the premises. They are characterised by a noisy and vibrant atmosphere, similar to a party and usually don't sell food. See also the description of the tags <code>amenity=pub;bar;restaurant</code> for a distinction between these.		
amenity	bbq		BBQ or Barbecue is a permanently built grill for cooking food, which is most typically used outdoors by the public. For example these may be found in city parks or at beaches. Use the tag <code>fuel=*</code> to specify the source of heating, such as <code>fuel=wood;electric;charcoal</code> . For mapping nearby table and chairs, see also the tag <code>tourism=picnic_site</code> . For mapping campfires and firepits, instead use the tag <code>leisure=firepit</code> .		
amenity	biergarten		Biergarten or beer garden is an open-air area where alcoholic beverages along with food is prepared and served. See also the description of the tags <code>amenity=pub;bar;restaurant</code> . A biergarten can commonly be found attached to a beer hall, pub, bar, or restaurant. In this case, you can use <code>biergarten=yes</code> additional to <code>amenity=pub;bar;restaurant</code> .		
amenity	cafe		Cafe is generally an informal place that offers casual meals and beverages; typically, the focus is on coffee or tea. Also known as a <code>coffeehouse/shop</code> , <code>bistro</code> or <code>sidewalk cafe</code> . The kind of food served may be mapped with the tags <code>cuisine=*</code> and <code>diet=*</code> . See also the tags <code>amenity=restaurant;bar;fast_food</code> .		
amenity	drinking_water		Drinking water is a place where humans can obtain potable water for consumption. Typically, the water is used for only drinking. Also known as a <code>drinking fountain</code> or <code>bubbler</code> .		



# API Overpass

→ Extraire les bars

```
1  /*
2  This has been generated by the overpass-turbo wizard.
3  The original search was:
4  "Bar"
5  */
6  [out:json][timeout:25];
7  // gather results
8  (
9  // query part for: "Bar"
10  node["amenity"="bar"]({{bbox}});
11  );
12  // print results
13  out body;
14  >;
15  out skel qt;
```






# API Overpass




## Boundary

### ■ Documentation

<http://wiki.openstreetmap.org/wiki/Boundaries>

 **Feature : Boundaries**



**Description**

Boundaries mark the borders of areas, mostly political, but also of other administrative areas.

**Tags**

`boundary=*`

### Other

#### `boundary=maritime`

for marking maritime borders (rather than land areas normally assumed by `boundary=political`)

#### `boundary=political`

is approved, should be documented in each country where they are used. Can be used

#### `boundary=vice_county`

for marking vice counties in Britain and Ireland [↗](#).

#### `boundary=national_park`

marks the borders of a national park.

#### `boundary=protected_area`

a more recently introduced tag with a more verbose tagging scheme which can deal with

#### `boundary=religious_administration`

*trial* for dioceses, parishes... see [FrViPofm/Tag:boundary=religious\\_administration](#)

#### `boundary=national`

is approved, but not documented, can somebody check tagwatch for usage?

#### `boundary=civil`

is approved, but not documented, can somebody check tagwatch for usage?

#### `boundary=metropole`

*trial*: When metropolitan areas don't match with an administrative subdivision (some



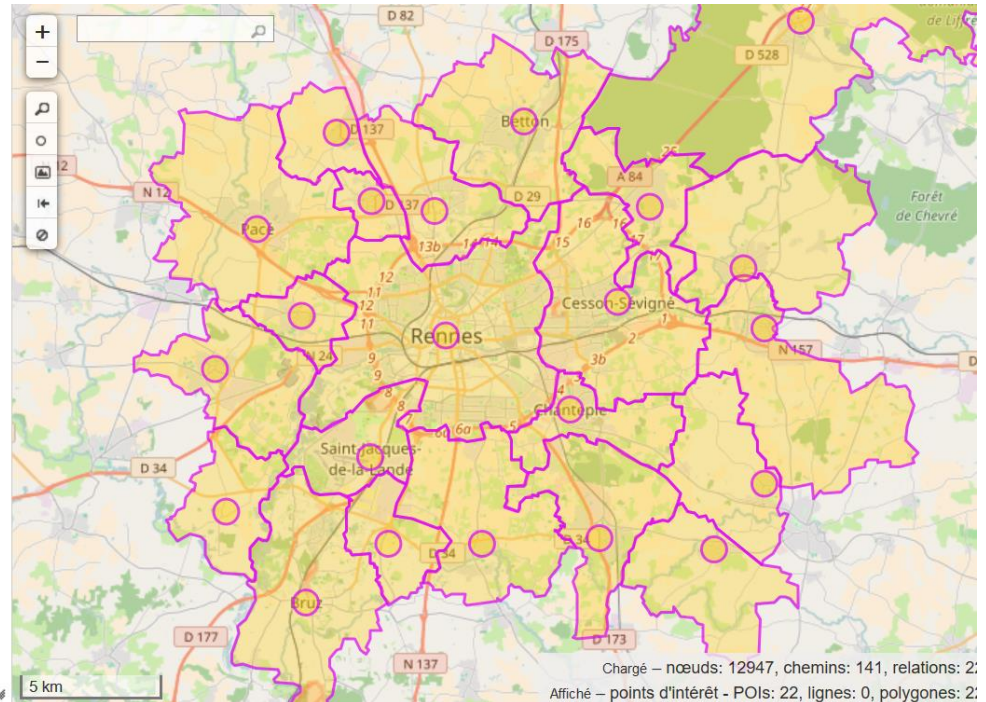
# API Overpass



## Extraire les communes adjacentes à Rennes

- Niveau communal en France ("8 ")
- <http://wiki.openstreetmap.org/wiki/Tag:boundary%3Dadministrative>

```
1 /*
2 This has been generated by the overpass-turbo wizard.
3 The original search was:
4 "Bar"
5 */
6 [out:json][timeout:25];
7 // gather results
8 (
9 // query part for: "Bar"
10 relation["boundary"="administrative"] ["admin_level" = "8"]
11 {{{bbox}}};
12 );
13 // print results
14 out body;
15 >;
16 out skel qt;
```






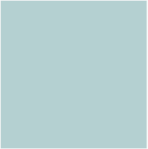








# + API Overpass



## Landuse

### ■ Documentation

<http://wiki.openstreetmap.org/wiki/FR:Key:landuse>

<b>landuse</b>	basin		<p>Zone d'eau artificielle de plusieurs types (infiltration, détention, rétention) qui finit par s'écouler dans une rivière.</p> <p>Utiliser avec <code>basin=*</code> pour les différents types.</p>		
<b>landuse</b>	brownfield		<p>Zone où des anciens bâtiments ont été rasés. La construction de nouveaux bâtiments est planifiée, mais pas encore en cours.</p>		
<b>landuse</b>	cemetery		<p>Cimetière. ajoutez <code>religion=*</code> s'il y a lieu (voir liste dans <code>amenity=place of worship</code>).</p> <p>Utiliser <code>amenity=grave_yard</code> pour les petites surfaces (à proximité d'une église par exemple)</p>		
					

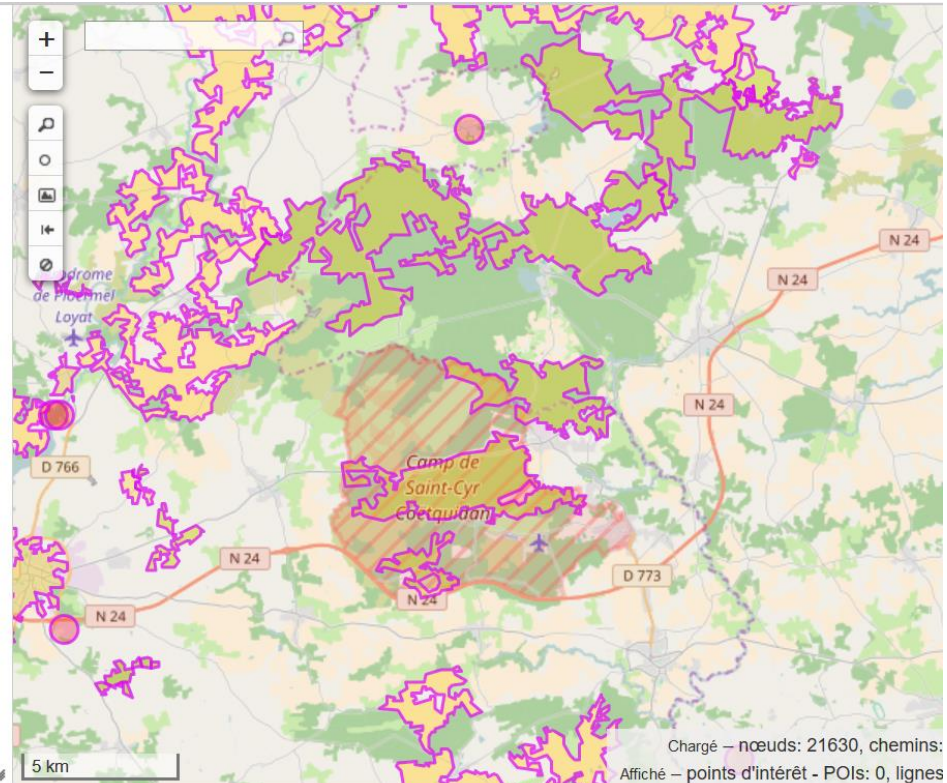


# API Overpass



Extraire les zones renseignées sur l'occupation des sols

```
1  /*
2  This has been generated by the overpass-turbo wizard.
3  The original search was:
4  "Bar"
5  */
6  [out:json][timeout:25];
7  // gather results
8  (
9
10 // query part for: "Bar"
11  relation["landuse"]>{{bbox}};
12 );
13 // print results
14 out body;
15 >;
16 out skel qt;
```



Chargé – nœuds: 21630, chemins:  
Affiché – points d'intérêt - POIs: 0, lignes

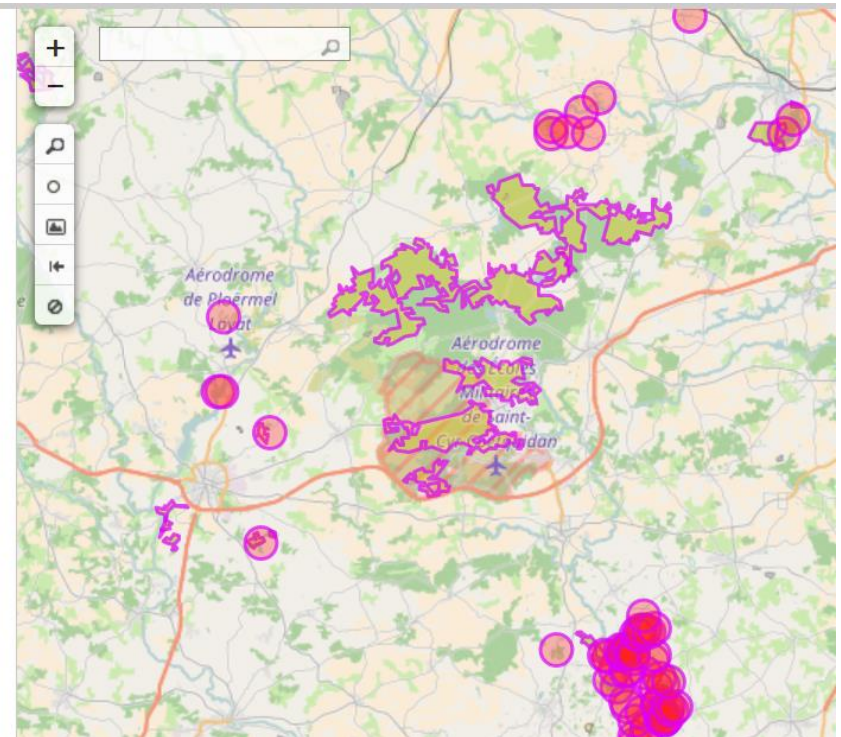


# API Overpass



Extraire les zones renseignées comme forêt

```
/*  
This has been generated by the overpass-turbo wizard.  
The original search was:  
"Bar"  
*/  
[out:json][timeout:25];  
// gather results  
(  
  // query part for: "Bar"  
  relation["landuse"="forest"]({{bbox}});  
);  
// print results  
out body;  
>;  
out skel qt;
```





# API Overpass

## Ajouter un critère de vitesse

- Toutes les routes avec une vitesse limite de plus de 50km/h

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

```
1 [out:json][timeout:25];
2 // gather results
3
4 (way[highway]
5   (if: (is_number(t[maxspeed]) && t[maxspeed] > 50))
6   ({{bbox}}));
7
8 // print results
9 out body;
10 >;
11 out skel qt;
```

**(way[highway]  
(if: (is\_number(t[maxspeed]) && t[maxspeed] > 50))  
({{bbox}}));  
);  
out body;  
>;  
out skel qt;**

Chargé — nœuds: 4333, chemins: 459, relations: 1  
Affiché — points d'intérêt - POIs: 0, lignes: 459, polygones: 1

<https://overpass-turbo.eu/index.html>



# API Overpass

## Ajouter un critère de nombre de voies

- Toutes les routes avec au moins deux voies (*lanes*)

```
1 /*
2 This is an example Overpass query.
3 Try it out by pressing the Run button above!
4 You can find more examples with the Load tool.
5 */
6
7 (way[highway]
8   (if: (is_number(t[lanes]) && t[lanes] > 1))
9   ({{bbox}});
10 );
11
12 out meta;
13 >>
14 out meta qt;
```

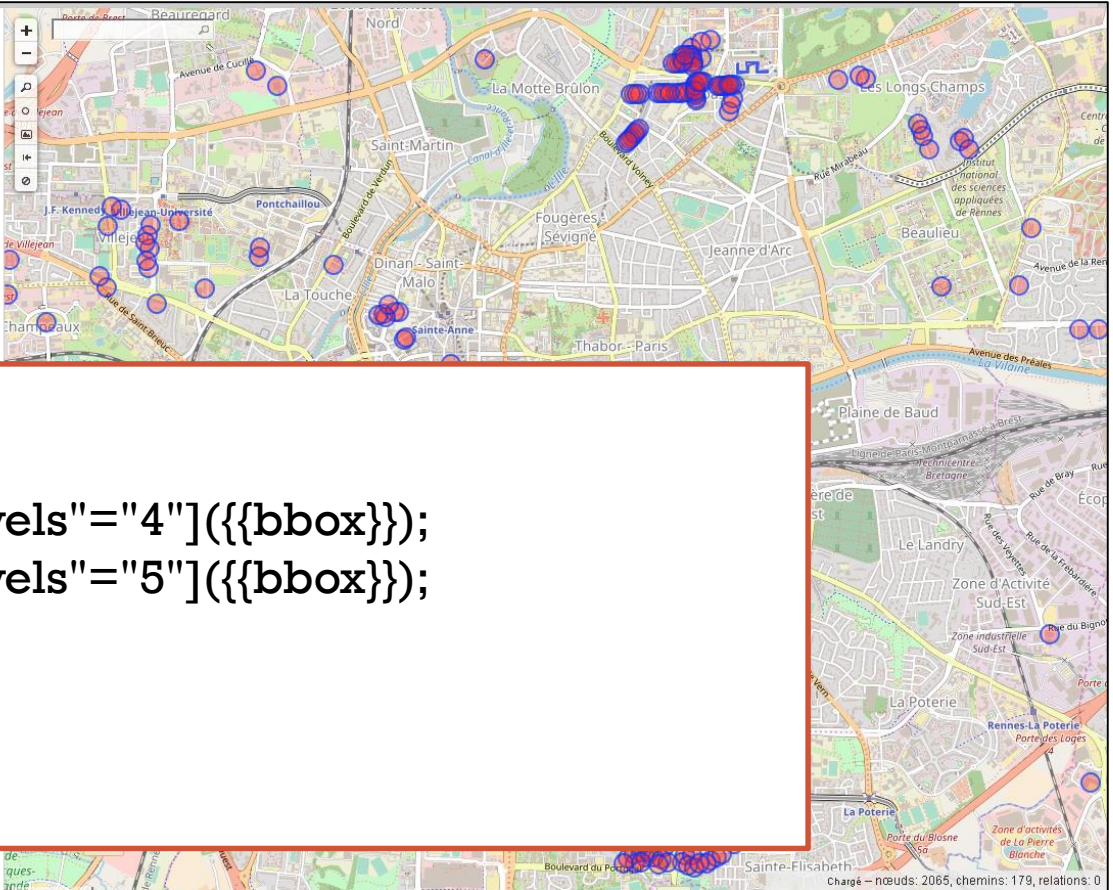
```
(way[highway]
  (if: (is_number(t[lanes]) && t[lanes] > 1)) ({{bbox}});
);
(way[highway]
  (if: (is_number(t[maxspeed]) && t[maxspeed] > 50)) ({{bbox}});
);
out body;
>>
out skel qt;
```



# + Sélection de hauteur

Sélectionner les bâtiments de 4 et 5 étages

```
[out:json][timeout:25];
(
  // query part for: "bar"
  way["building"]["building:levels"="4"]>{{bbox}};
  way["building"]["building:levels"="5"]>{{bbox}};
);
// print results
out body;
>;
out skel qt;
```



[out:json][timeout:25];  
(  
way["building"]["building:levels"="4"]>{{bbox}};  
way["building"]["building:levels"="5"]>{{bbox}};  
);  
out body;  
>;  
out skel qt;

# + Comptage d'entités



Afficher des statistiques sur les bâtiments de Rennes

```
[out:csv(::count, ::"count:nodes", ::"count:ways",
::"count:relations")][timeout:25];
{{geocodeArea:Rennes}}->.searchArea;
(
  node["building"="yes"](area.searchArea);
  way["building"="yes"](area.searchArea);
  relation["building"="yes"](area.searchArea);
);
out count;
```

```
1 @count @count:nodes @count:ways @count:relations
2 37821 6 37626 189
3
```

# + Comptage d'entités



## Afficher des statistiques sur les routes de Rennes

```
Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo
```

```
1 [out:csv(:"count", ::"count:nodes", ::"count:ways", ::"count:relations")][timeout:25];
2 {{geocodeArea:Rennes}}->.searchArea;
3 {
4   node["highway"] (area.searchArea);
5   way["highway"] (area.searchArea);
6   relation["highway"] (area.searchArea);
7 };
8 out count;
```

	@count	@count:nodes	@count:ways	@count:relations
1	23862	6792	17049	21
2				
3				

## Afficher des statistiques sur les bars de Rennes

```
1 [out:csv(:"count", ::"count:nodes")][timeout:25];
2 {{geocodeArea:Rennes}}->.searchArea;
3 {
4   node["amenity"="bar"] (area.searchArea);
5 };
6 out count;
```

	@count	@count:nodes
1	112	112
2		
3		





# Requête et style



## Sélectionner et représenter les types de bâtiments

```
[out:json][timeout:25];
```

```
( way["building" = "apartments"]({{bbox}});  
way["building" = "residential"]({{bbox}});  
way["building" = "house"]({{bbox}});  
way["building" = "school"]({{bbox}});
```

```
{{style:
```

```
way[building=apartments]  
{ color:blue; fill-color:blue; }
```

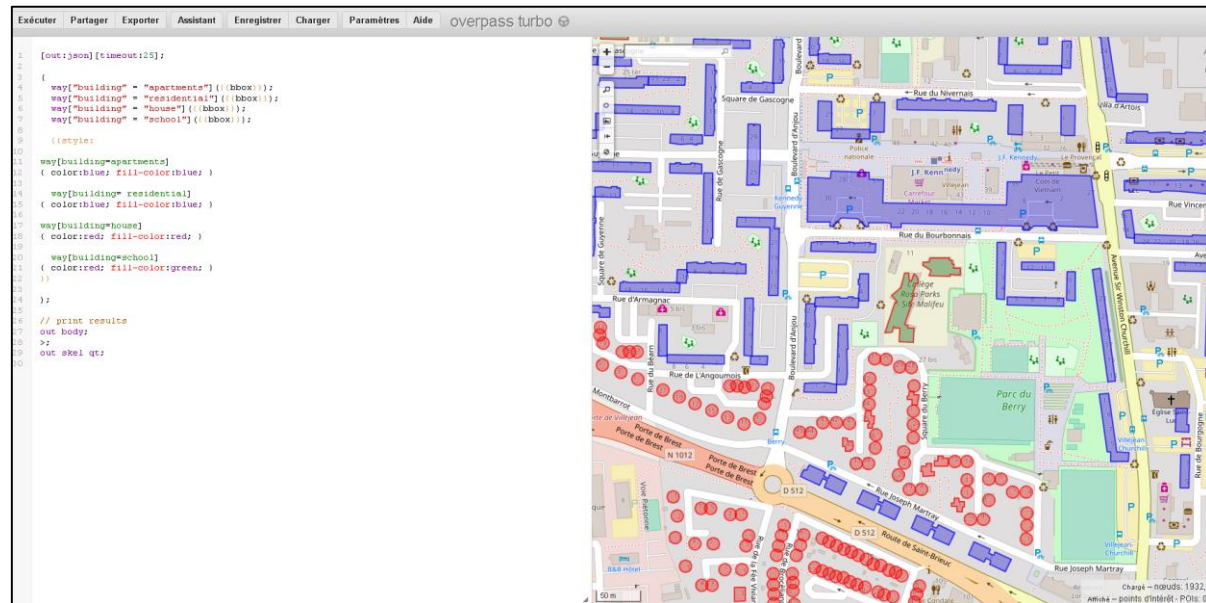
```
way[building=residential]  
{ color:blue; fill-color:blue; }
```

```
way[building=house]  
{ color:red; fill-color:red; }
```

```
way[building=school]  
{ color:red; fill-color:green; }  
}}
```

```
);
```

```
// print results  
out body;  
>;  
out skel qt;
```



# + Requête et style

Sélectionner et représenter les arrêts de bus, stations de métros et stations de vélos en libre service

```
[out:json][timeout:25];

{{geocodeArea:rennes}}->.searchArea;

( node["public_transport"="stop_position"] ["subway"="yes"]
(area.searchArea);
 node["highway"="bus_stop"](area.searchArea);
 node["amenity"="bicycle_rental"](area.searchArea);

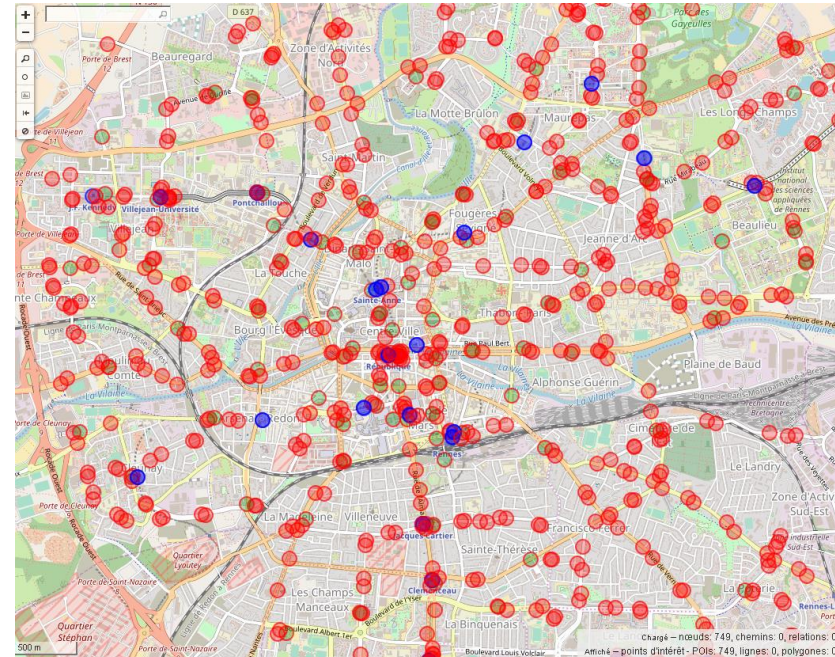
{{style:

node[public_transport=stop_position]
{ color:blue; fill-color:blue; }

node[highway=bus_stop]
{ color:red; fill-color:red; }

node[amenity=bicycle_rental]
{ color:red; fill-color:green; }
}}

);
out body;
>;
out skel qt;
```





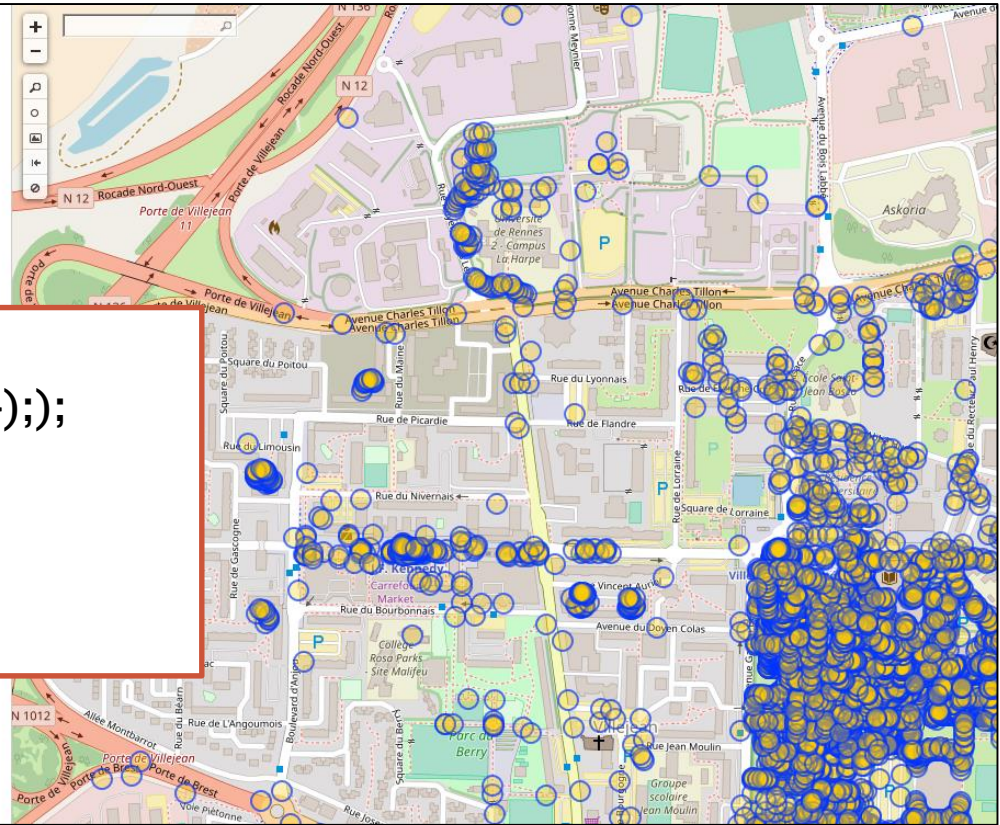
# Extraction par utilisateur



```
[out:json][timeout:25];  
(node(user:PanierAvide)({{bbox}}));  
  
out body;  
>;  
out skel qt;
```

```
[out:json][timeout:25];  
(node(user:PanierAvide)({{bbox}}));
```

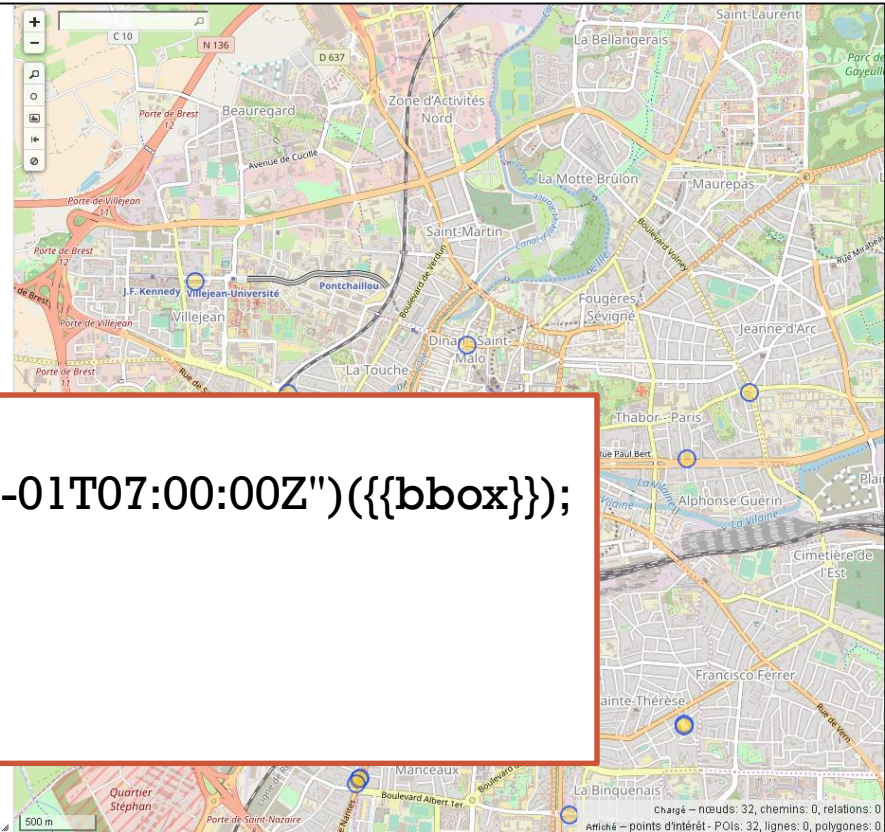
```
out body;  
>;  
out skel qt;
```



# + Extraction par date

Sélectionner les nouvelles aménités depuis le 1<sup>er</sup> novembre

```
1 [out:json][timeout:25];  
2 { node["amenity"](newer:"2017-11-01T07:00:00Z")({{bbox}});  
3 };  
4 out body;  
5 >;  
6 out skel qt;  
7 |
```

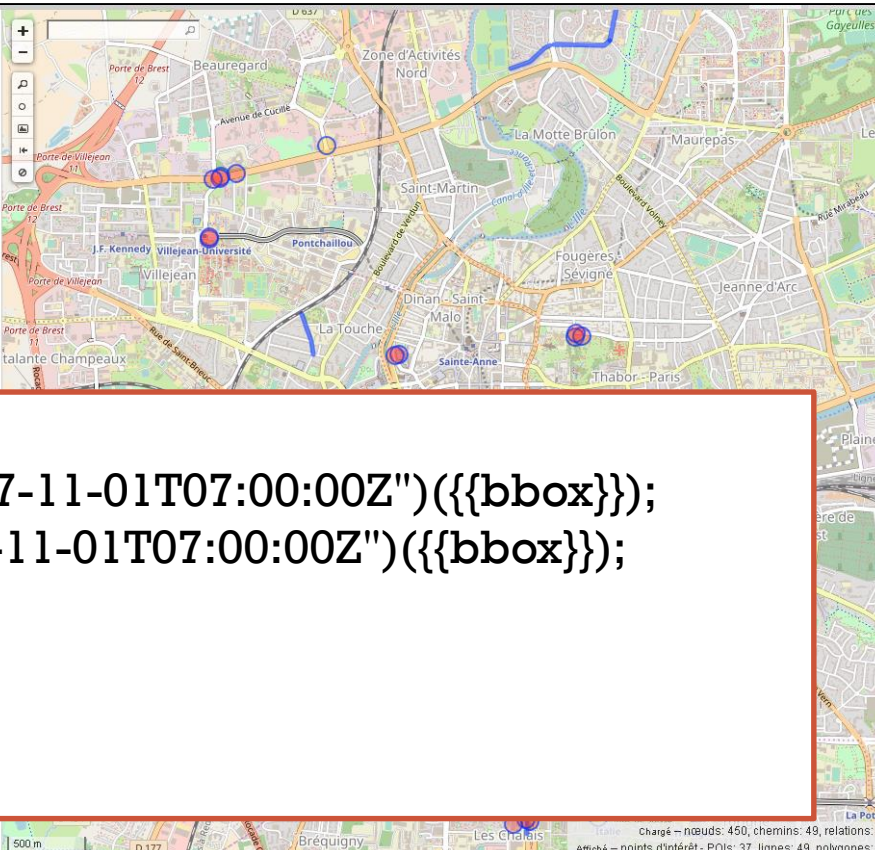


```
[out:json][timeout:25];  
( node["amenity"](newer:"2017-11-01T07:00:00Z")({{bbox}});  
);  
out body;  
>;  
out skel qt;
```

# + Extraction par date

Sélectionner les routes mises à jour depuis le 1<sup>er</sup> novembre

```
[out:json][timeout:25];
(
  node["highway"] (changed:"2017-11-01T07:00:00Z") ({{bbox}});
  way["highway"] (changed:"2017-11-01T07:00:00Z") ({{bbox}});
);
out body;
>;
out skel qt;
```



**[out:json][timeout:25];**  
**( node["highway"](changed:"2017-11-01T07:00:00Z")({{bbox}});**  
**way["highway"](changed:"2017-11-01T07:00:00Z")({{bbox}});**  
**);**  
**out body;**  
**>;**  
**out skel qt;**

Chargé — nœuds: 450, chemins: 49, relations: 0  
Amièhé — points d'intérêt - POIs: 37, lignes: 49, polygones: 0

# + Sélection spatiale

Sélectionner tous les bars à moins de 100m d'un arrêt de bus

```
1 [out:json][timeout:25];  
2   area[name="Rennes"];  
3   node(area)[highway=bus_stop];  
4   node(around:100)[amenity=bar];  
5   out;  
6 out body;  
7 >;  
8 out skel qt;  
9 |
```

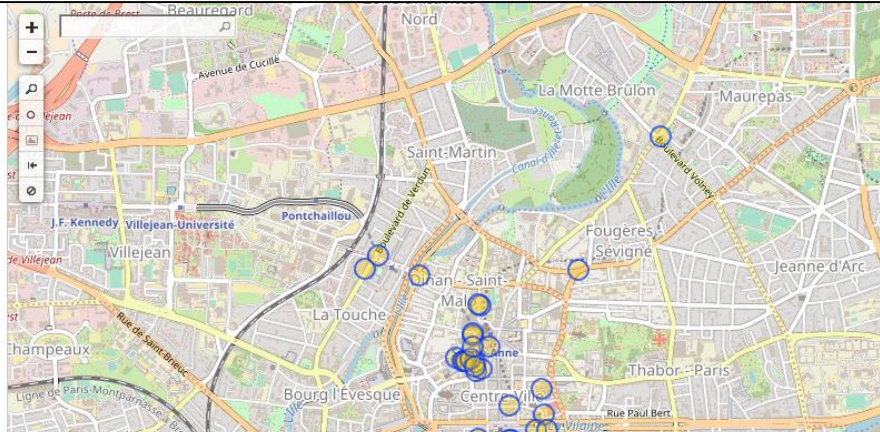
[out:json][timeout:25];  
area[name="Rennes"];  
node(area)[highway=bus\_stop];  
node(around:100)[amenity=bar];  
out;  
out body;  
>;  
out skel qt;

Chargé - nœuds: 176, chemins: 0, relations: 0  
Affiché - points d'intérêt - POIs: 88, lignes: 0, polygones: 0

# + Sélection spatiale

Sélectionner tous les bars à moins de 200m d'une station de métro

```
1 [out:json][timeout:25];  
2   area[name="Rennes"];  
3   node(area)["public_transport"="stop_position"] ["subway"="yes"] ;  
4   node(around:200)[amenity=bar];  
5   out;  
6 out body;  
7 >;  
8 out skel qt;  
9
```



```
[out:json][timeout:25];  
area[name="Rennes"];  
node(area)["public_transport"="stop_position"] ["subway"="yes"] ;  
node(around:200)[amenity=bar];  
out;  
out body;  
>;  
out skel qt;
```

# + Sélection spatiale

Sélectionner les arrêts de bus à moins de 200m d'une station de métro

The screenshot displays the QGIS interface with the following elements:

- Top Bar:** Includes buttons for 'Exécuter', 'Partager', 'Exporter', 'Assistant', 'Enregistrer', 'Charger', 'Paramètres', 'Aide', and 'overpass turbo'. On the right, there are tabs for 'Carte' and 'Données', and a user profile 'Flattr this!'.
- Console (Left):** Contains a SQL query:

```
1 [out:json][timeout:25];
2 area[name="Rennes"];
3 node(area)["public_transport"="stop_position"] ["subway"="yes"];
4 node(around:200)["highway"="bus_stop"];
5 out;
6 out body;
7 >;
8 out sskel qt;
9
```
- Map (Center):** Shows a map of Rennes, France, with numerous bus stops highlighted by blue circles. The map includes street names, district names, and a scale bar (500m).
- Bottom Right:** A status bar indicates 'Chargé - nœuds: 280, chemins: 0, relations: 0' and 'Affiché - points d'intérêt: POIs: 140, lignes: 0, polygones: 0'.

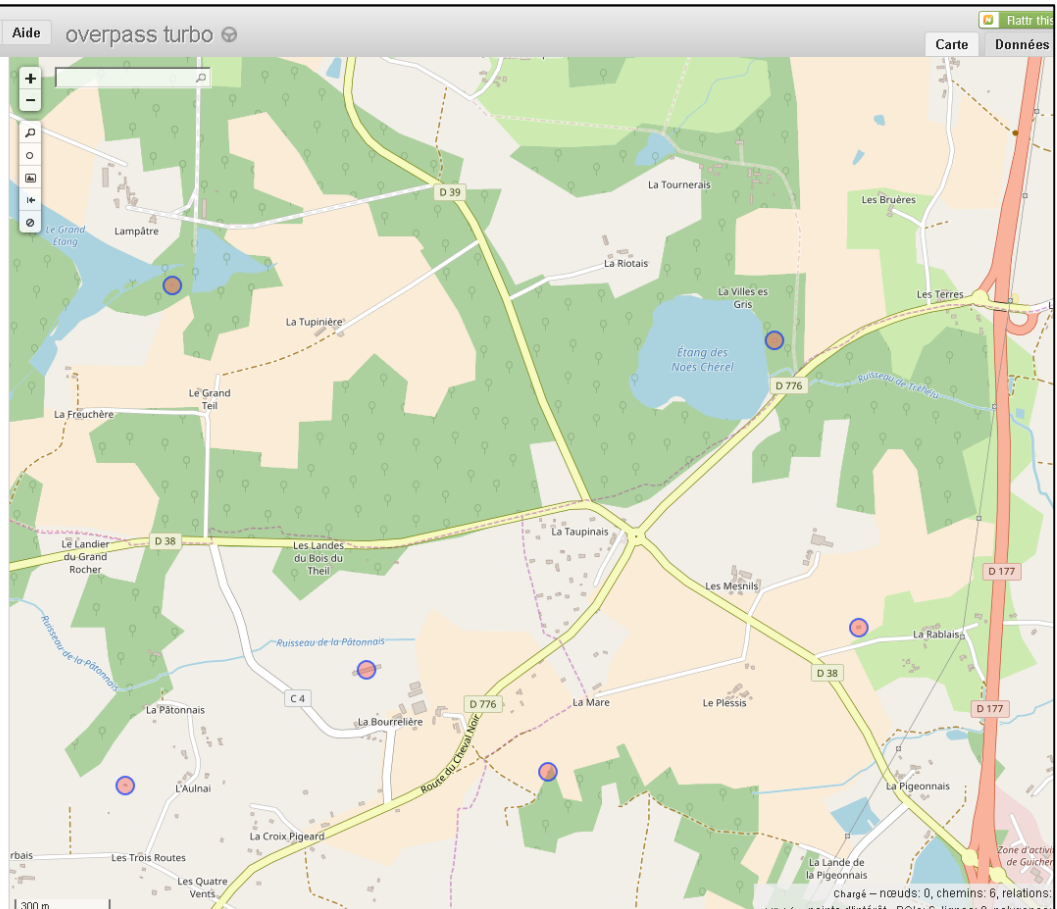


# + Sélection spatiale

Sélectionner les bâtiments isolés (100m)

```
Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo
```

```
1 way[building]({{bbox}})->.a;  
2 foreach .a {  
3   way.a(around:100);  
4   way._(if:count(ways) == 1);  
5   out center;  
6 };
```



way[building]({{bbox}})->.a;  
foreach .a (  
 way.a(around:100);  
 way.\_(if:count(ways) == 1);  
 out center;  
);  
out body;  
>;  
out skel qt;

chargé - nœuds: 0, chemins: 6, relations: 0  
points d'intérêt: 0, POI: 6, lignes: 0, polygones: 0

# + Sélection spatiale

Sélectionner les intersections entre routes principales et petites routes

```
[bbox:({bbox})];  
way[highway~"^(motorway|trunk|primary|secondary|tertiary|  
(motorway|trunk|primary|secondary)_link)$"]->.major;  
way[highway~"^(unclassified|residential|living_street|service)$"]->  
.minor;  
node(w.major)(w.minor);  
out;
```

```
chargé - nœuds: 300, chemins: 0, relations: 0  
Aménagé - points d'intérêt - POIs: 300, lignes: 0, polygones: 0
```



+ Edition de données avec  
l'éditeur OSM ID



# L'éditeur du site OSM ID

<http://www.openstreetmap.org/#map=5/48.821/9.053>

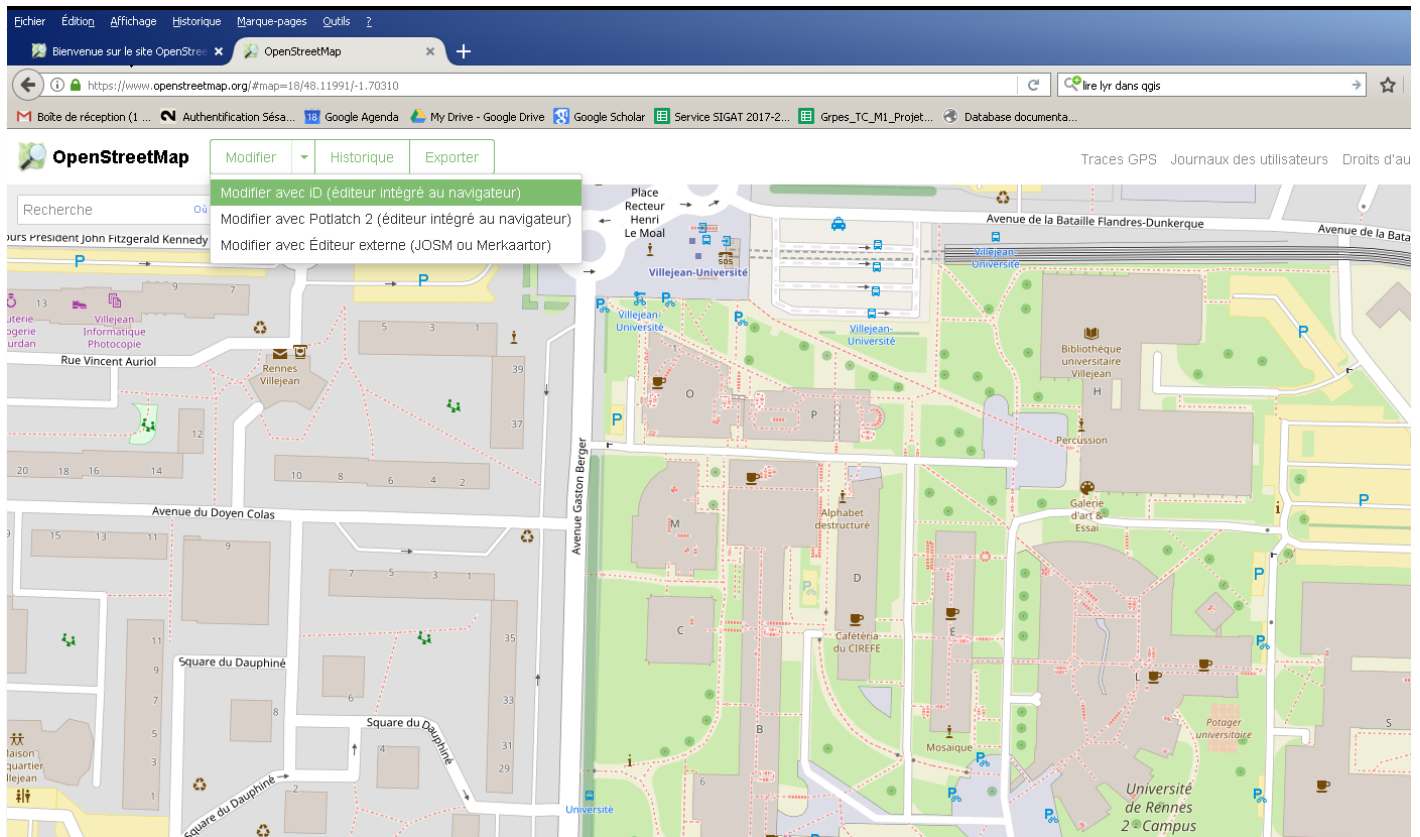
The screenshot shows the OpenStreetMap website interface. At the top left, there is a search bar with the text "Recherche" and a "Où suis-je ?" button. Below the search bar, there is a welcome message: "Bienvenue dans OpenStreetMap !" and a brief description of the project. To the right of the search bar, there are buttons for "Modifier", "Historique", and "Exporter". In the top right corner, there are links for "Traces GPS", "Journaux des utilisateurs", "Droits d'auteur", "Aide", and "À propos". A red box highlights the "Se connecter" button. Below the "Se connecter" button is the "S'inscrire" button. The main part of the page is a map of Europe, showing various countries and cities. The map is centered on a location in central Europe, with a scale bar at the bottom left indicating 300 km and 100 mi. The map is surrounded by a navigation panel on the right side, including a zoom in (+) and zoom out (-) button, a home button, a full screen button, and a help button.



# L'éditeur du site OSM ID



Modifier avec ID (éditeur intégré au navigateur)





# L'éditeur du site OSM ID



Modifier avec ID (éditeur intégré au navigateur)

The screenshot displays the OpenStreetMap ID editor interface. On the left, a sidebar titled "Modifier l'élément" contains a form for editing a building. The form includes fields for "Nom courant (si existant)", "Adresse" (Unité: 123, Rue, Voie; Code postal: Ville), "Étages" (4), and "Sources" (cadastre-dgi-fr source: Direction Générale des Impôts - Cadastre. Mise à jour: 2010). Below the form is a table of tags:

Tous les tags (4)			
building	apartments	🗑️	ℹ️
building:levels	4	🗑️	ℹ️
roof:shape	flat	🗑️	ℹ️
source	cadastre-dgi-fr so...	🗑️	ℹ️

At the bottom of the sidebar, there is a section for "Toutes les relations (0)". The main map area shows a satellite view of a city block with a red building highlighted. The map includes street names like "Avenue du Doyen Colas", "Square du Dauphine", and "Avenue Gaston Berger". The top of the interface features the OpenStreetMap logo, navigation buttons ("Modifier", "Historique", "Exporter"), and user information ("ninanoun").