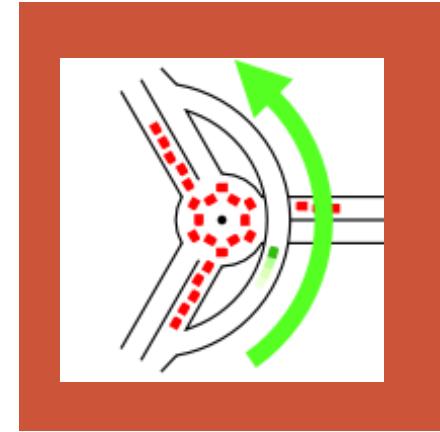


+



# Introduction à l'API Overpass

Interrogation, filtrage et extraction  
de données OSM

@Boris Mericskay



+ Retour sur les données  
OSM (géométrie et  
sémantique)



# Modèle de données OSM

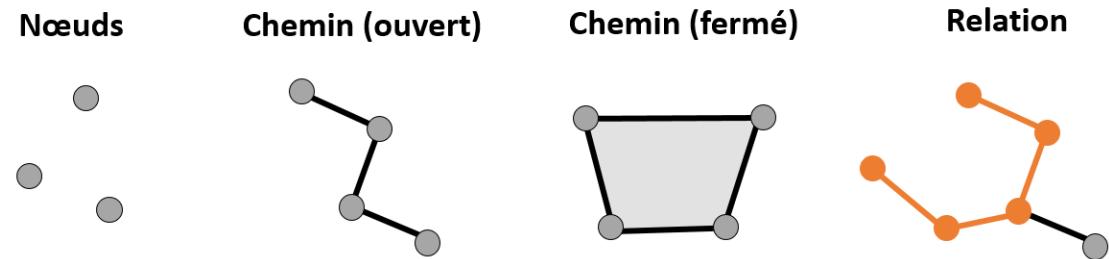


Dans un SIG, les données cartographiques sont représentées de trois façons différentes

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

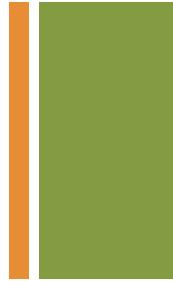
Dans OpenStreetMap, ces trois concepts sont modélisés différemment -> 3 objets primitifs (**Eléments**) :

- **Nœuds** (*nodes*)
- **Lignes** (*ways*)
- **Relations**





# Modèle de données OSM



## Node

- Eléments de base du système OSM
- Les nœuds consistent en une latitude et une longitude
- Peuvent être utilisés seul ou en groupe pour former un chemin

## Way

- Interconnexion entre au moins deux nœuds caractérisant une ligne
- Chemin ouvert / Chemin fermé / Zones

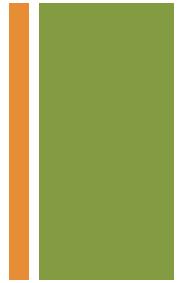


## Relation

- Servent à regrouper différents objets qui considérés les uns avec les autres forment un nouvel objet (ligne de bus, campus...)

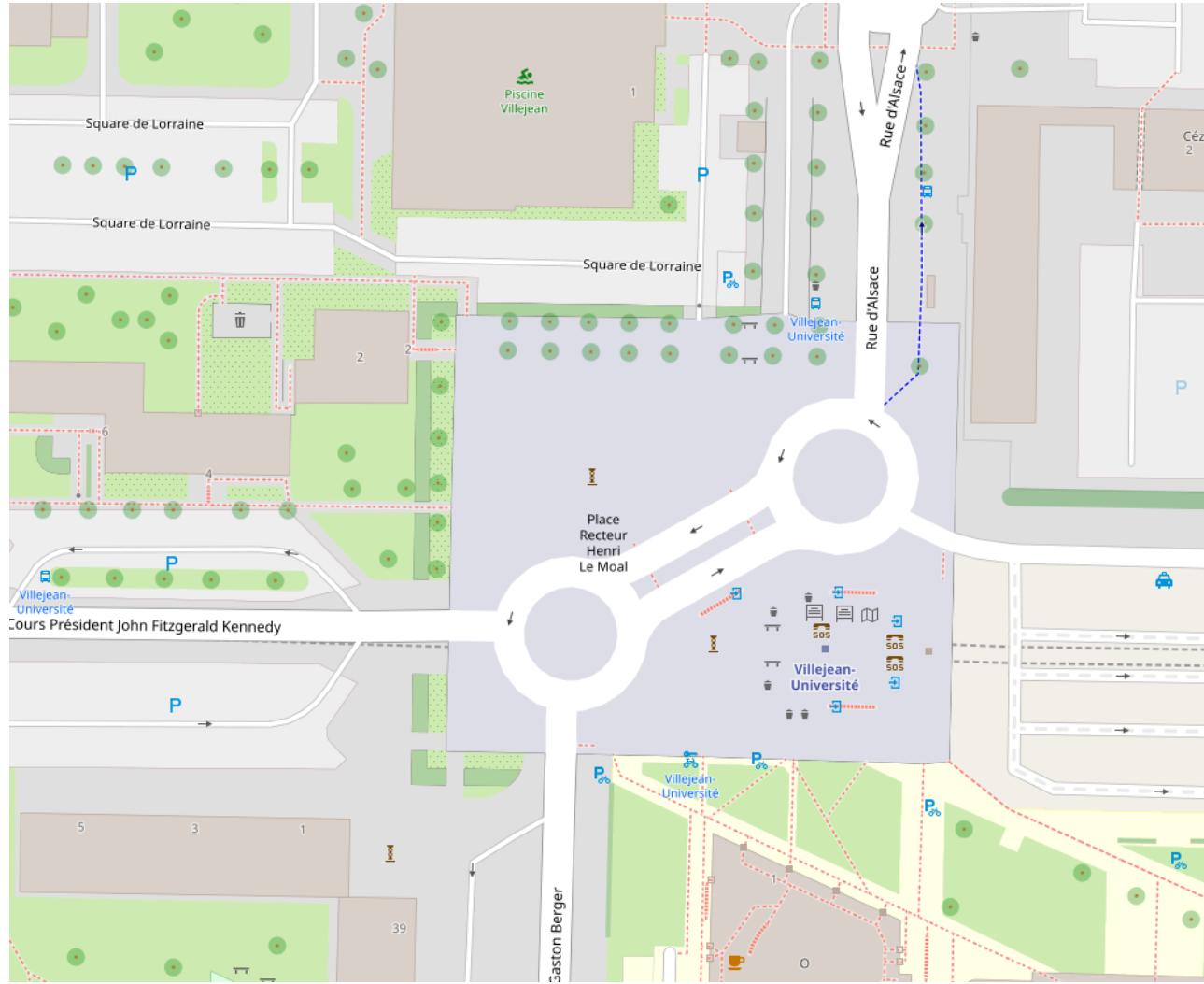


# La modélisation dans OSM



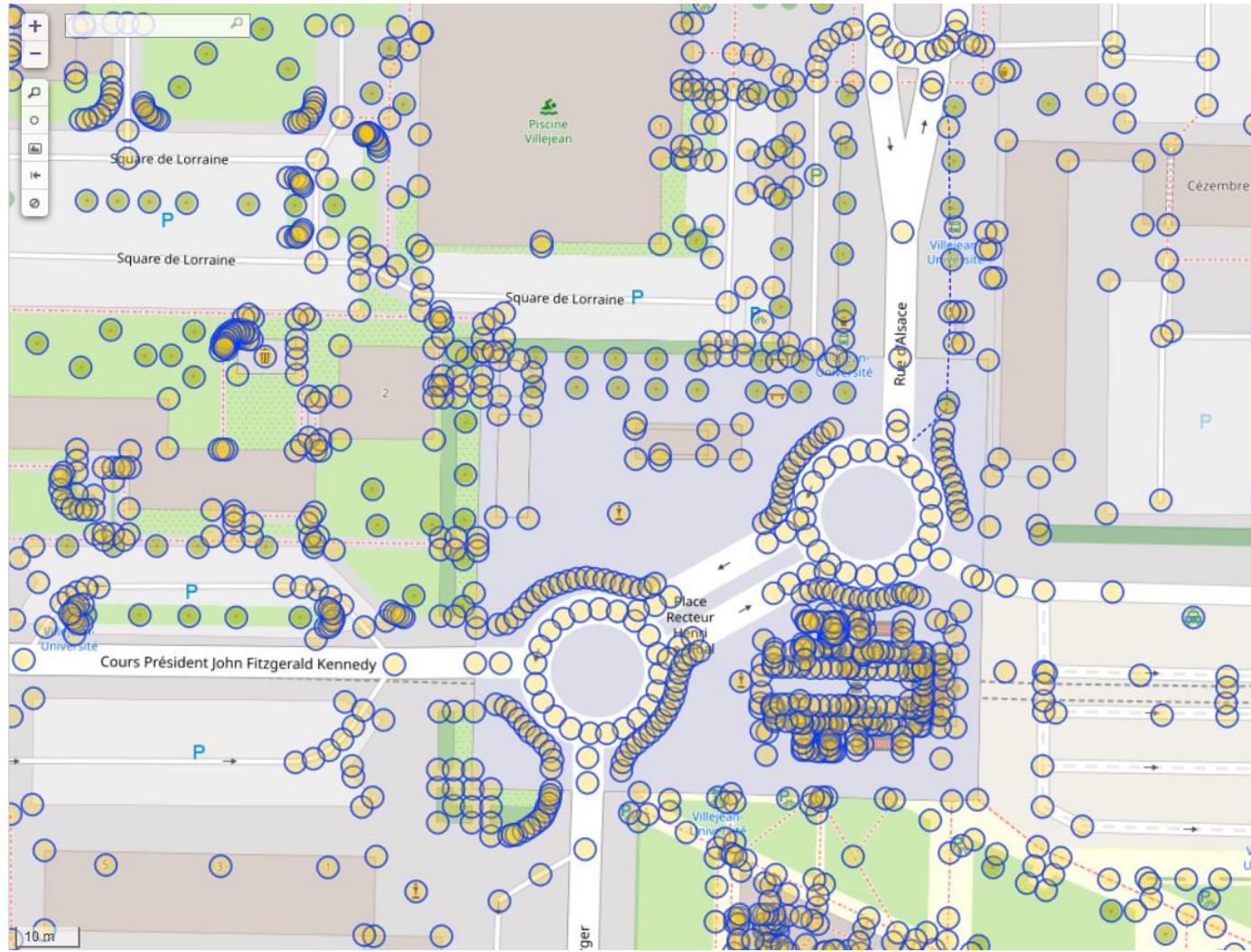


# La modélisation dans OSM



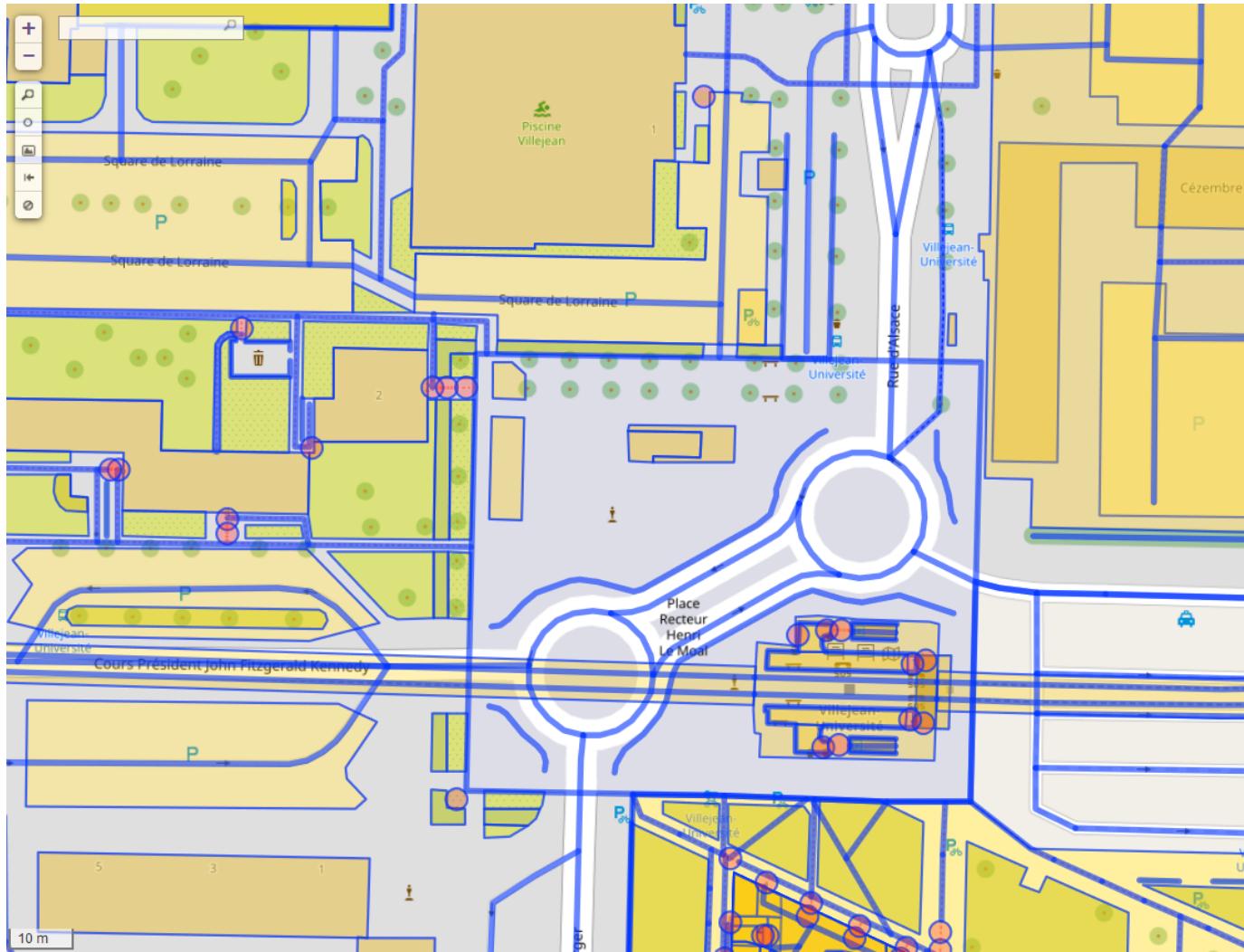


# Nodes



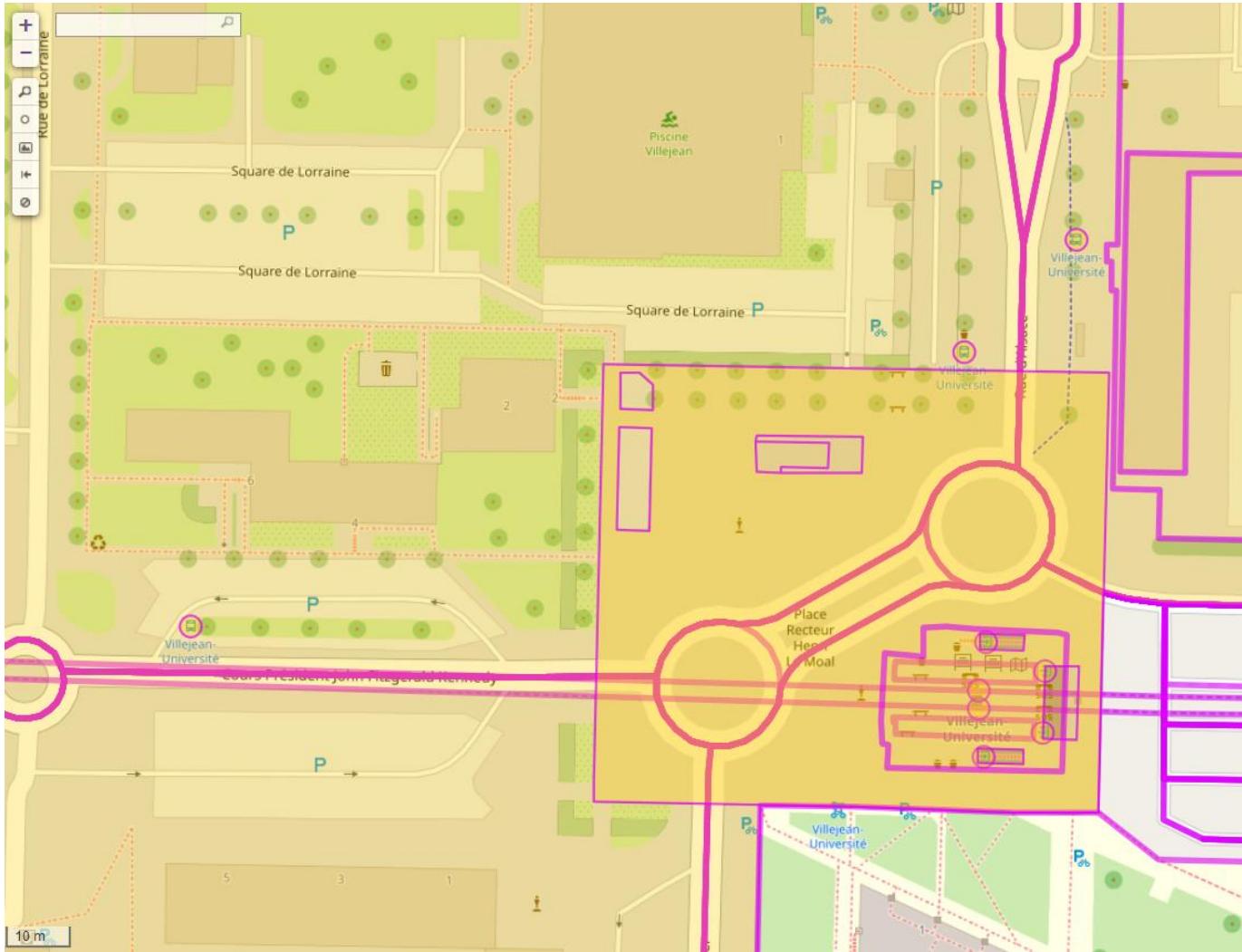


# Ways





# Relations





# La folksonomie OSM



## Une sémantique bien particulière

À chacun des trois éléments peuvent être associés un ou plusieurs **tags (balises)** permettant de le caractériser (étiquetage) les objets cartographiés

- Approche basée sur une **folksonomie** en perpétuelle évolution
- « Indexation personnelle, est un système de classification collaborative décentralisée spontanée, basé sur une indexation effectuée par des non-spécialistes »
- La folksonomie d'OSM bénéficie d'une grande flexibilité et surtout d'une véritable évolutivité



# La folksonomie OSM

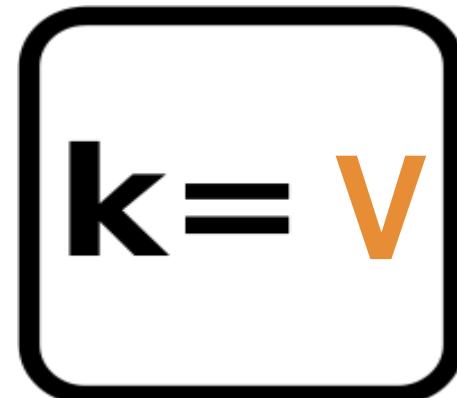


Le schéma des tags repose sur le fonctionnement  
`<clé>=<valeur>` (`key=value`)

→ Possibilité d'associer plusieurs tags à un objet

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





# La folksonomie OSM



Route résidentielle



highway = residential

name = Rue d'Alsace

oneway = yes

surface = paving\_stones

maxspeed = 30

Arrêts de bus



bus = yes

highway = bus\_stop

name = Villejean-Université

public\_transport = platform

shelter = no

wheelchair = yes

Immeuble résidentiel



building = yes

building = apartments

building:colour = white

building:levels = 2

building:material = glass



# La folksonomie OSM



## ⌘ OBJETS

Nombre d'objets: 10 174 266 183  
Nombre de tags: 3 219 411 894

## ⌘ TAGS

Nombre de clés distinctes: 96 975  
Nombre de tags distincts: 157 610 501

## ⌘ NŒUDS

Nombre de nœuds: 9 141 126 090  
Nombre de nœuds avec au moins un tag: 233 827 759  
Pourcentage de nœuds avec au moins un tag: 2.55%  
Nombre de tags sur les nœuds: 802 116 399  
Nombre moyen de tags par nœud taggué: 3.43

## ⌘ CHEMINS

Nombre de chemins: 1 021 004 441  
Nombre de chemins fermés: 719 678 944  
Nombre de tags sur les chemins: 2 368 661 817  
Nombre moyen de tag par chemin: 2.31

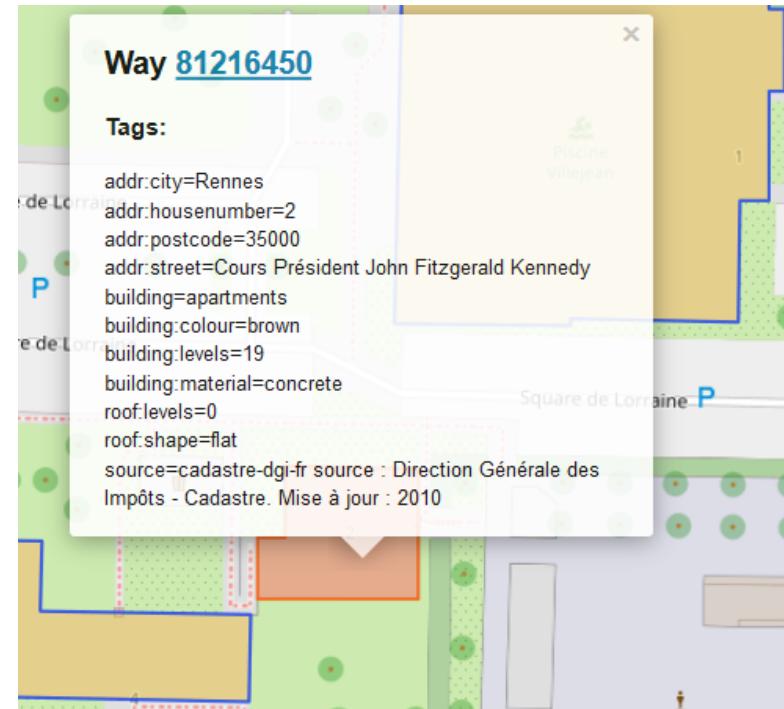
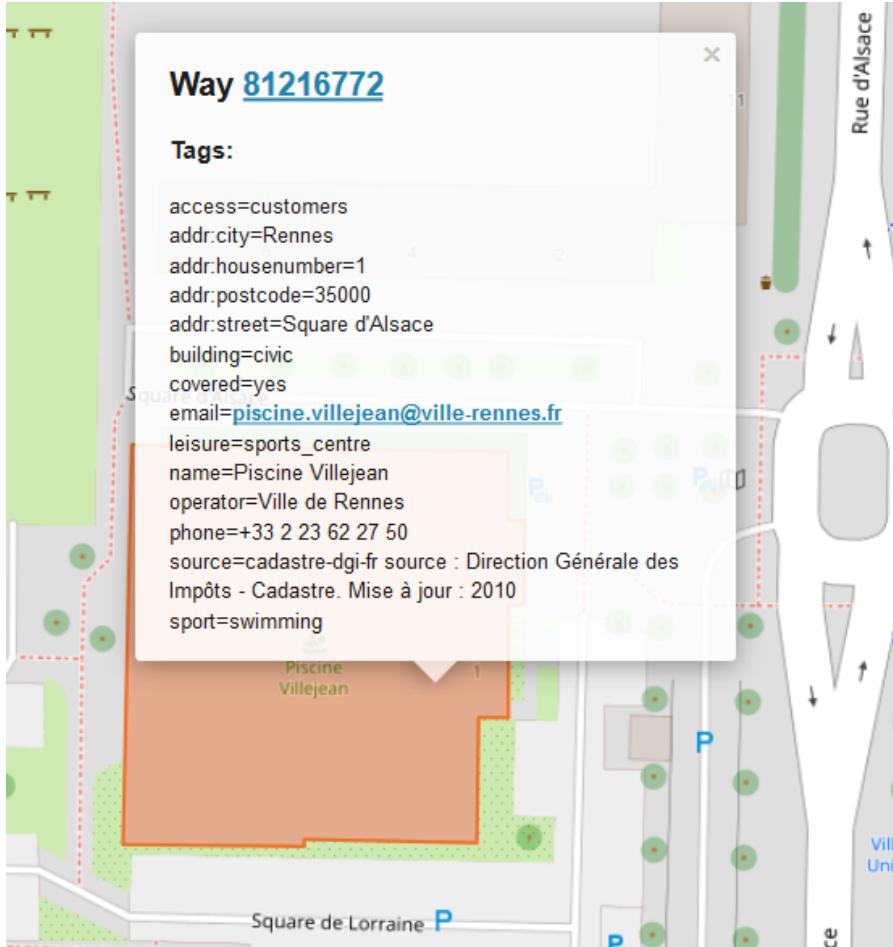
## ⌘ RELATIONS

Nombre de relations: 12 135 652  
Nombre de tags sur les relations: 48 633 678  
Nombre moyen de tag par relation: 4.0

<https://taginfo.openstreetmap.org/>



# La folksonomie OSM





# La folksonomie OSM

**Node [1226405956](#)**

**Tags:**

- addr:street=Place Recteur Henri Le Moal
- amenity=bicycle\_rental
- capacity=26
- name=Villejean-Université
- network=LE vélo STAR
- old\_ref=52
- operator=Keolis Rennes
- payment:credit\_cards=yes
- ref=5552
- source=Keolis Rennes (septembre 2018)
- wheelchair=yes

**Coordinates:**

48.1210768 / -1.7042335 (lat/lon)

**Node [1425717225](#)**

**Tags:**

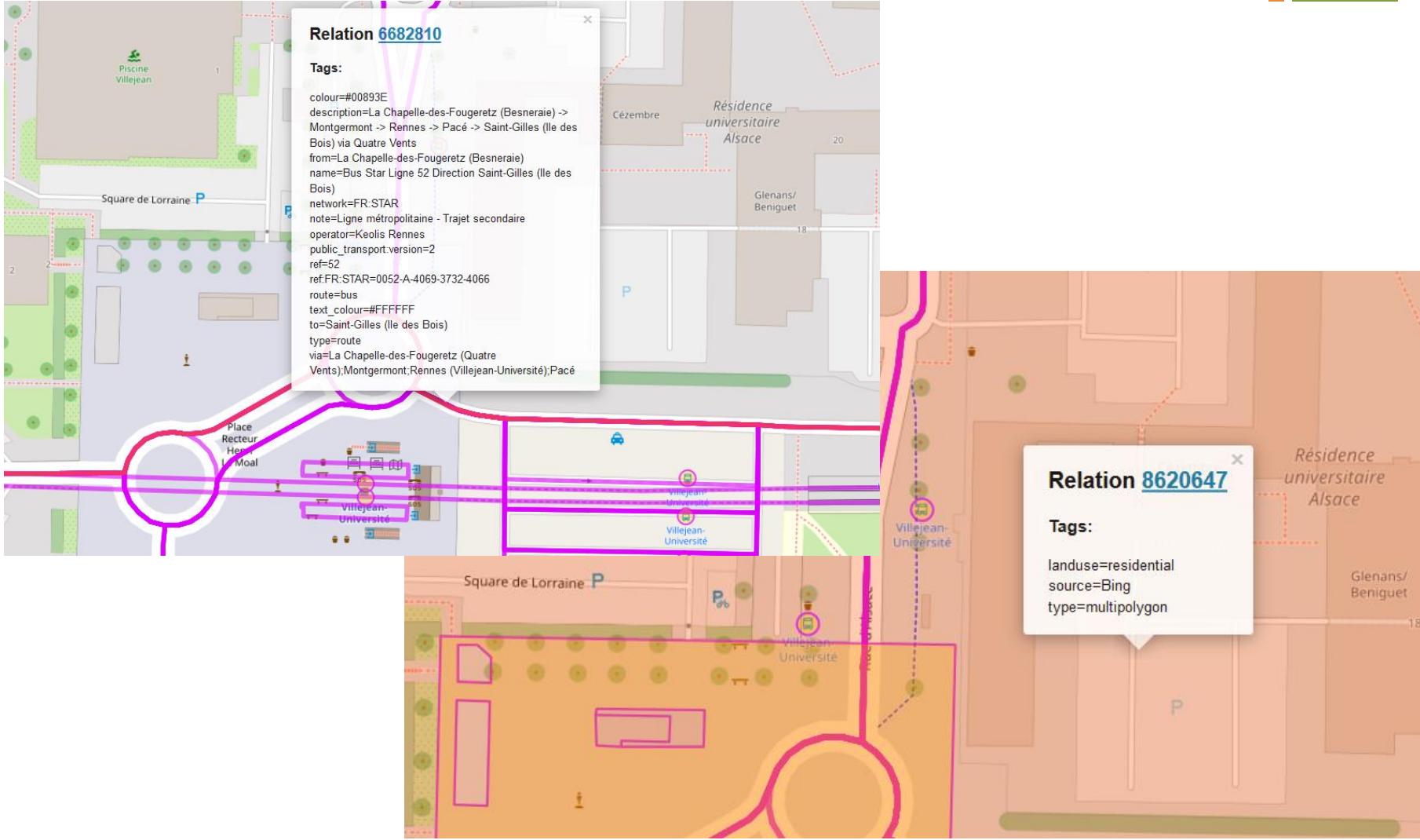
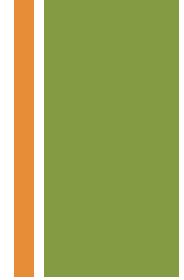
- bench=yes
- highway=bus\_stop
- name=Villejean-Université
- network=FR:STAR
- operator=Keolis Rennes
- public\_transport=platform
- public\_transport:version=2
- ref.FR:STAR=1494
- shelter=yes
- source=Keolis Rennes 5 mai 2016
- wheelchair=yes

**Coordinates:**

48.1212655 / -1.7029669 (lat/lon)



# La folksonomie OSM





# La folksonomie OSM



## Routes

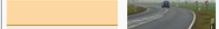
### ■ Documentation

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

Key = **highway**

Value =

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

Key	Value	Element	Comment	Rendering	Photo
<b>Roads</b>					
These are the principal tags for the road network. They range from the most to least important.					
highway	motorway	<input checked="" type="checkbox"/>	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	<input checked="" type="checkbox"/>	The most important roads in a country's system that aren't motorways. (Need not necessarily be a divided highway.)	 	
highway	primary	<input checked="" type="checkbox"/>	The next most important roads in a country's system. (Often link larger towns.)	 	
highway	secondary	<input checked="" type="checkbox"/>	The next most important roads in a country's system. (Often link towns.)	 	
highway	tertiary	<input checked="" type="checkbox"/>	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 system.					

+

## Extraction de données OSM



# <https://www.openstreetmap.org>



OpenStreetMap [Modifier](#) ▾ [Historique](#) [Exporter](#)

Traces GPS Journaux des utilisateurs Droits d'auteur Aide À propos Se connecter S'inscrire

**Exporter**

47.18448  
-1.45485 -1.45064  
47.18294

Sélectionner manuellement une autre zone

**Licence**  
Les données d'OpenStreetMap sont publiées sous la licence [Open Data Commons Open Database \(ODbL\)](#).

[Exporter](#)

Si l'export ci-dessus échoue, veuillez envisager l'utilisation de l'une des sources listées ci-dessous:

**API Overpass**  
Télécharger ce cadre englobant depuis un miroir de la base de données OpenStreetMap

**Planète OSM**  
Copies régulièrement mises à jour de la base de données complète d'OpenStreetMap

**Téléchargements de Geofabrik**  
Extractions régulièrement mises à jour des continents, des pays et des villes sélectionnées

**Autres sources**  
Sources supplémentaires listées sur le wiki d'OpenStreetMap

© Contributrices d'OpenStreetMap ▾ Faire un don Conditions du site web et de l'API



# <https://extract.bbbike.org/>



en de fr ru - intro - à propos

Format [?](#)  
Shapefile (Esri)

Nom de la zone à extraire [?](#) ou rechercher

Votre adresse électronique [?](#)

Déplacez la carte dans la zone souhaitée.  
Cliquez ensuite [\[ \]](#) pour créer un rectangle de sélection. [?](#)

[Donate](#)

VISA MASTERCARD AMEX

accueil | aide | garmin | statut | téléchargement | support commercial  
(©) 2022 BBBike.org par Wolfram Schneider  
Map data (©) OpenStreetMap.org contributors

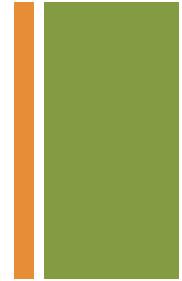
2km

1km

OpenStreetMap contributors, (©) OpenCycleMap  
Permalink 12.97583, 52.62820



<http://download.geofabrik.de/europe/france.html>



#### Download OpenStreetMap data for this region:

## France

[\[one level up\]](#)

The OpenStreetMap data files provided on this server do **not** contain the user names, user IDs and changeset IDs of the OSM objects because these fields are assumed to contain personal information about the OpenStreetMap contributors and are therefore subject to data protection regulations in the European Union.  
[Extracts with full metadata](#) are available to OpenStreetMap contributors only.

#### Commonly Used Formats

- [france-latest.osm.pbf](#), suitable for Osmium, Osmosis, imposm, osm2pgsql, mkgmap, and others. This file was last modified 10 hours ago and contains all OSM data up to 2022-06-07T20:21:39Z. File size: 3.9 GB; MD5 sum: [4705a840197a75d5da69cf1cc0311424](#).
- [france-latest-free.osm.pbf](#) is not available for this region; try one of the sub-regions.

#### Other Formats and Auxiliary Files

- [france-latest.osm.bz2](#), yields OSM XML when decompressed; use for programs that cannot process the .pbf format. This file was last modified 4 days ago. File size: 6.1 GB; MD5 sum: [c5f5d7caa976d1fe1b767af07e56e3e](#).
- [france-internal.osm.pbf](#) The history file contains personal data and is available on the [internal server](#) only. See notice above for further information.
- [poly\\_file](#) that describes the extent of this region.
- [osm.gz\\_files](#) that contain all changes in this region, suitable e.g. for Osmosis updates
- [raw directory index](#) allowing you to see and download older files

#### Sub Regions

Click on the region name to see the overview page for that region, or select one of the file extension links for quick access.

Sub Region	Quick Links		
	<a href="#">.osm.pbf</a>	<a href="#">.shp.zip</a>	<a href="#">.osm.bz2</a>
Alsace	<a href="#">[.osm.pbf]</a> (109 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Aquitaine	<a href="#">[.osm.pbf]</a> (238 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Auvergne	<a href="#">[.osm.pbf]</a> (120 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Basse-Normandie	<a href="#">[.osm.pbf]</a> (106 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Bourgogne	<a href="#">[.osm.pbf]</a> (155 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Bretagne	<a href="#">[.osm.pbf]</a> (269 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Centre	<a href="#">[.osm.pbf]</a> (188 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Champagne Ardenne	<a href="#">[.osm.pbf]</a> (87 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Corse	<a href="#">[.osm.pbf]</a> (28.4 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Franche Comté	<a href="#">[.osm.pbf]</a> (101 MB)	<a href="#">[.shp.zip]</a>	<a href="#">[.osm.bz2]</a>
Guadeloupe	<a href="#">[.osm.pbf]</a> (25.9 MB)	✗	<a href="#">[.osm.bz2]</a>
Guyane	<a href="#">[.osm.pbf]</a> (11.6 MB)	✗	<a href="#">[.osm.bz2]</a>

GEOFABRIK® [downloads](#)



Not what you were looking for? Geofabrik is a consulting and software development firm based in Karlsruhe, Germany specializing in OpenStreetMap services. We're happy to help you with data preparation, processing, server setup and the like. [Check out our website](#) and contact us if we can be of service.

Nicht das Richtige dabei? Die Geofabrik ist ein auf OpenStreetMap spezialisiertes Beratungs- und Softwareentwicklungsunternehmen in Karlsruhe. Gern helfen wir Ihnen bei der Datenaufbereitung, Datenkonvertierung, Serverinstallation und ähnlichen Aufgaben. [Besuchen Sie unsere Webseite](#) und sprechen Sie mit uns, wenn wir Ihnen helfen können.

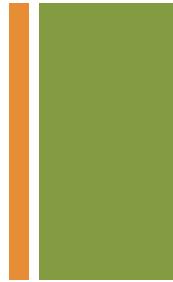
+

Quick OSM

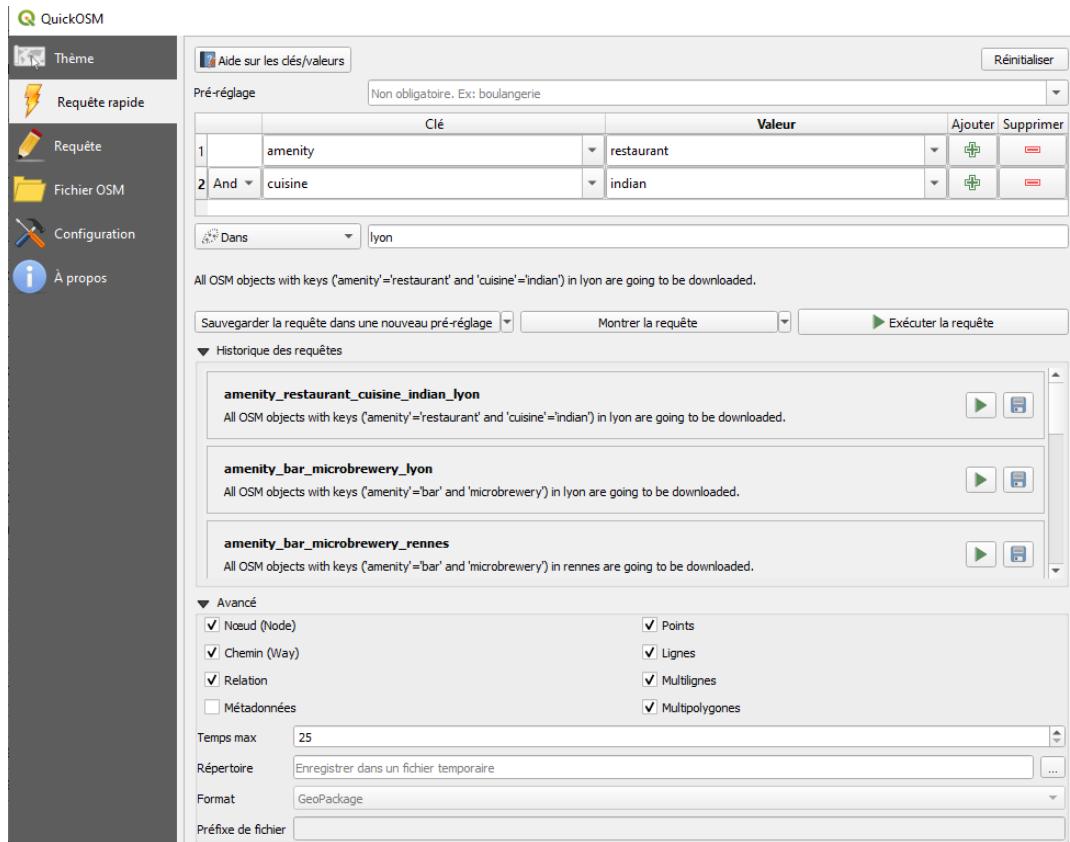


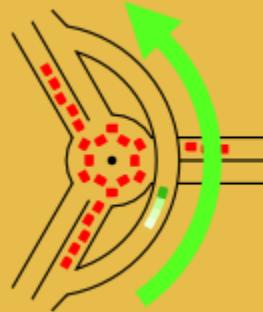


# Quick OSM



- Extension QGIS qui permet de récupérer des données OSM (via l'API Overpass) selon des requêtes (avec des critères) et une entrée géographique





API Overpass  
+  
&  
Overpass turbo



# API Overpass



L'API Overpass est une **API en lecture seule qui sert à importer des données cartographiques OSM**.

Elle se comporte comme une base de données sur le web : le client envoie une requête à l'API et récupère l'ensemble de données correspondant à la requête.

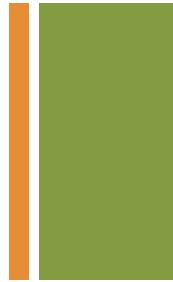
Contrairement à l'[API](#) principale qui est optimisée pour l'édition, l'API Overpass est optimisée pour les utilisateurs de données qui ont besoin de données

- sélectionnés par des critères de recherche comme par exemple l'emplacement, le type d'objets, les propriétés des balises, leur proximité, ou leurs combinaisons.





# Overpass turbo



Overpass turbo, un outil web **d'exploration**, de **filtrage** et de **récupération** de données OSM

Avec Overpass turbo on peut exécuter (plus simplement) des requêtes de l'API Overpass et visualiser directement les données OSM sur une carte interactive

Il existe un assistant intégré qui permet de créer des requêtes facilement

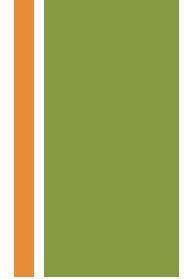
Plus d'informations au sujet de [overpass turbo](#) et sur la création de [requêtes Overpass](#) peuvent être trouvées sur le wiki OSM.

+

# Requêtes classiques



# L'interface globale



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

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

Carte Données

```
/*
This is an example Overpass query.
Try it out by pressing the Run button above!
You can find more examples with the Load tool.
*/
node
[amenity=drinking_water]
({{bbox}});
out;
```

The map displays the historical center of Rome, including the Colosseum, the Imperial Forums (Fori Imperiali), and various streets like Via Sacra, Via dei Fori Imperiali, and Via dei Cerchi. The brown icons represent drinking water fountains located throughout the area.



# Changer le fond de carte

Paramètres

Par exemple « 400px », laisser vide pour la valeur par défaut

Carte

Serveur de tuiles

▼

Opacité des Tuiles

0.8 ▲ ▼

Opacité des tuiles d'arrière-plan: 0=transparent ... 1=visible

Affiche un repère au centre de la carte

N'affiche pas les petits objets comme points d'intérêts.

Affiche des statistiques à propos des données chargées et affichées.

Partager

Inclure l'état actuel de la carte dans les permaliens

[réinitialiser](#) [Enregistrer](#) [fermer](#)

#URLs that work with Overpass Turbo Go to Settings > Map and change the URI to your server of choice.

The default is: `//{s}.tile.openstreetmap.org/{z}/{x}/{y}.png`

The following tile servers also work:

- Humanitarian: `//{s}.tile.openstreetmap.fr/hot/{z}/{x}/{y}.png`
- Mapnik, Grayscale: `//tiles.wmflabs.org/bw-mapnik/{z}/{x}/{y}.png`
- Thunderforest
  - Landscape: `//{s}.tile.thunderforest.com/landscape/{z}/{x}/{y}.png`
  - Neighbourhood: `//{s}.tile.thunderforest.com/neighbourhood/{z}/{x}/{y}.png`
  - Outdoors: `//{s}.tile.thunderforest.com/outdoors/{z}/{x}/{y}.png`
  - Mobile Atlas: `//{s}.tile.thunderforest.com/mobile-atlas/{z}/{x}/{y}.png`
  - Transport: `//{s}.tile.thunderforest.com/transport/{z}/{x}/{y}.png`
- Mapbox
  - Satellite: `{s}.tiles.mapbox.com/v4/mapbox.satellite/{z}/{x}/{y}.jpg?access_token=pk.eyJ1Ijoib3BlbnN0cmVldG1hcCIsImEiO1Jja2w5YWt5bnYnJzMFnFwZjhtbHk1MnA1In0.eq2aumBK6JuRoIuBMm6Gew`
  - Wheatpaste: `https://{s}.tiles.mapbox.com/v4/mapbox.wheatpaste/{z}/{x}/{y}.png?access_token=pk.eyJ1Ijoib3BlbnN0cmVldG1hcCIsImEiO1Jncjlmd0t3In0.DmZsIeOW-3x-C5eX-wAqTw`
  - Comic: `http://api.mapbox.com/v4/mapbox.comic.html?access_token=pk.eyJ1Ijoib3BlbnN0cmVldG1hcCIsImEiO1Jncjlmd0t3In0.DmZsIeOW-3x-C5eX-`

<https://gist.github.com/govvin/a095afd2fd53c9e5f2af232ef55ffde1>



# Requêtes classiques



## Extraire les routes selon une valeur (hierarchie)

- Les routes principales (trunk)

Type

Key

Value

way

highway

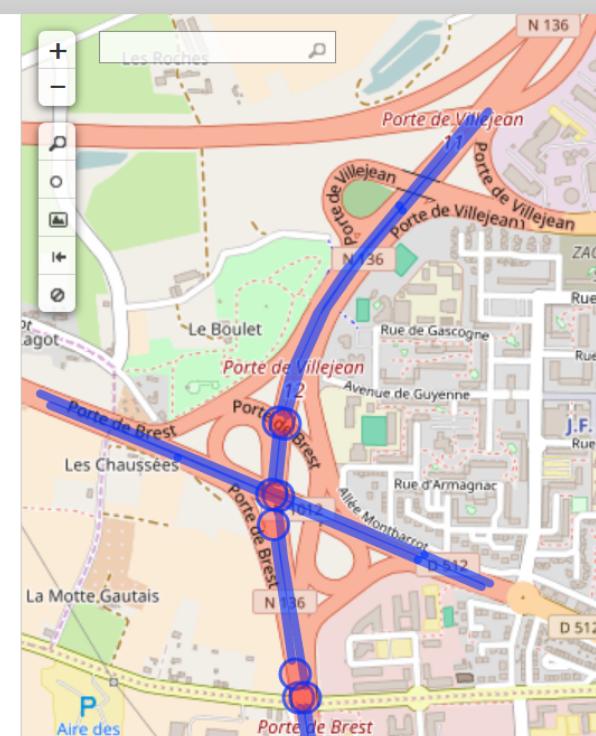
trunk

This has been generated by the overpass-turbo wizard.  
The original search was:

```
8  // gather results
9  // query part for: route
10 way["highway"="trunk"]({{bbox}});
11 )
12 // print results
13 out body;
14 >;
15 out skel qt;
```

{{bbox}}

Emprise



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



# Sortie de la requête

## Sortie des données (out)

L'instruction `out` s'exécute sur un lot d'entrée. Elle ne retourne pas de lot résultat. Le lot d'entrée peut être changé par le nom de variable voulu en début d'instruction.

L'instruction `out` peut être configurée en ajoutant un nombre variable de paramètres séparés par des espaces, entre le mot `out` et le point-virgule final.

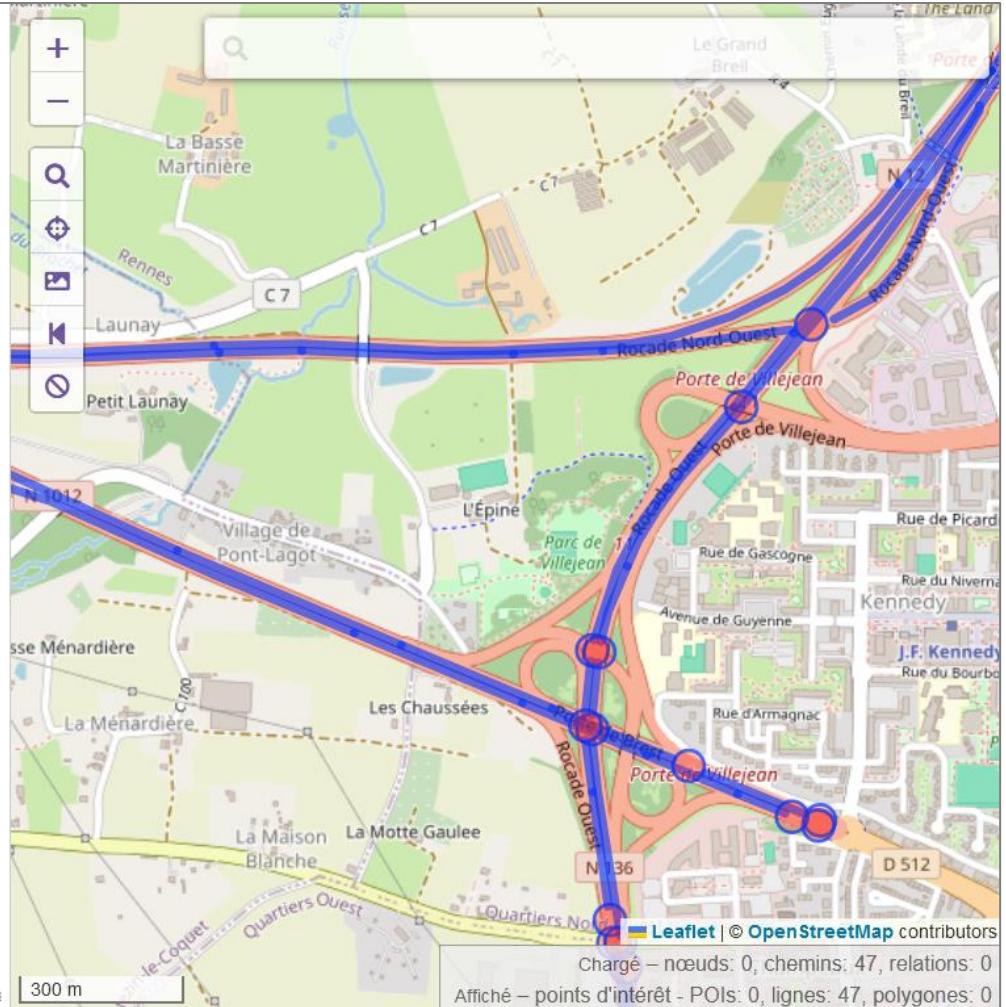
Les paramètres autorisés, dans un ordre quelconque, sont :

- un niveau de verbosité parmi les suivants (sélection `body` défaut) :
  - `ids` : sort seulement les IDs des éléments.
  - `skel` : sort aussi les informations minimum nécessaires pour la géométrie. Ce sont entre autres les coordonnées des *nœuds*, des *chemins*, la liste ordonnée des identifiants des *nœuds* membres de *chemins* ou la liste des types et identifiants des éléments membres de *relations*.
  - `body` : sort toutes les informations nécessaires pour utiliser les données. Ce sont entre autres les attributs (clés et valeurs) de tous les éléments et les rôles des membres de relation.
  - `tags` : sort seulement les identifiants et attributs de chaque élément sans les coordonnées des membres (ce sont les seules informations disponibles à partir d'éléments dérivés générés par des instructions "make" ou "convert", tels les attributs géométriques des *nodes*, et les éléments de contenu des *ways* et des *relations* sont absentes).
  - `meta` : sort toutes les informations associées aux éléments. Cela complète la sortie `body` en incluant pour chaque élément son numéro de version, l'identifiant du groupe de modifications (*changeset*), son horodatage, et des données pour identifier le dernier utilisateur qui a touché l'élément (notez que ces métadonnées d'attributs manquent également pour les éléments dérivés générés par des instructions "make" ou "convert").
  - `count` : sort seulement le [comptage des éléments](#).
- un des arguments suivants pour des informations géolocalisées, `geom` par défaut. (notez que cette information est manquante pour les éléments dérivés créés par des instructions spéciales "make" ou "convert") :
  - `geom` : ajoute la géométrie complète à chaque objet. Cela ajoute les coordonnées à chaque *nœud*, ainsi qu'à chaque *nœud* membre d'un *chemin* ou d'une *relation*, et ajoute une liste de membres `nd` avec leurs coordonnées à toutes les *relations*.
  - `bb` : ajoute la boîte de sélection de chaque élément à ces éléments. Pour les *nœuds* ceci est équivalent à `geom`. Pour les *chemins* c'est la boîte de sélection englobant tous leurs *nœuds*. Pour les *relations*, c'est la boîte de sélection englobant tous les *nœuds* et *chemins* membres, sans que cela s'applique aux *relations* membres.
  - `center` : ajoute le centre de la boîte de sélection mentionnée ci-dessus aux *chemins*' et *relations*. Note : il n'est pas garanti que le point central se trouve à l'intérieur du polygone si ce dernier n'est pas convexe ([exemple ↗](#)).



# Sortie de la requête / out geom

```
[out:json][timeout:25];  
  
// Données recherchées  
way["highway" = "trunk"]({{bbox}});  
  
// print results  
out geom;
```



Chargé – noeuds: 0, chemins: 47, relations: 0

Affiché – points d'intérêt - POIs: 0, lignes: 47, polygones: 0



# Sortie de la requête / out ids

```
[out:json][timeout:25];  
  
// Données recherchées  
way["highway" = "trunk"]({{bbox}});  
  
// print results  
out ids;  
  
1 {  
2   "version": 0.6,  
3   "generator": "Overpass API 0.7.62.1 084b4234",  
4   "osm3s": {  
5     "timestamp_osm_base": "2024-06-26T15:35:59Z",  
6     "copyright": "The data included in this document is from  
7   },  
8   "elements": [  
9  
10  {  
11    "type": "way",  
12    "id": 5031212  
13  },  
14  {  
15    "type": "way",  
16    "id": 5077157  
17  },  
18  {  
19    "type": "way",  
20    "id": 5406831  
21  },  
22  {  
23    "type": "way",  
24    "id": 23811949  
25  },  
26  {  
27    "type": "way",  
28    "id": 23811952  
29  },  
30  {  
31    "type": "way",  
32    "id": 23884297
```



# Sortie de la requête / out tags

```
[out:json][timeout:25];  
  
// Données recherchées  
way["highway" = "trunk"]({{bbox}});  
  
// print results  
out tags;  
  
1 {  
2   "version": 0.6,  
3   "generator": "Overpass API 0.7.62.1 084b4234",  
4   "osm3s": {  
5     "timestamp_osm_base": "2024-06-26T15:34:57Z",  
6     "copyright": "The data included in this document is from www.  
7   },  
8   "elements": [  
9  
10  {  
11    "type": "way",  
12    "id": 5031212,  
13    "tags": {  
14      "bridge": "yes",  
15      "highway": "trunk",  
16      "int_ref": "E 03",  
17      "lanes": "2",  
18      "layer": "1",  
19      "legal": "motorroad",  
20      "lit": "no",  
21      "mapillary": "956910288409482",  
22      "maxspeed": "90",  
23      "motorroad": "yes",  
24      "name": "Rocade Ouest",  
25      "name:br": "Hent-tro Kornôg",  
26      "oneway": "yes",  
27      "ref": "N 136",  
28      "source:name:br": "proper translation",  
29      "surface": "asphalt",  
30      "survey:date": "2019-04-30"  
31    }  
32  },  
33}
```



# Sortie de la requête / out count

```
[out:json][timeout:25];  
  
// Données recherchées  
way["highway" = "trunk"]({{bbox}});  
  
// print results  
out count;  
  
1 {  
2   "version": 0.6,  
3   "generator": "Overpass API 0.7.62.1 084b4234",  
4   "osm3s": {  
5     "timestamp_osm_base": "2024-06-26T15:32:56Z",  
6     "copyright": "The data included in this document is from www.o  
7   },  
8   "elements": [  
9  
10  {  
11    "type": "count",  
12    "id": 0,  
13    "tags": {  
14      "nodes": "0",  
15      "ways": "47",  
16      "relations": "0",  
17      "total": "47"  
18    }  
19  }  
20  ]  
21}  
22}  
23}
```



# Sortie de la requête / out skel

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

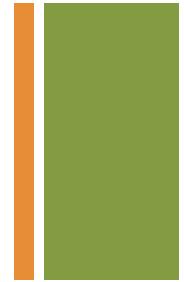
// Données recherchées
way["highway" = "trunk"]({{bbox}});

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

1  {
2    "version": 0.6,
3    "generator": "Overpass API 0.7.62.1 084b4234",
4    "osm3s": {
5      "timestamp_osm_base": "2024-06-26T15:55:45Z",
6      "copyright": "The data included in this document is from www.
7    },
8    "elements": [
9      {
10        "type": "way",
11        "id": 5031212,
12        "nodes": [
13          27247589,
14          8298514332,
15          1339099288
16          ]
17      },
18    ],
19    {
20      "type": "way",
21      "id": 5077157,
22      "nodes": [
23        8614364648,
24        27247587,
25        34478777,
26        27247588
27      ]
28    },
29    {
30      "type": "way",
31      "id": 5406831,
32      "nodes": [
```



# Paramètres d'export



## out body

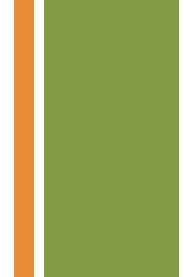
	<code>id</code>	<code>@id</code>	<code>highway</code>	<code>int_ref</code>	<code>lanes</code>	<code>legal</code>	<code>lit</code>	<code>looks_like</code>	<code>maxspeed</code>	<code>motorroad</code>	<code>name</code>	<code>oneway</code>	<code>ref</code>
1	node/6746011943	node/6746011943	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
2	node/6746011942	node/6746011942	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
3	node/6746011947	node/6746011947	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
4	node/6746011944	node/6746011944	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
5	node/603272286	node/603272286	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
6	node/5081278620	node/5081278620	motorway_junction	NULL	NULL	NULL	NULL	NULL	NULL	Porte de Brest	NULL	12	
7	node/6746011941	node/6746011941	traffic_signals	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
8	node/6746011940	node/6746011940	traffic_signals	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
9	node/676266083	node/676266083	motorway_junction	NULL	NULL	NULL	NULL	NULL	NULL	Porte de Saint-Malo	NULL	13b	
10	node/6746011948	node/6746011948	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
11	node/4249241146	node/4249241146	motorway_junction	NULL	NULL	NULL	NULL	NULL	NULL	Porte de Beuregard	NULL	13a	
12	node/3794418362	node/3794418362	milestone	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	A84	
13	node/4544148244	node/4544148244	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
14	node/4544148233	node/4544148233	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
15	node/3029169940	node/3029169940	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
16	node/2626016994	node/2626016994	traffic_signals	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
17	node/3560888991	node/3560888991	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
18	node/3029169958	node/3029169958	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
19	node/4616799148	node/4616799148	motorway_junction	NULL	NULL	NULL	NULL	NULL	NULL	Porte de Normandie	NULL	16	
20	node/4556247439	node/4556247439	motorway_junction	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
21	node/4773815229	node/4773815229	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
22	node/4616799169	node/4616799169	motorway_junction	NULL	NULL	NULL	NULL	NULL	NULL	Porte de Normandie	NULL	16	
23	node/4544148379	node/4544148379	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
24	node/1333397667	node/1333397667	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
25	node/4544148283	node/4544148283	give_way	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## out skel

	<code>id</code>	<code>@id</code>
1	node/1842799974	node/1842799974
2	node/1842799971	node/1842799971
3	node/1842799979	node/1842799979
4	node/1842799976	node/1842799976
5	node/1842799965	node/1842799965
6	node/1842799959	node/1842799959
7	node/1842799970	node/1842799970
8	node/1842799967	node/1842799967
9	node/1842800008	node/1842800008
10	node/1842800007	node/1842800007
11	node/1842800012	node/1842800012
12	node/1842800009	node/1842800009
13	node/1842800003	node/1842800003



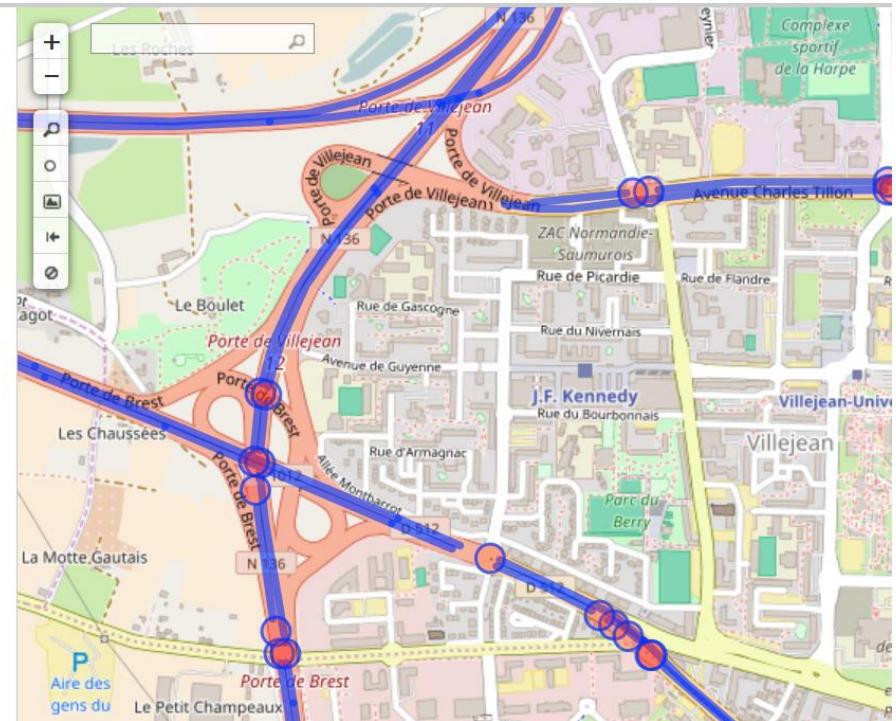
# Requêtes classiques



## 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;
```



+

# Requêtes classiques

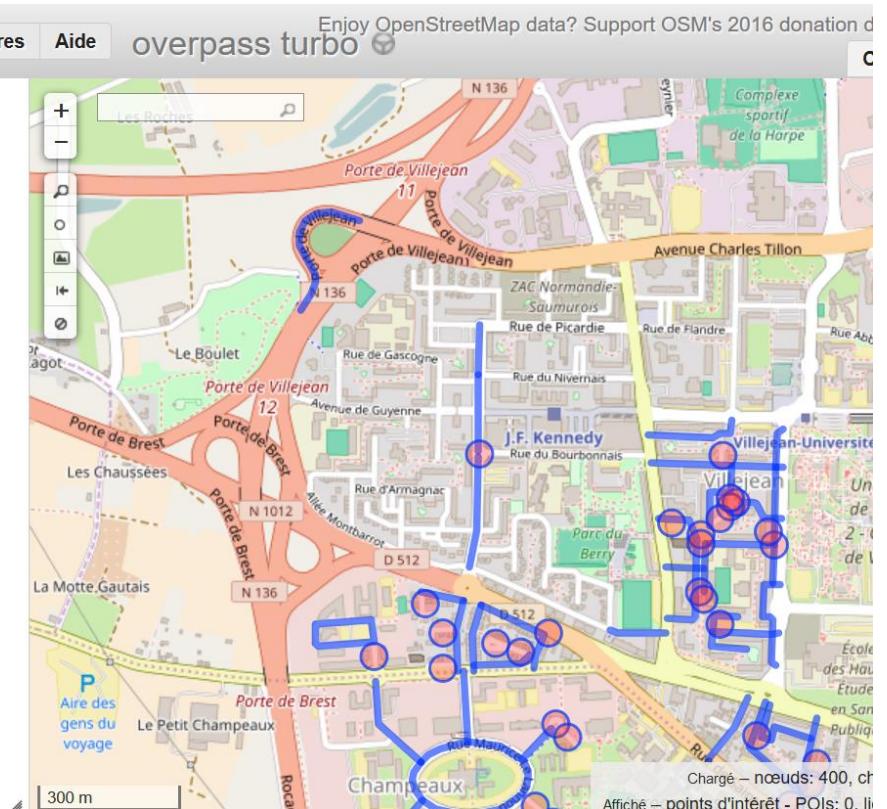


## Ajouter un critère de vitesse

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

```
Exécuter Partager Exporter Assistant Enregistrer CLI

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;
```





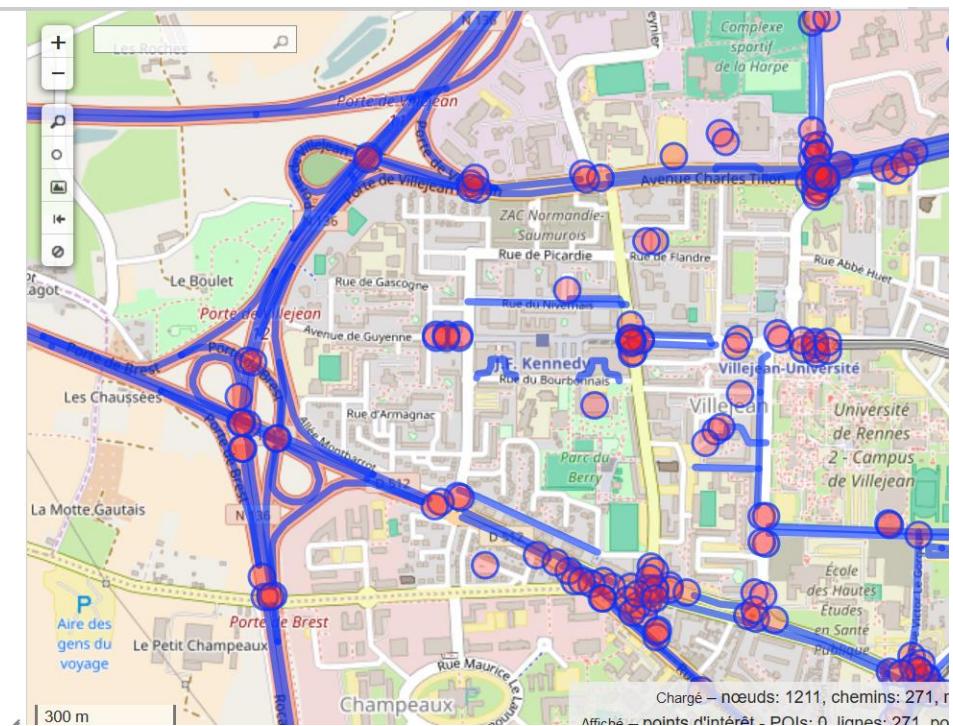
# Requêtes classiques



## Ajouter un critère de direction

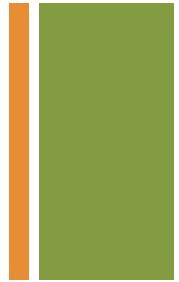
- Les routes en sens unique (*oneway*)
- <http://wiki.openstreetmap.org/wiki/Key:oneway>

```
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  // print results
13  out body;
14  >;
15  out skel qt;
```





# Modifier l'emprise de la recherche



Deux options :

1. Utiliser l'emprise de la carte (bbox)

```
way ["highway"="trunk"]({{bbox}});
```

2. Utiliser un nom de lieu (ville, région, pays,...)

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

```
way ["highway"="trunk"](area.searchArea);
```



# Modifier l'emprise de la recherche



raccourci	description	exemple	remplacé par	
			(XML-query)	(OverpassQL)
<code>{{bbox}}</code>	Est remplacé par les coordonnées de la boîte englobante de la fenêtre de la carte courante.	<code>{{bbox}}</code>	s="48..." w="16..." n="49..." e="17..."	48...,16...,49...,17...
<code>{{center}}</code>	Est remplacé par les coordonnées du centre de la fenêtre de la carte courante.	<code>{{center}}</code>	lat="48..." lon="16..."	48...,16...
<code>{{date:string}}</code>	Est remplacé par le date-time-string ISO 8601 ↗ d'une date relative dans le passé. Par exemple, <code> {{date:1 day}}</code> est remplacé par l'horodatage 24 heures avant. Les unités de temps autorisées sont : seconde(s), minute(s), heure(s), jour(s), semaine(s), mois et année(s).	<code>{{date:1 day}}</code>	2013-11-24T22:55:36.201Z	
<code>{{geocodeId:name}}</code>	Est remplacé par l'ID OSM du premier résultat d'une recherche géographique (ex par Nominatim) du <code>name</code> .	<code>{{geocodeId:Vienna}}</code>	type="relation" ref="1645..."	relation(1645...)
<code>{{geocodeArea:name}}</code>	Est remplacé par l'ID Overpass-Area du premier résultat d'une recherche géographique (ex par Nominatim) du <code>name</code> .	<code>{{geocodeArea:Vienna}}</code>	type="area" ref="3601645..."	area(3601645...)
<code>{{geocodeBbox:name}}</code>	Est remplacé par la bbox du premier résultat d'une recherche géographique (ex par Nominatim) du <code>name</code> .	<code>{{geocodeBbox:Vienna}}</code>	s="48..." w="16..." n="49..." e="17..."	48...,16...,49...,17...
<code>{{geocodeCoords:name}}</code>	Est remplacé par les coordonnées (barycentre) du premier résultat d'une recherche géographique (ex par Nominatim) du <code>name</code> .	<code>{{geocodeCoords:Vienna}}</code>	lat="48..." lon="16..."	48...,16...

[https://wiki.openstreetmap.org/wiki/FR:Overpass\\_turbo/Requ%C3%AAtes\\_Overpass\\_Turbo\\_%C3%A9tendues](https://wiki.openstreetmap.org/wiki/FR:Overpass_turbo/Requ%C3%AAtes_Overpass_Turbo_%C3%A9tendues)



# Emprise de la recherche

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

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

<https://overpass-turbo.eu/s/1Nrk>

```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"highway=trunk in Rennes"

*/
[out:json][timeout:25];
// fetch area "Rennes" to search in
{{geocodeArea:Rennes}}->.searchArea;
// gather results
(
    // query part for: "highway=trunk"
    way["highway"]["maxspeed"="30"]["oneway"="yes"]
(area.searchArea);
);
// print results
out body;
>;
out skel qt;
```

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

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

```
 way ["highway"] ["oneway"="yes"] ["maxspeed"="30"] (area.searchArea);
```

```
out body;
```

```
>;
```

```
out skel qt;
```





# Emprise de la recherche

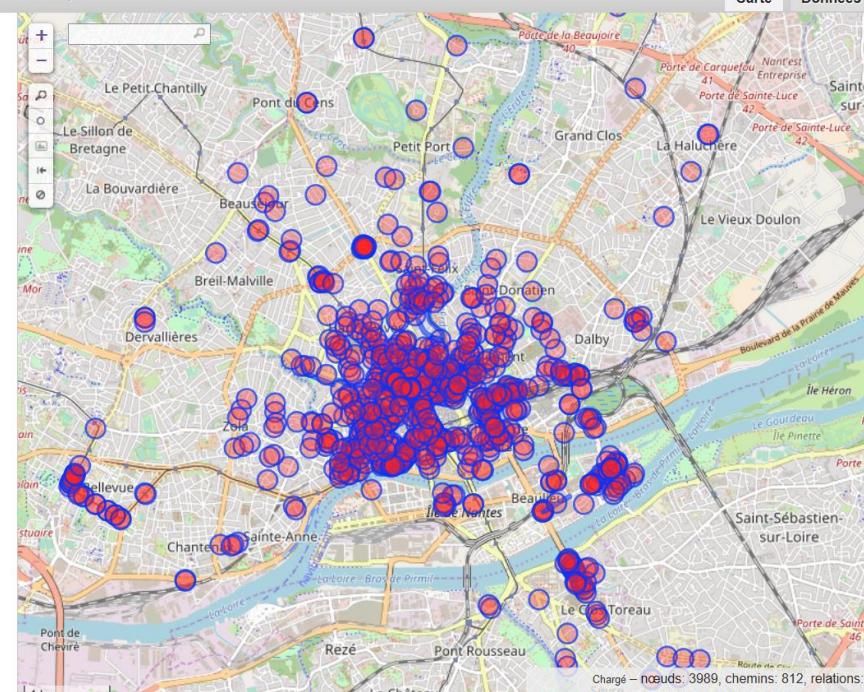


## Choisir la zone d'interrogation (emprise de la requête)

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

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aideoverpass turboCarte Données

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



Chargé – noeuds: 3989, chemins: 812, relations: 0  
Affiché – points d'intérêt - POIs: 0, lignes: 812, polygones: 0



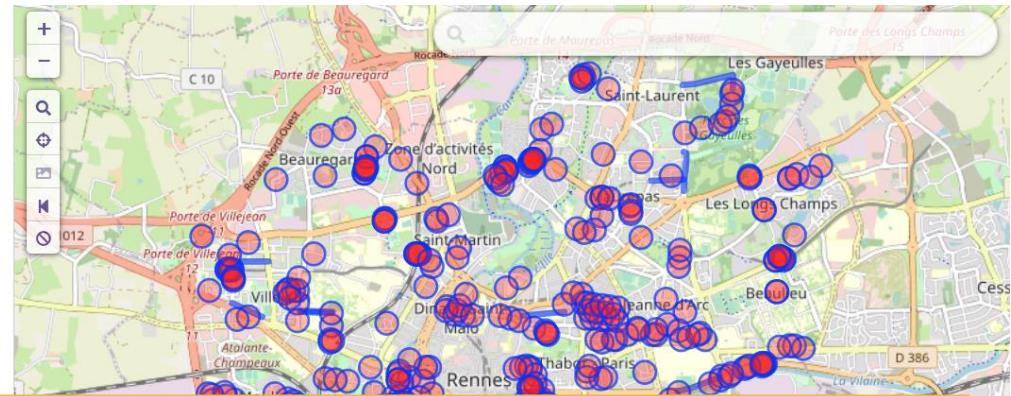
# Emprise de la recherche



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

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

```
1 [out:json][timeout:25];
2
3 area["ref:INSEE"]=="35238";
4
5 way ["highway"] ["oneway"]=="yes" ["maxspeed"]=="30"] (area);
6
7 out body;
8 >;
9 out skel qt;
```



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

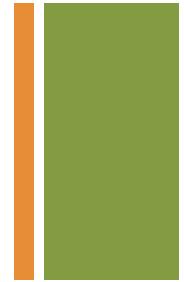
area["ref:INSEE"]=="35238";

way ["highway"] ["oneway"]=="yes" ["maxspeed"]=="30"] (area);

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



# Emprise de la recherche



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

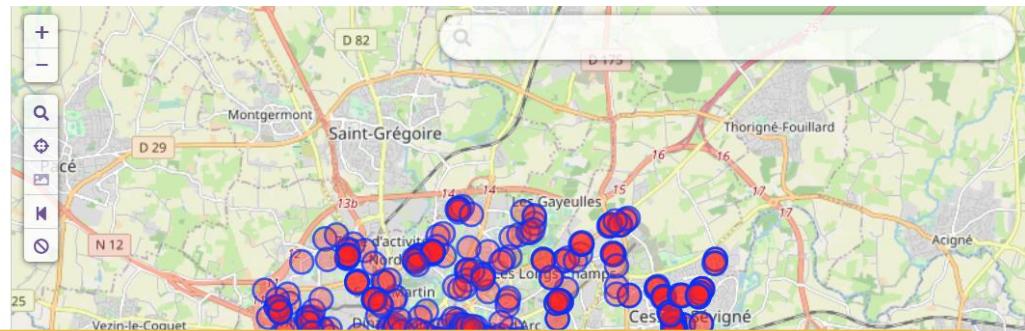
- Récupérer les **routes à sens uniques et limitées à 30km/h** à **Rennes, Saint Jaques de la Lande et Cesson Sévigné**

```
1 [out:json][timeout:25];
2
3
4 area["ref:INSEE"="35238"]; // Rennes
5 area["ref:INSEE"="35281"]; // St Jacques
6 area["ref:INSEE"="35051"]; // Cesson
7 )->.villes;
8
9 way ["highway"] ["oneway"="yes"] ["maxspeed"="30"] (area.villes);
10
11 out body;
12 >;
13 out skel qt;
14 |
```

```
( area["ref:INSEE"="35238"]; // Rennes
  area["ref:INSEE"="35281"]; // St Jacques
  area["ref:INSEE"="35051"]; // Cesson
)->.villes;

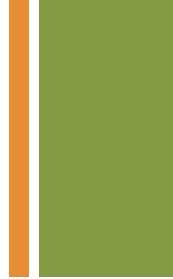
way ["highway"] ["oneway"="yes"] ["maxspeed"="30"] (area);

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





# Requêtes classiques



## Amenity

### ■ Documentation

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

Key = **highway**

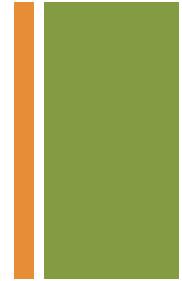
Value =

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

Key	Value	Element	Comment	Rendering	Photo
Sustenance					
amenity	bar	<input type="checkbox"/> <input checked="" type="checkbox"/>	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	<input type="checkbox"/>	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	<input type="checkbox"/> <input checked="" type="checkbox"/>	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	<input type="checkbox"/> <input checked="" type="checkbox"/>	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	<input type="checkbox"/>	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 drinking fountain or bubbler.		

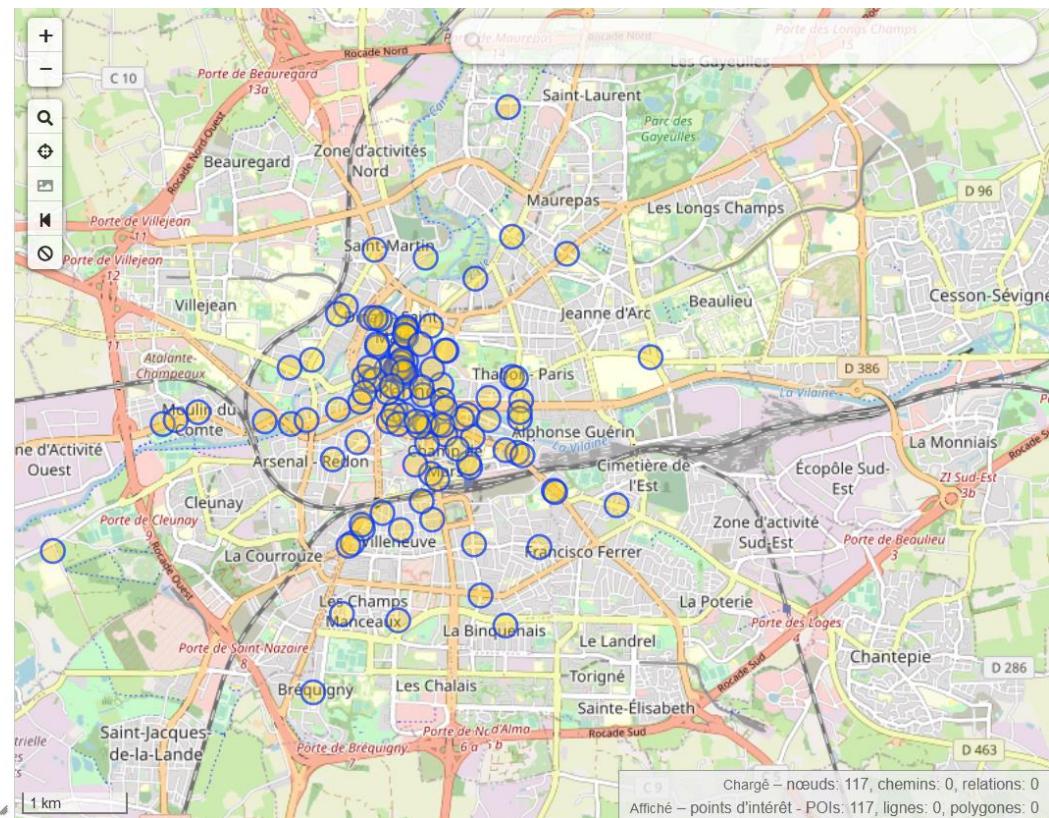


# Requêtes classiques



→ Extraire les bars

```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"bar in rennes"
*/
[out:json][timeout:25];
// fetch area "rennes" to search in
{{geocodeArea:rennes}}->.searchArea;
// get the area
// query part for: "bar"
node["amenity"="bar"](.searchArea);
.
// print results
out body;
>;
out skel qt;
```



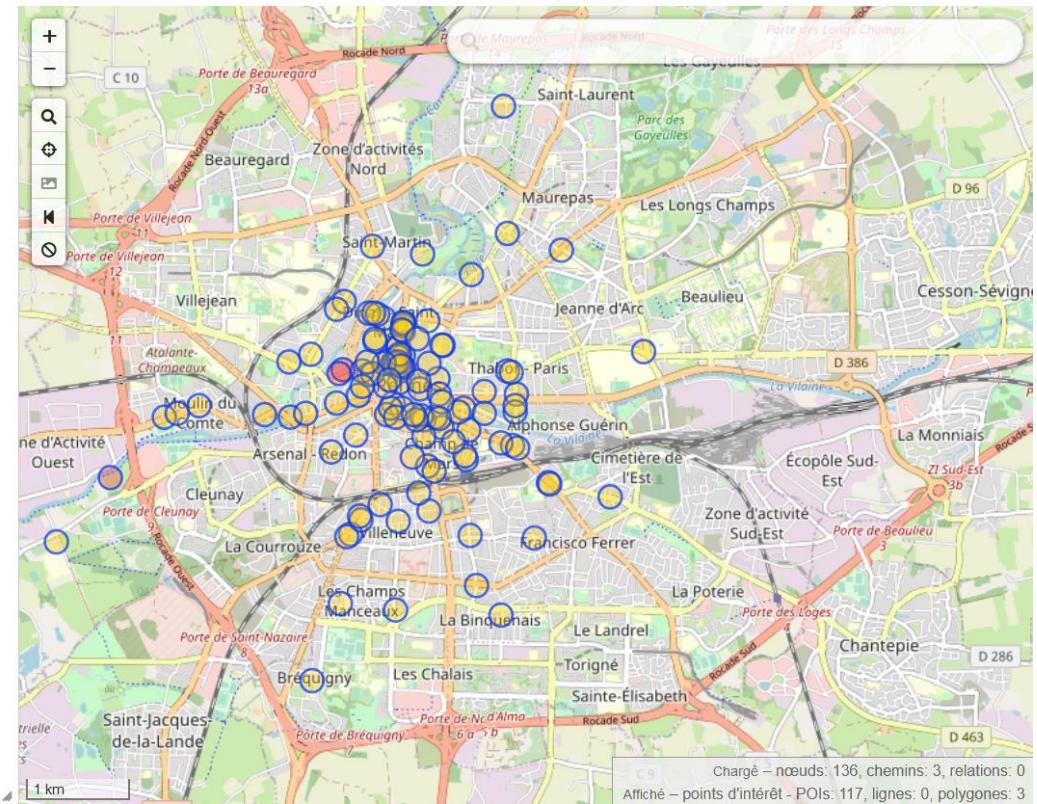


# Requêtes classiques



→ Extraire les bars (plusieurs types d'objets)

```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"bar in rennes"
*/
[out:json][timeout:25];
// fetch area "rennes" to search in
{{geocodeArea:rennes}}->.searchArea;
// gather results
// query part for: "bar"
node["amenity"]="bar"](.area.searchArea);
way["amenity"]="bar"](.area.searchArea);
relation["amenity"]="bar"](.area.searchArea);
.
// print results
out body;
>;
out skel qt;
```





# Requêtes classiques

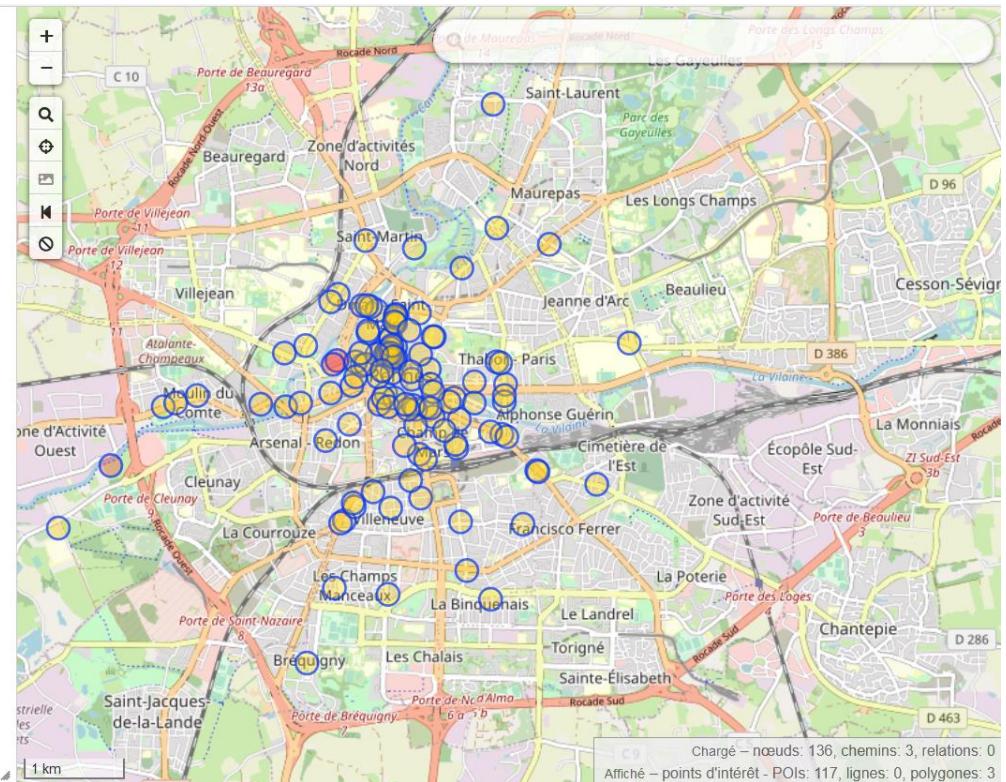


→ Extraire les bars (plusieurs types d'objets)

L'instruction **nwr** permet en une requête de chercher les nodes, ways et relations, en y appliquant les mêmes filtres

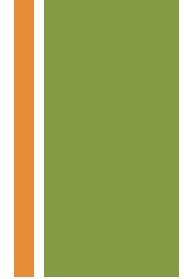
```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"bar in rennes"
*/
[out:json][timeout:25];
// fetch area "rennes" to search in
{{geocodeArea:rennes}}>>.searchArea;
// gather results
);
// query part for: "bar"
nwr["amenity"]="bar"](.searchArea);
// print results
out body;
>;
out skel qt;
```

<https://overpass-turbo.eu/s/1Npj>





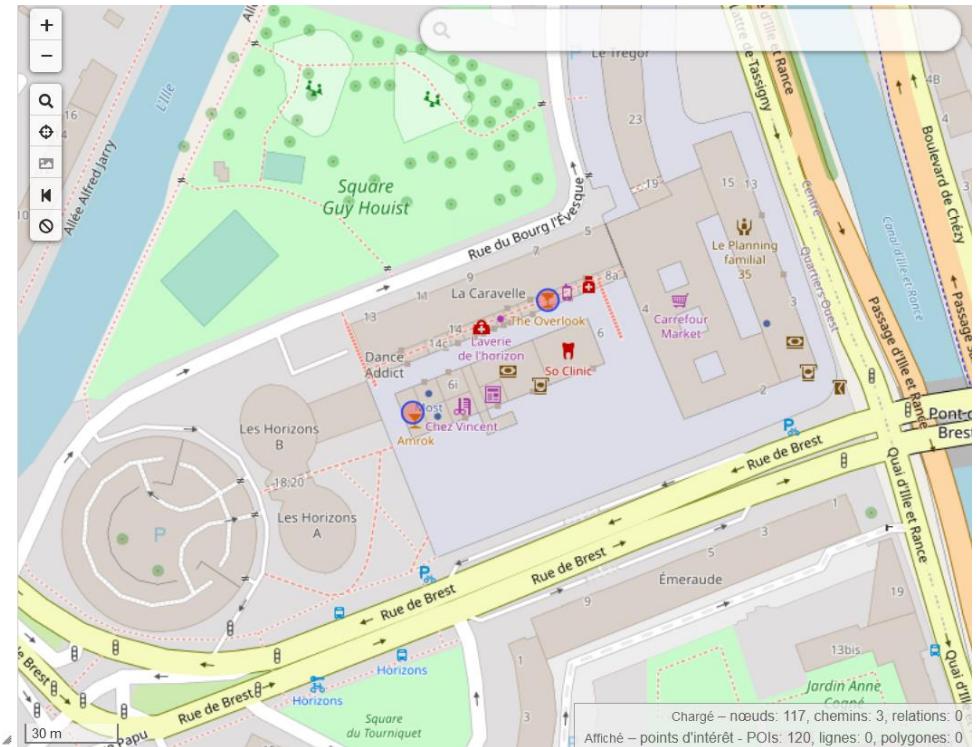
# Requêtes classiques



→ Extraire les bars (plusieurs types d'objets)

L'instruction **out center** permet de transformer les ways et relations en nodes (centroïdes)

```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"bar in rennes"
*/
[out:json][timeout:25];
// fetch area "rennes" to search in
{{geocodeArea:rennes}}->.searchArea;
// gather results
(
  // query part for: "bar"
  nwr["amenity"="bar"](.searchArea);
)
// print results
out center;
```





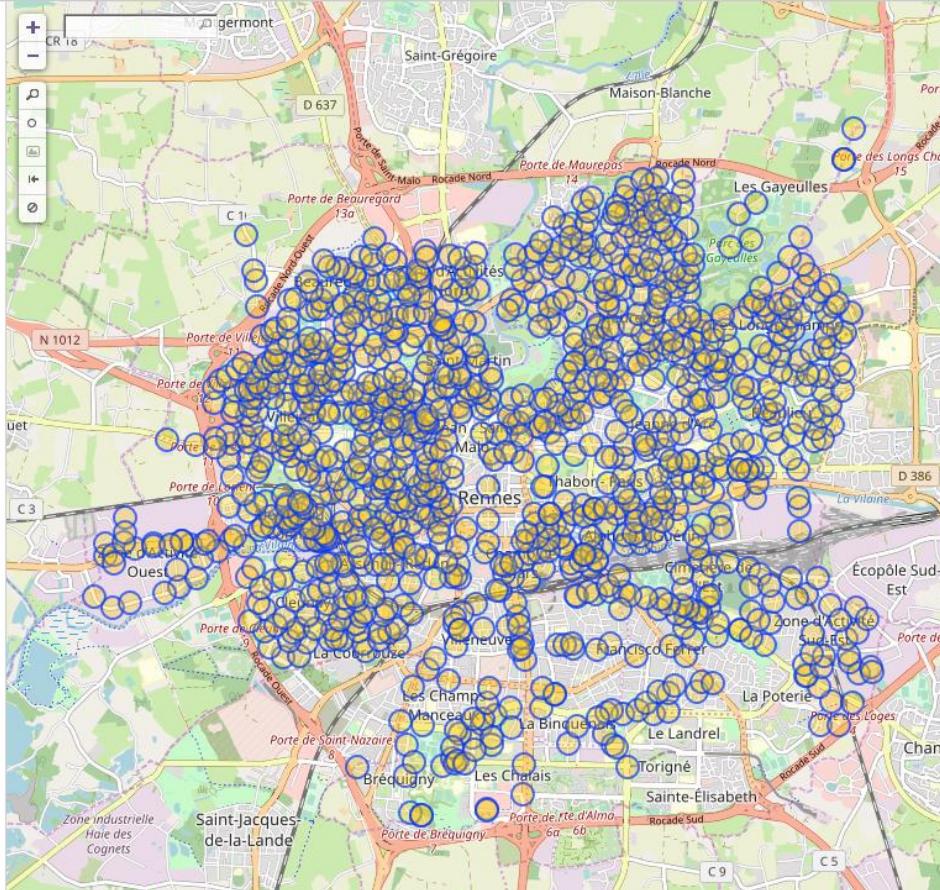
# Requêtes classiques



## Extraire les bornes incendies

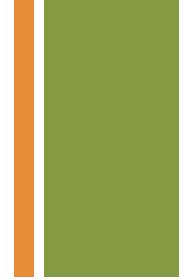
Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo ⚙

```
1  /*
2  This has been generated by the overpass-turbo wizard.
3  The original search was:
4  "bar 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: "bar"
12   node["emergency"="fire_hydrant"] (area.searchArea);
13 }
14 // print results
15 out body;
16 >;
17 out skel qt;
```





# Requêtes classiques



## Extraire plusieurs type d'objets

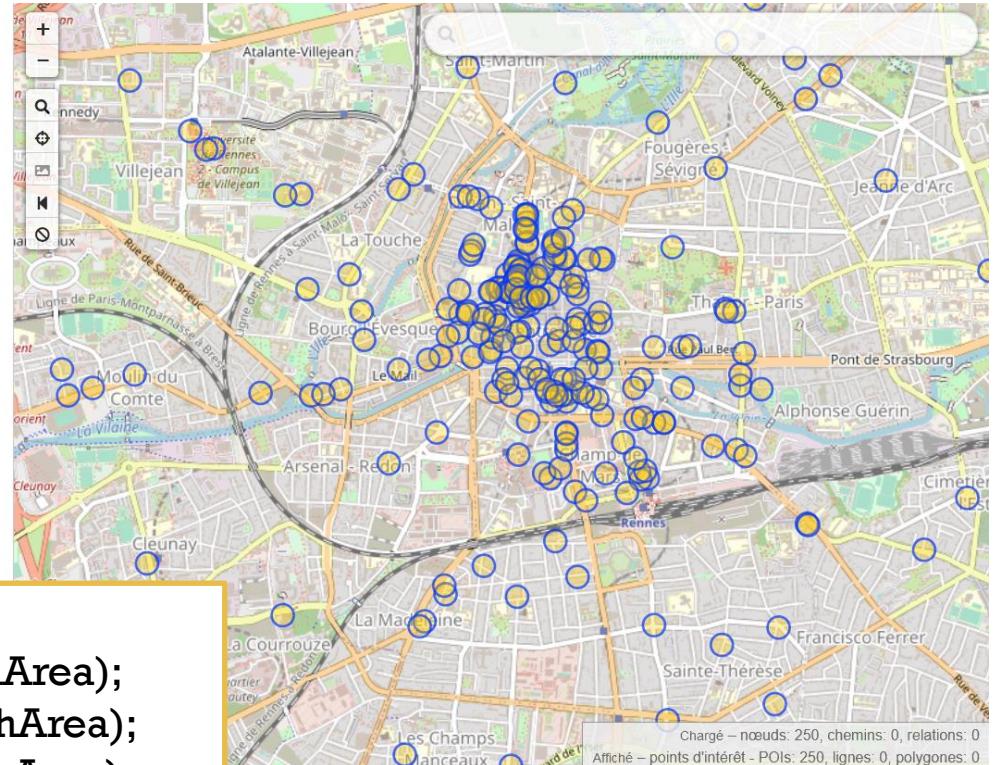
```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"bar in rennes"
*/
[out:json][timeout:25];

// fetch area "rennes" to search in
{{geocodeArea:rennes}}->.searchArea;

(
node["amenity"]="bar"](area.searchArea);
node["amenity"]="cafe"](area.searchArea);
node["amenity"]="pub"](area.searchArea);
);

// print results
out center;
```

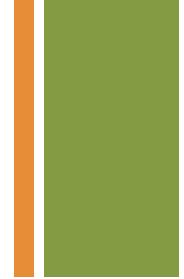
```
(  
node["amenity"]="bar"](area.searchArea);  
node["amenity"]="cafe"](area.searchArea);  
node["amenity"]="pub"](area.searchArea);  
);
```



<https://overpass-turbo.eu/s/1Npk>



# Requêtes classiques



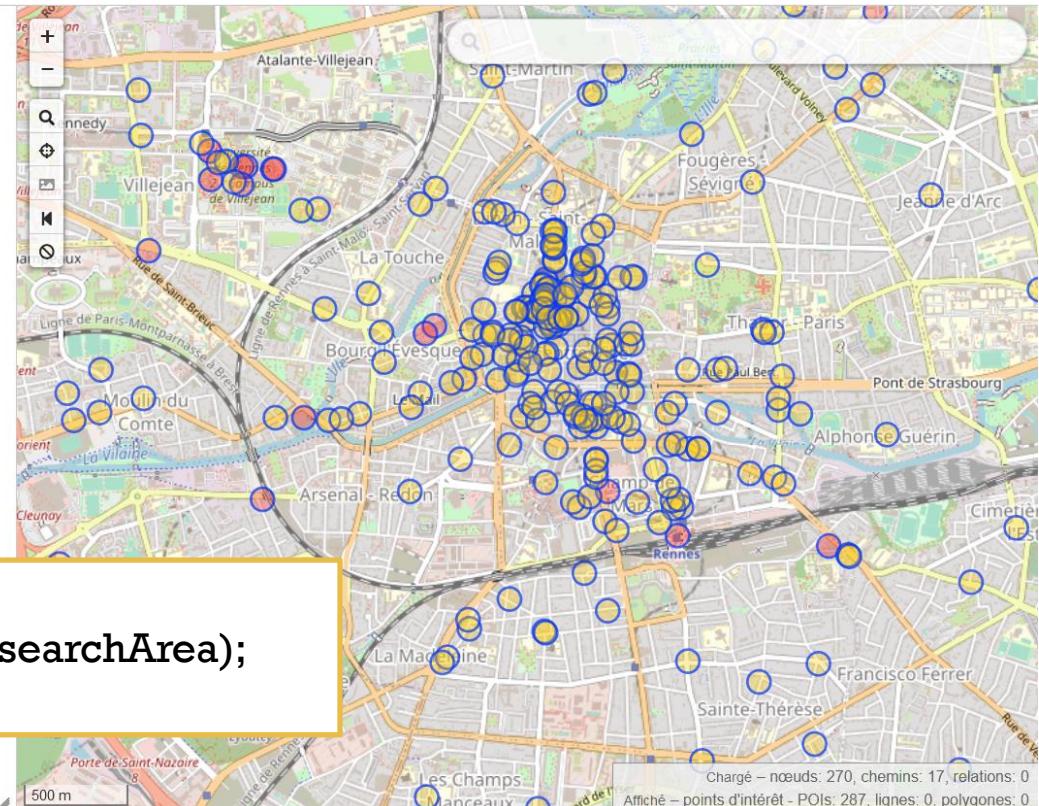
Extraire plusieurs type d'objets avec la même clef en simplifiant la requête au maximum

```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"bar in rennes"
*/
[out:json][timeout:25];

// fetch area "rennes" to search in
{{geocodeArea:rennes}}->.searchArea;

(
    // query part for: "bar, cafe et pub "
    nwr[amenity~"(bar|cafe|pub)"](.searchArea);
);

// print results
out center;
```

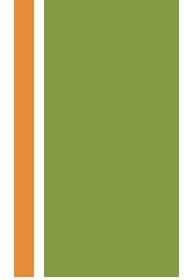


```
(  
nwr[amenity~"(bar|cafe|pub)"](.searchArea);  
);
```

<https://overpass-turbo.eu/s/1j83>



# Requêtes classiques



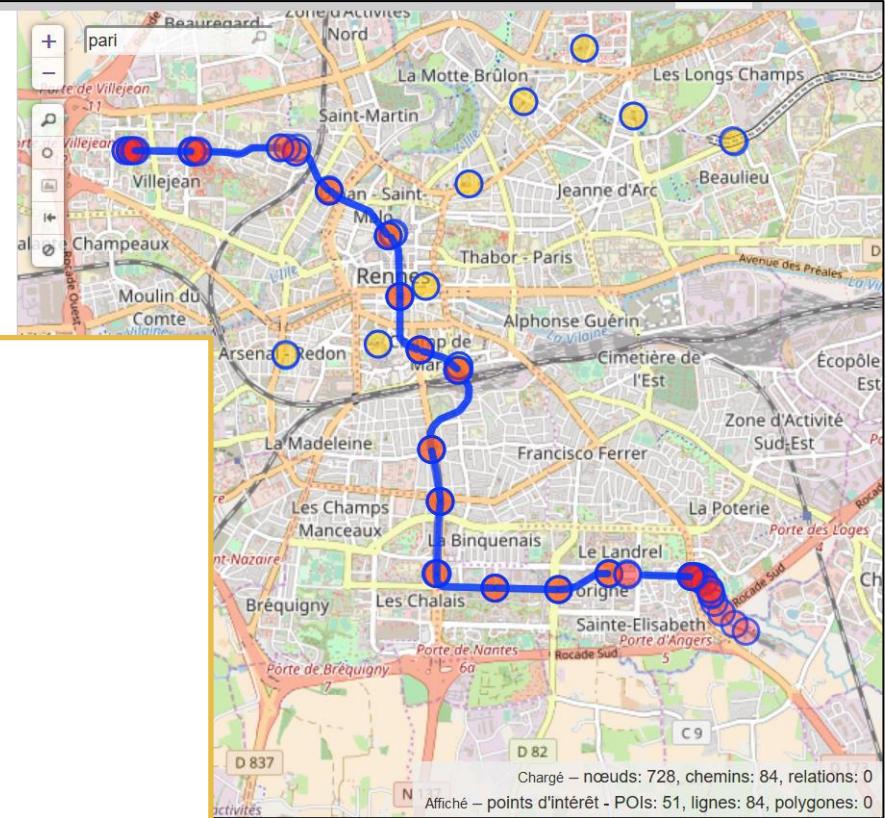
Extraire plusieurs type d'objets

```
1 [out:json][timeout:25];
2 {{geocodeArea:rennes}}->.searchArea;
3
4 (node["public_transport"="stop_position"] ["subway"="yes"] (area.searchArea);
5 way["railway"="subway"] (area.searchArea);
6 );
7
8 out body;
9 >;
10 out skel qt;
```

```
[out:json][timeout:25];
{{geocodeArea:rennes}}->.searchArea;

(node["public_transport"="stop_position"]
["subway"="yes"])(area.searchArea);
way["railway"="subway"])(area.searchArea);
);

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





# Exports des données



Exporter ×

Données

GeoJSON	<a href="#">télécharger</a>	<a href="#">copier</a>
GPX	<a href="#">télécharger</a>	<a href="#">copier</a>
KML	<a href="#">télécharger</a>	<a href="#">copier</a>
données brutes de OSM	<a href="#">télécharger</a>	<a href="#">copier</a>

données brutes depuis l'[API Overpass](#) ↗  
charger les données dans un éditeur externe OSM : [JOSM](#), [Level0](#) ↗

## Carte

en [image png](#)

en [Carte interactive](#)

Vue actuelle sur [OpenStreetMap.org](#) (zone de recherche, centre, etc.)



# Exports de la requête



## Requête

requête autonome

<a href="#">télécharger</a>	<a href="#">copier</a>
-----------------------------	------------------------

requête brute

<a href="#">télécharger</a>	<a href="#">copier</a>
-----------------------------	------------------------

wiki d'osm

<a href="#">télécharger</a>	<a href="#">copier</a>
-----------------------------	------------------------

URL de données distante

umap

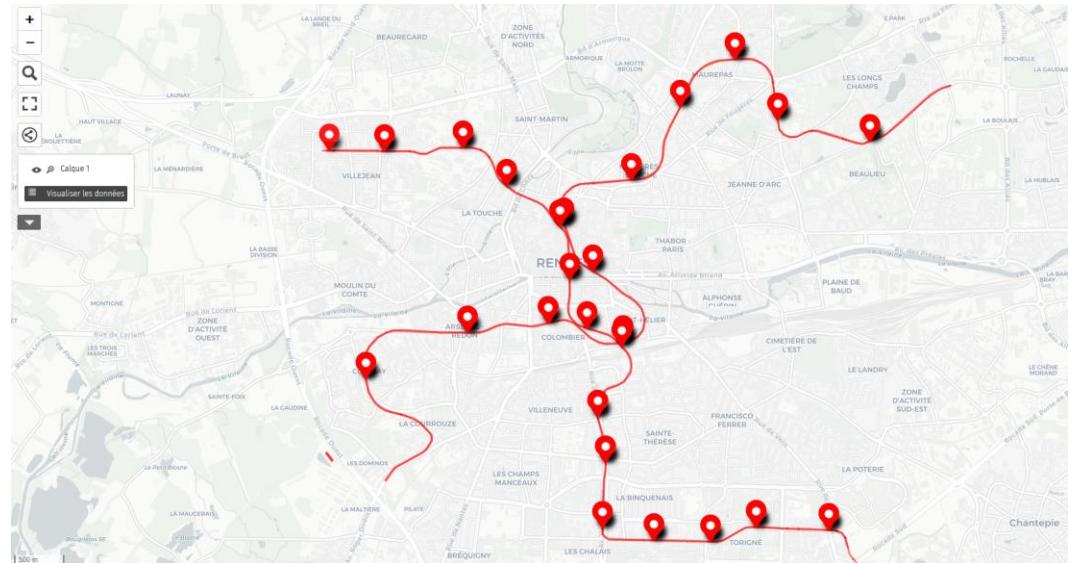
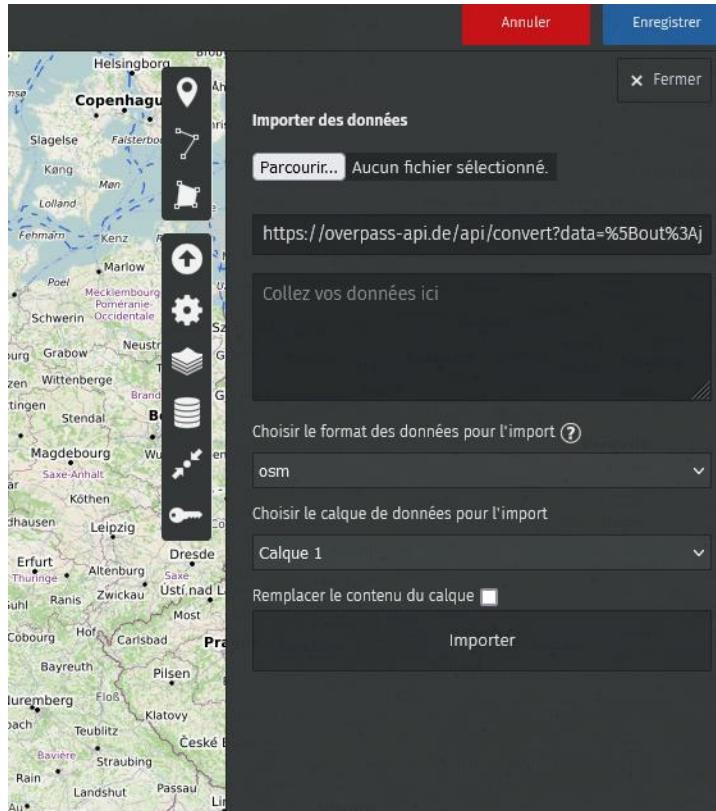
<a href="#">télécharger</a>	<a href="#">copier</a>
-----------------------------	------------------------

en [Overpass-XML ↗](#)

en [OverpassQL ↗](#) ([compact ↗](#))

# Exports de la requête

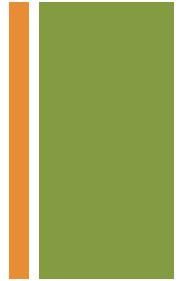
## Intégrer les données d'une requête au sein d'une carte Umap



[https://umap.openstreetmap.fr/fr/map/metro-lyon\\_1087653](https://umap.openstreetmap.fr/fr/map/metro-lyon_1087653)



# Requêtes classiques



## 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 [\(en\)](#).

#### `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 (someti

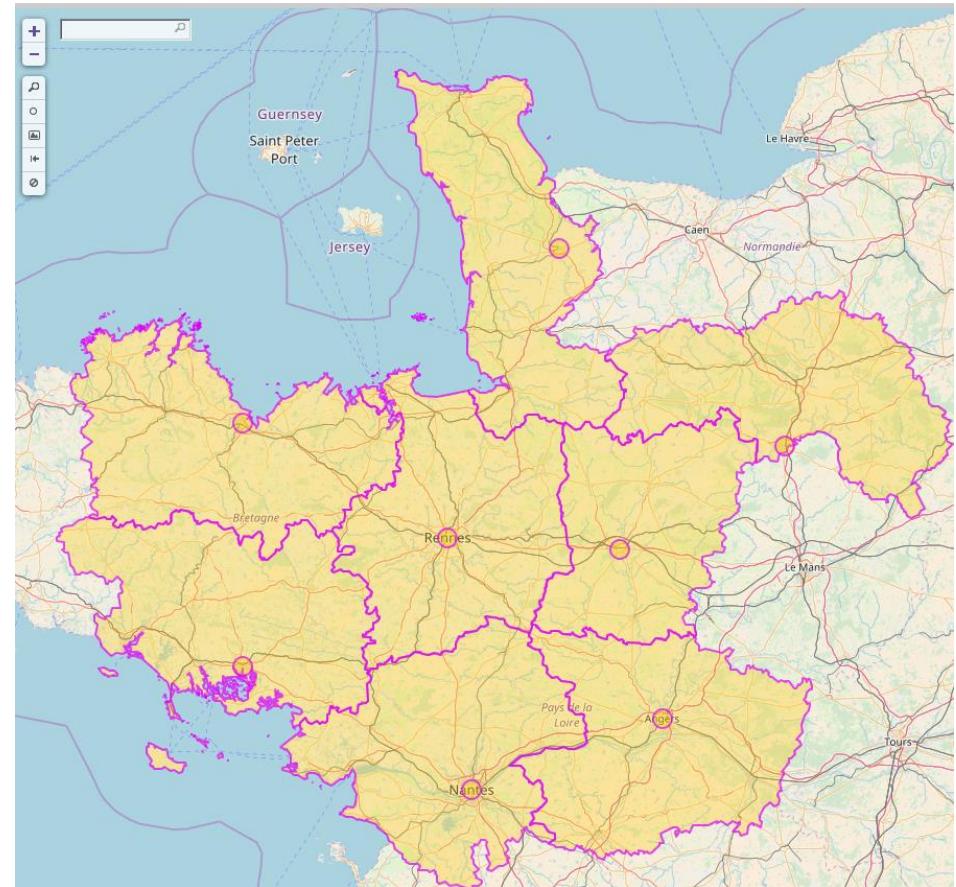


# Requêtes classiques



## Limites administratives

- Régions = 4
- Départements = 6
- Arrondissements= 7
- Communes = 8
- Quartiers = 9





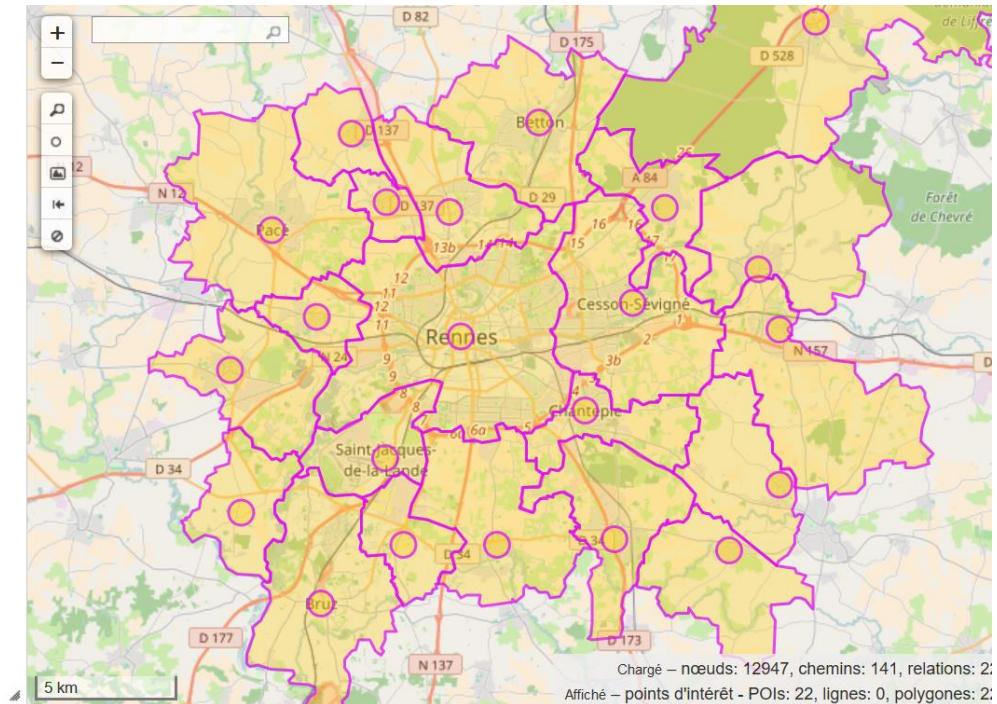
# Requêtes classiques



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;
```



<https://overpass-turbo.eu/s/z8w>



# Requêtes classiques

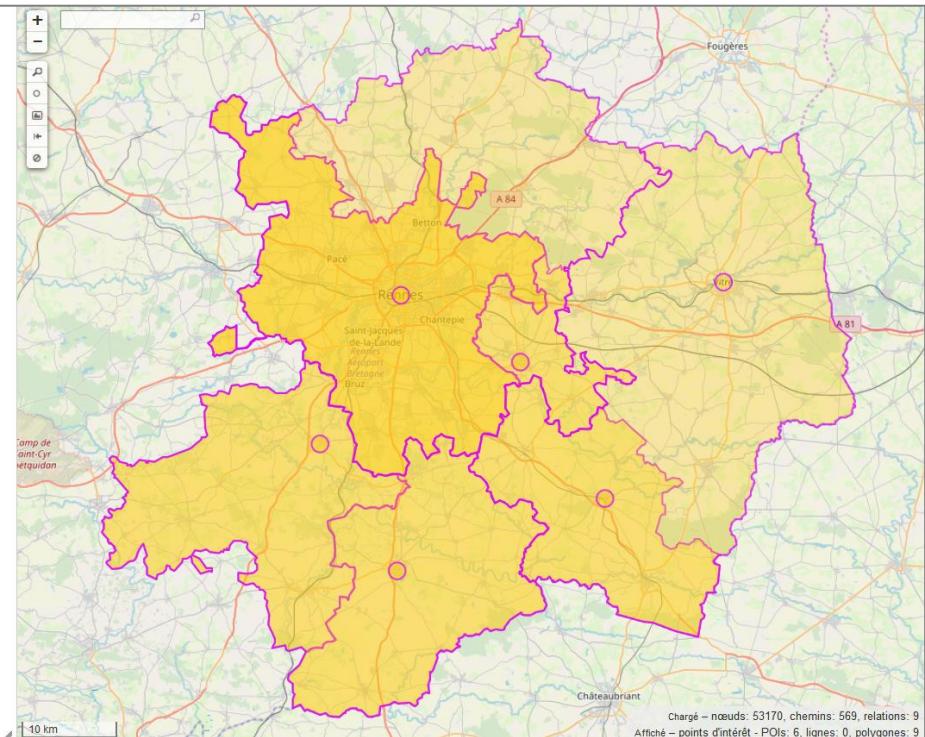


Obtenir les contours des EPCI

*Tag : boundary=local\_authority*

[https://wiki.openstreetmap.org/wiki/Tag:boundary%3Dlocal\\_authority](https://wiki.openstreetmap.org/wiki/Tag:boundary%3Dlocal_authority)

```
[out:json][timeout:25];
relation ["boundary"]=="local_authority"]((bbox));
// print results
out body;
>;
out skel qt;|
```



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



# Requêtes classiques



## Landuse

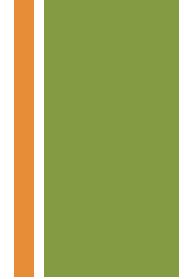
### ■ Documentation

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

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

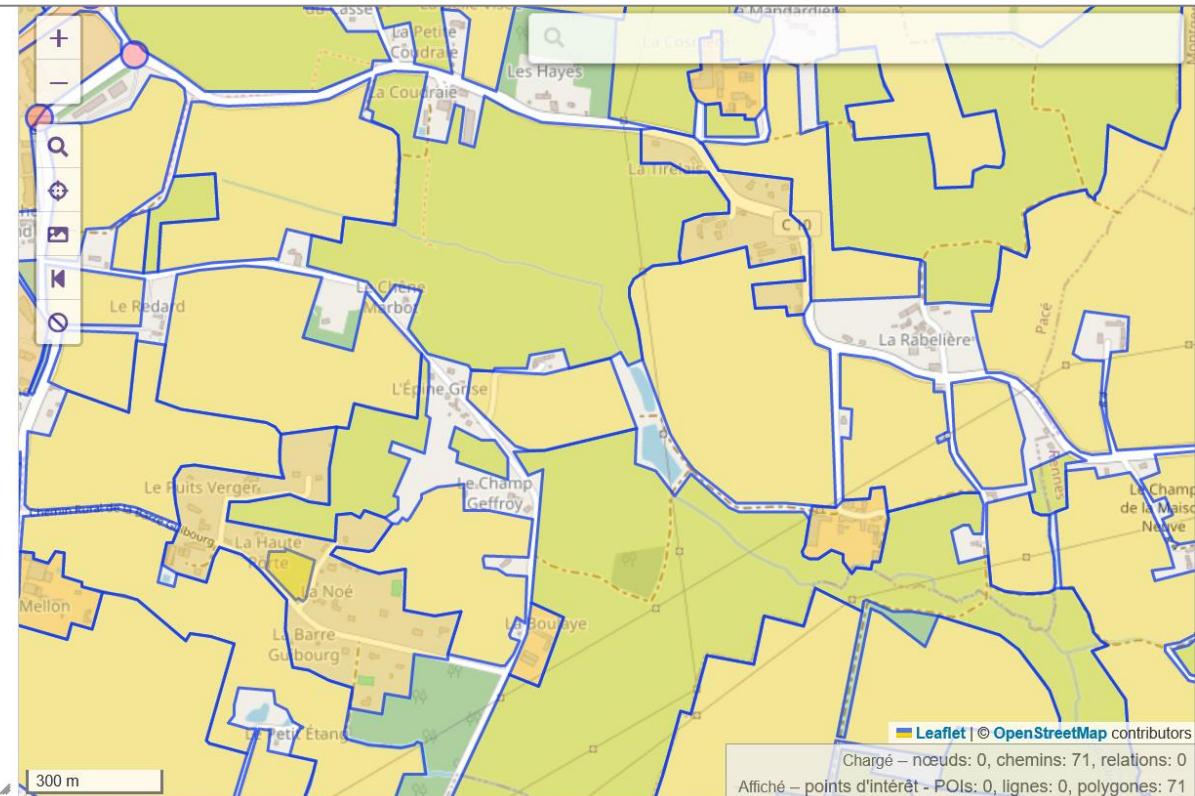


# Requêtes classiques



Extraire les zones renseignées sur l'occupation des sols

```
1 [out:json][timeout:25];
2
3 // gather results
4 nwr["landuse"]({{bbox}});
5
6 // print results
7 out geom;
```



<https://overpass-turbo.eu/s/1Ns2>

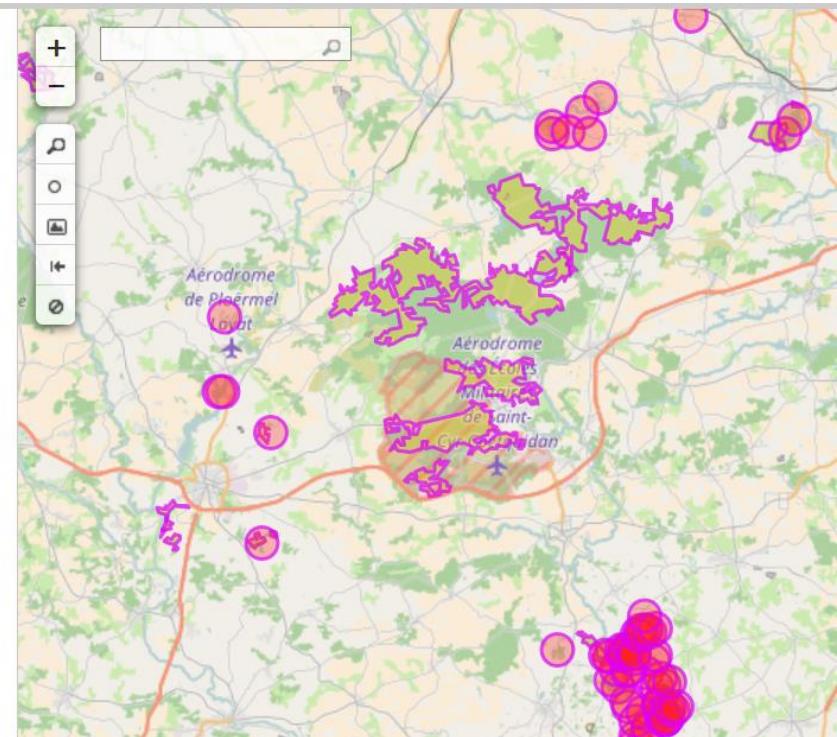


# Requêtes classiques



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;
```





# Requêtes classiques



## Mobiliser les ID de Wididata

<https://fr.wikipedia.org/wiki/Wikidata>

Wiki Loves Monuments: Photograph a monument, help Wikipedia and win!

[Learn more](#)

**IKEA (Q54078)**

Trademark used for retail of furniture, appliances, and home furnishings

[edit](#)

▼ In more languages

Configure

Language	Label	Description	Also known as
English	IKEA	Trademark used for retail of furniture, appliances, and home furnishings	
French	IKEA	entreprise de mobilier et d'objets de décoration fondée en Suède	Ingvar Kamprad Elmtaryd Agun...
Spanish	IKEA	empresa de fabricación de muebles y distribución, fundada en Suecia	
German	IKEA	multinationaler Einrichtungskonzern	

All entered languages

Statements

Wikipedia (79 entries) [edit](#)

- af IKEA
- als IKEA
- an IKEA
- ar إيكيا
- ast IKEA
- azb ایکیا
- az IKEA
- ba IKEA
- be\_x\_old IKEA
- be IKEA
- bg IKEA
- br Ikea
- bs Ikea
- ca Ikea

<https://www.wikidata.org/wiki/Q54078>



# Requêtes classiques



Mobiliser les ID de Wididata > magasins IKEA

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

Carte Données

```
1 /*  
2 This has been generated by the overpass-turbo wizard.  
3 The original search was:  
4 "ikea"  
5 */  
6 [out:json] [timeout:25];  
// gather results  
7 (  
8   // query part for: "ikea"  
9   node["brand:wikidata"="Q54078"]({{bbox}});  
10  way["brand:wikidata"="Q54078"]({{bbox}});  
11 );  
// print results  
12 out body;  
13 >;  
14 out skel qt;
```

Chargé – noeuds: 3012, chemins: 146, relations: 0  
Affiché – points d'intérêt - POIs: 49, lignes: 0, polygones: 146

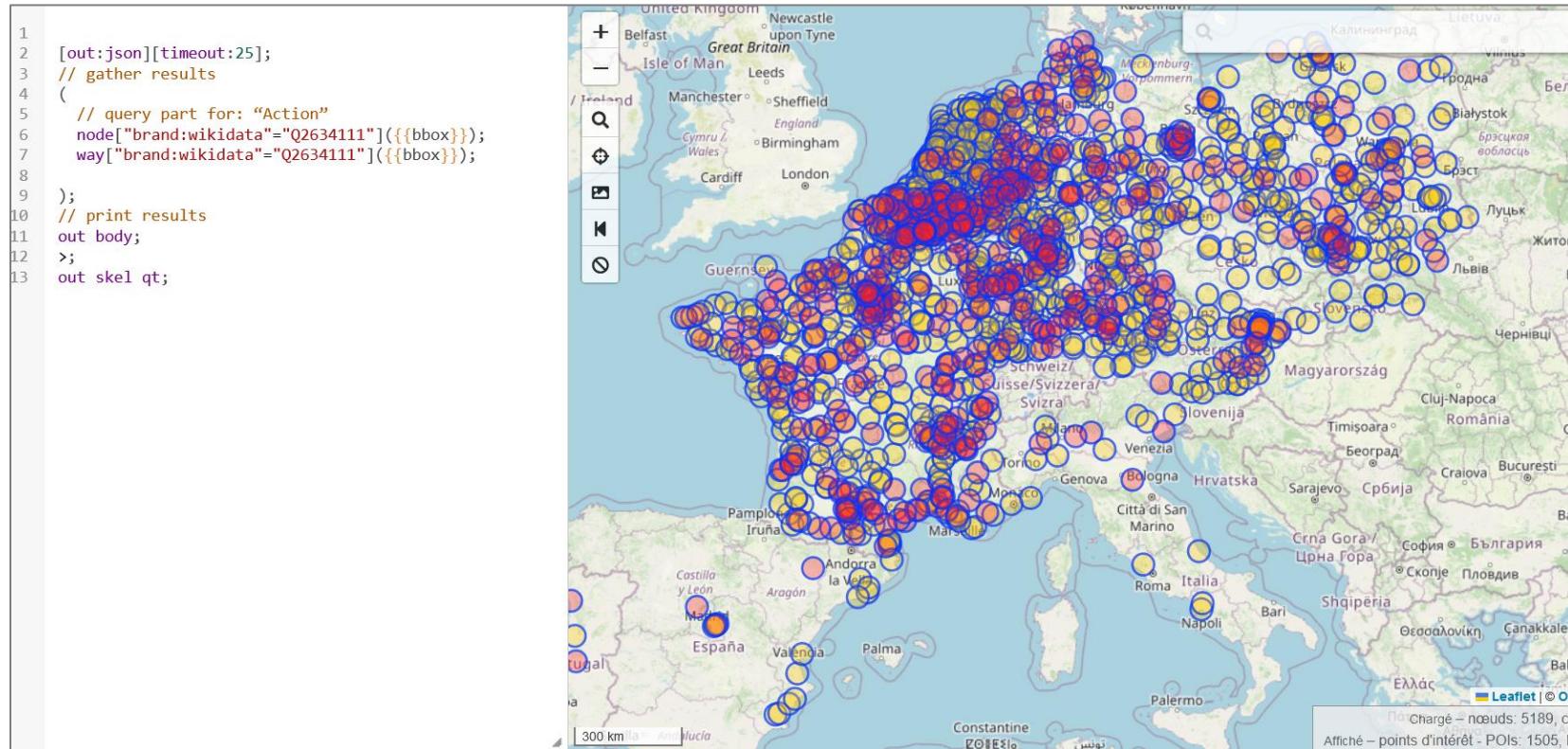
<https://t.co/kmZ8am31j6?amp=1>



# Requêtes classiques



Mobiliser les ID de Wididata > magasins Actions



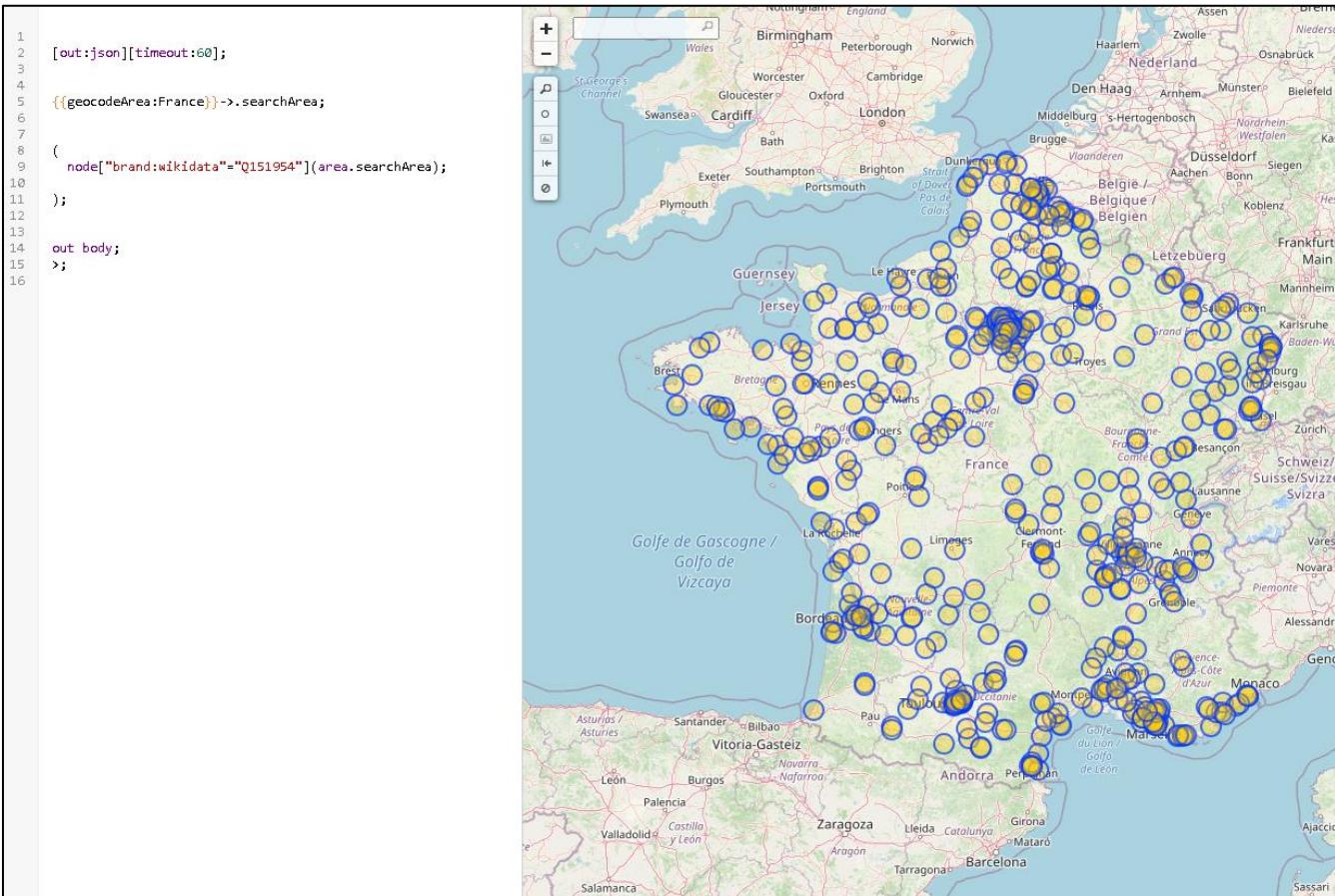
<https://overpass-turbo.eu/s/1Nrr>



# Requêtes classiques



## Tester les magasins LIDL en France

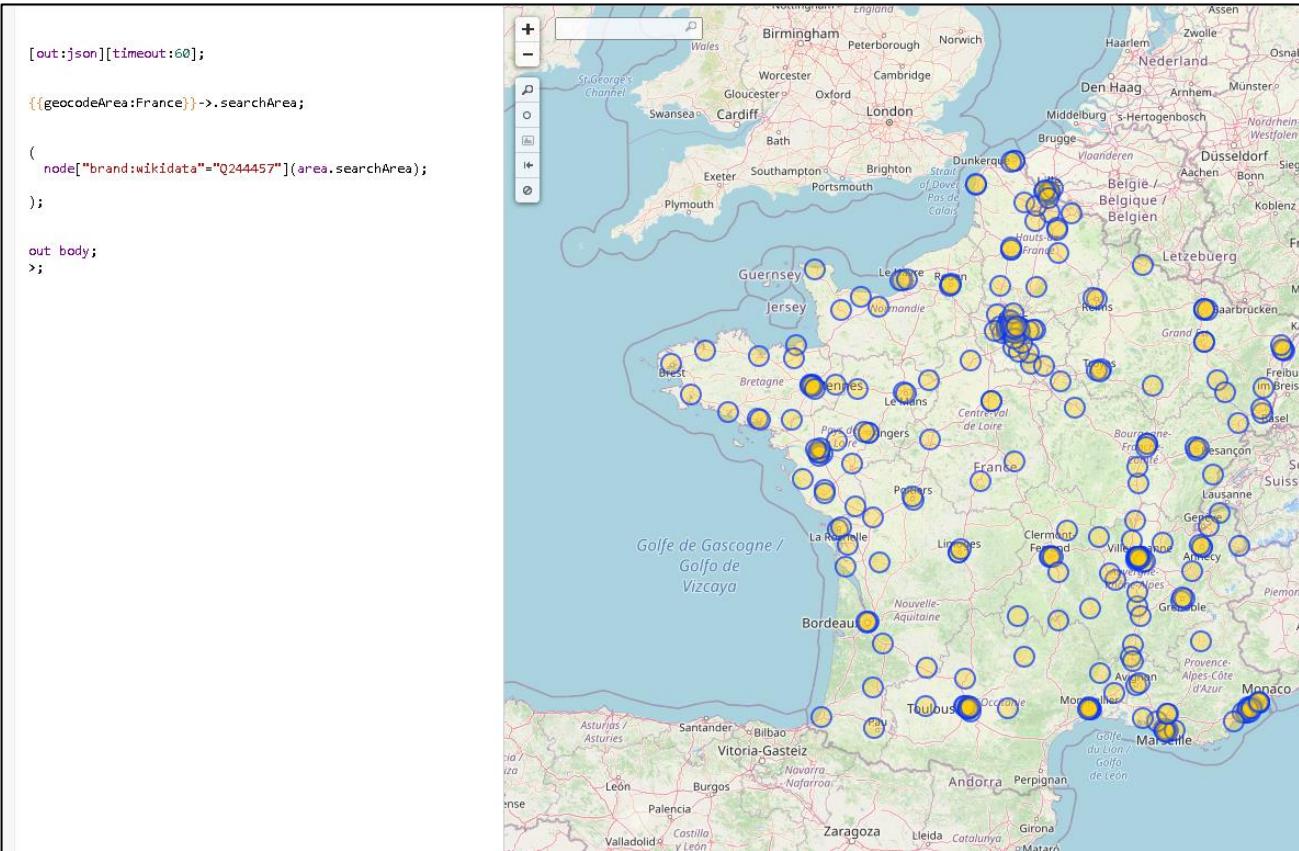




# Requêtes classiques



## Les restaurants Subway en France

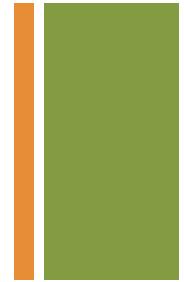


+

# Requêtes complexes



# Requêtes complexes



## Critère minimum ou maximum

- Toutes les routes avec une vitesse limite supérieure à 50km/h

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

```
[out:json][timeout:25];
// gather results

(way[highway]
(if: (is_number(t["maxspeed"]) && t["maxspeed"] > 50))
({{bbox}});
);
// print results
out body;
>;
out skel qt;
```

The screenshot shows the Overpass Turbo interface with a map of a French town. Numerous blue lines representing highways are highlighted, and red circular markers are placed along these lines at various points, likely indicating specific locations of interest or waypoints defined in the query. The map includes labels for various streets and landmarks.

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

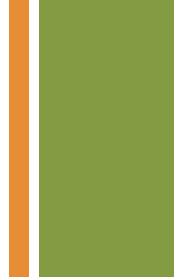
  

```
(way["highway"]
(if: (is_number(t["maxspeed"]) && t["maxspeed"] > 50))
({{bbox}});
);
out body;
>;
out skel qt;
```

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



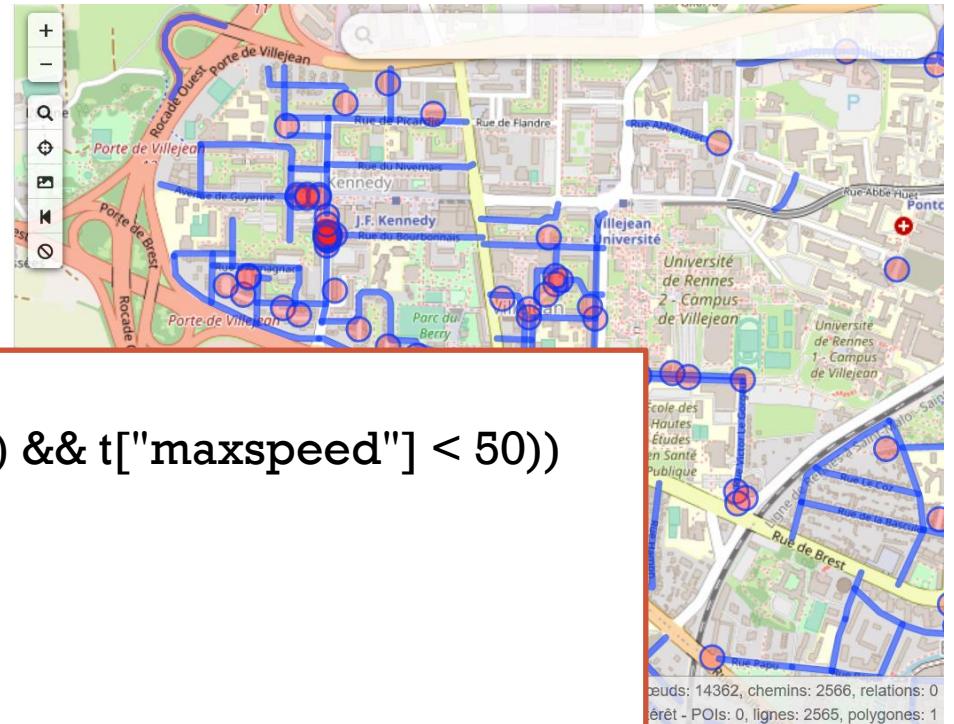
# Requêtes complexes



## Critère minimum ou maximum

- Toutes les routes avec une vitesse inférieure à 50km/h

```
1 (way["highway"]
2   (if: (is_number(t["maxspeed"]) && t["maxspeed"] < 50))
3   ({bbox}));
4 );
5 out body;
6 >;
7 out skel qt;
```



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



# Requêtes complexes



## Ajouter un critère de nombre de voies

- Toutes les routes avec une vitesse maximale de 50km/h et avec au moins deux voies (*lanes*)

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

Carte Données

```
1 (way["highway"]
2   (if: (is_number(t["lanes"])) && t["lanes"] > 1) ({{bbox}}));
3 );
4 (way["highway"]
5   (if: (is_number(t["maxspeed"])) && t["maxspeed"] > 50) ({{bbox}}));
6 );
7 out body;
8 >;
9 out skel qt;
10 |
```

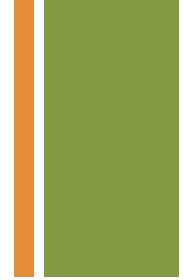
(way["highway"]  
  (if: (is\_number(t["lanes"])) && t["lanes"] > 1) ({{bbox}}));  
(way["highway"]  
  (if: (is\_number(t["maxspeed"])) && t["maxspeed"] > 50) ({{bbox}}));  
);  
out body;  
>;  
out skel qt;

2 km

Affiché – points d'intérêt - POIs: 0. lignes: 775. polyvalones: 0

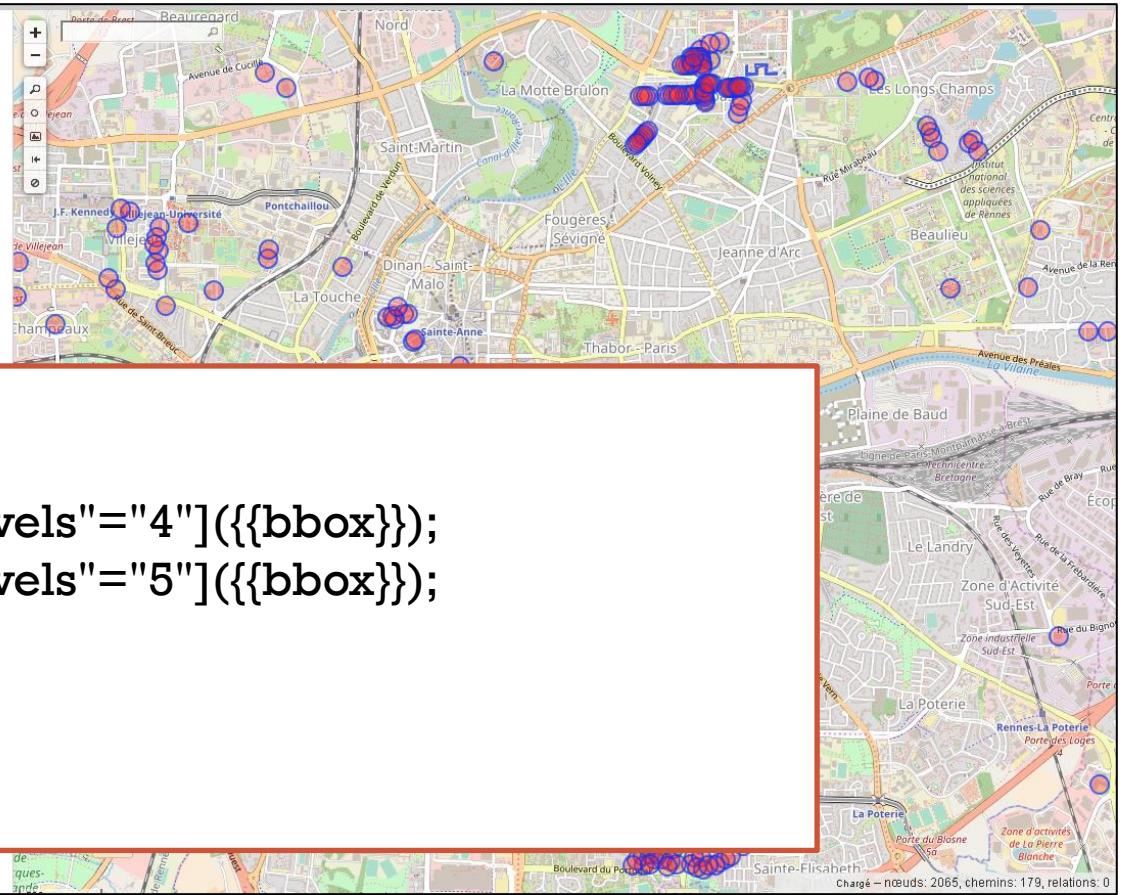


# Requêtes complexes



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;
```





# 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"](.searchArea);
  way["building"="yes"](.searchArea);
  relation["building"="yes"](.searchArea);
);
out count;
```

<https://overpass-turbo.eu/s/1Nrl>

	@count	@count:nodes	@count:ways	@count:relations
1				
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;
9
```

	@count	@count:nodes	@count:ways	@count:relations
1				
2	23862	6792	17049	21
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;
7
```

	@count	@count:nodes
1		
2	112	112
3		



# Comptage d'entités

Compter le nombre de pharmacies par communes dans le 35

```
[out:csv(_row;false)];  
  
//Toutes les communes du département du 35  
area["ref:INSEE"~"^35"];  
  
// Compte des pharmacies par communes  
foreach->.region(  
    // Collect all Nodes, Ways and Relations wth amenity=pharmacy in the current  
    area  
    ( node(area.region)[amenity=pharmacy];  
      way(area.region)[amenity=pharmacy];  
      rel(area.region)[amenity=pharmacy];) -> .r;  
  
    make out _row = " | -"; out;  
    make out _row = " | " + region.set(t[ "de:regionalschluessel"]) +  
        " | " + region.set(t["name"]) +  
        " | " + (r.count(nodes) + r.count(ways) + r.count(relations));  
    out;  
);  
make out _row = " | }"; out;
```

```
1 | - || Ille-et-Vilaine || 253  
2 | - || Redon || 25  
3 | - || Rennes || 152  
4 | - || Bain-de-Bretagne || 8  
5 | - || Guichen || 7  
6 | - || Montfort-sur-Meu || 8  
7 | - || Pipriac || 3  
8 | - || Plélan-le-Grand || 4  
9 | - || Redon || 6  
10 | - || Saint-Aubin-d'Aubigné || 6  
11 | - || Sixt-sur-Aff || 1  
12 | - || Bains-sur-Oust || 1  
13 | - || Redon || 4  
14 | - || Bruc-sur-Aff || 0  
15 | - || Pipriac || 2  
16 | - || Saint-Ganton || 0  
17 | - || Saint-Just || 0  
18 | - || Guipry || 0  
19 | - || Renac || 0  
20 | - || Sainte-Marie || 1  
21 | - || La Chapelle-de-Brain || 0  
22 | - || Langon || 0  
23 | - || Guipry-Messac || 2  
24 | -
```



# Exports sous formes de tableaux de données

Exporter une liste des bars de Rennes avec leurs horaires

```
1 [out:csv(bar, name, "opening_hours")];
2 area[admin_level=8][name="Rennes"];
3 nwr[amenity = bar]["opening_hours"](|area);
4 out;
```

```
1 bar name opening_hours
2 Combi Bar Tu-Sa 19:00-01:00
3 L'Heure du Jeu Mo-Fr 17:00-01:00; Sa 14:00-01:00; Su 14:00-22:00
4 Le Vincennes "Mo-Fr 07:00-19:45; Sa 08:00-19:00; PH,Su 08:30-13:00"
5 Le Papier Timbré Mo-Fr 16:30-01:00; Sa 12:00-01:00
6 La Maison Tu-Sa 11:00-01:00
7 Chez Marco Mo-Fr 11:00-15:00
8 Le café de Rennes Mo-Th 09:00-01:00; Fr-Sa 09:00-03:00; Su 09:00-01:00
9 L'Amaryllis Tu-Fr 16:00-01:00; Sa 11:00-01:00
10 Le Jules Ferry Mo-Fr 06:30-20:00; Sa 07:30-12:00
11 WarpZone Mo 17:00-22:00; Tu-Fr 17:00-01:00; Sa 14:00-01:00; Su 14:00-22:00
12 Les Grands Gamins Mo 10:00-23:00; Tu-Fr 10:00-01:00; Sa 11:00-01:00; Su 11:00-23:00
13 Black Bear Mo-Su 16:00-01:00
14 Le Beret Mo-Fr 10:00-01:00; Sa 10:30-01:00
15 L'attrape-rêve Tu-Sa 16:00-01:00; Su 16:00-23:00
16 La Bolée Tu-Fr 11:30-01:00
17 Nakama Mo-Sa 17:00-01:00
18 Le Saint-Martin Mo-Fr 06:30-20:00; Su 08:00-14:00
19 Le Since 42 Tu-Sa 10:42-00:42
```

```
[out:csv(bar, name, opening_hours)];
area[admin_level=8][name="Rennes"];
nwr[amenity = bar]["opening_hours"](|area);
out;
```

```
31 Melody NELSON Tu-Sa 19:00-05:00
32 Chez ma tante Mo-Tu 07:00-22:30; Fr-Sa 09:00-23:30
33 Barex'po "Mo,Tu 12:00-23:00; We 12:00-00:00; Th-Sa 12:00-01:00; Su 11:00-15:00"
34 La Piste Mo-Sa 11:00-01:00; Su 11:00-22:00
35 Le Gabier Noir "We,Th 11:30-14:00,18:00-21:00; Fr 11:30-14:00,18:00-00:30; Sa 18:00-00:30"
```



# Exports sous formes de tableaux de données

Exporter une liste des pharmacies de Rennes, accessibles aux fauteuils roulants avec leurs horaires

```
1 [out:csv(name, "opening_hours", "wheelchair")];
2 area[admin_level=8][name="Rennes"];
3 nwr[amenity = pharmacy]["opening_hours"] ["wheelchair"](area);
4 out;
```

```
1 name      opening_hours    wheelchair
2 Pharmacie de la Poterie "Mo-Th 09:00-12:30,14:30-19:30, Fr 09:00-19:30, Sa 09:00-18:30" yes
3 Pharmacie de Maurepas   "Mo-Fr 09:00-13:00,14:00-19:30; Sa 09:00-13:00,14:00-19:00" no
4 Pharmacie de Bretagne   Mo-Fr 09:00-20:00; Sa 10:00-19:00  no
5 Pharmacie Nguyen        "Mo-Fr 09:00-12:30,14:00-19:30; Sa 09:00-12:30,14:00-18:00" yes
6 Pharmacie Pascal        "Mo-Fr 08:30-20:00; Sa 09:00-13:00,14:30-20:00" yes
7 Pharmacie Clemenceau   "Mo-Fr 09:00-12:30, 14:00-19:00; Sa 09:00-12:30, 14:00-18:00" yes
8 Pharmacie Anatole France "Mo 14:00-19:30; Tu-Fr 09:00-19:30; Sa 09:00-12:30,14:00-17:00" yes
9 Pharmacie Square de la Rance "Mo 09:30-12:30,14:00-19:30; Tu-Fr 09:00-12:30,14:00-19:30; Sa
10 Pharmacie de la Madeleine "Mo-Fr 09:00-13:00,14:00-20:00; Sa 09:30-13:30" yes
11 Pharmacie de Villejean  "Mo-Fr 08:30-19:30; Sa 09:00-13:00,14:00-19:00" limited
12 Pharmacie Métro Jacques Cartier "Mo-Fr 08:30-13:00,14:00-19:30; Sa 09:00-13:00" yes
13 Pharmacie Saint-Hélier   "Mo 14:00-19:30; Tu-Fr 09:00-12:30,14:00-19:30; Sa 09:30-12:30,14:30-18
14 Pharmacie Beauregard    Mo-Fr 09:00-19:30; Sa 09:00-13:00  yes
15 Pharmacie du Pont de Nantes "Mo-Fr 09:00-12:30,14:00-19:30; Sa 09:00-12:30,14:00-18:00" no
16 Pharmacie du Verger      "Mo-Fr 09:00-12:30,14:00-19:30; Sa 09:00-13:00" no
17 Pharmacie Kerangal      "Mo-Fr 09:00-12:30,14:30-19:30; Sa 09:00-12:30,14:30-19:00" yes
18
```

```
[out:csv(name, opening_hours, wheelchair)];
area[admin_level=8][name="Rennes"];
nwr[amenity = pharmacy]["opening_hours"] ["wheelchair"](area);
out;
```



# Exports sous formes de tableaux de données

Exporter une liste des magasins de Rennes avec leurs noms, le type, les horaires, le téléphone, l'accessibilité PMR....

```
1 [out:csv(name, "shop", "opening_hours", "phone", "wheelchair")];
2 area[admin_level=8][name="Rennes"];
3 nwr[shop](area);
4 out;
```

```
1 name shop opening_hours phone wheelchair
2 Super U Rennes Fougères supermarket Mo-Sa 09:00-20:00 +33 2 99 36 04 26 yes
3 Carrefour City Rennes Metz supermarket Mo-Sa 07:00-21:00; Su 09:00-13:00 yes
4 Picard frozen_food
5 Respire bakery yes
6 i-Run clothes
7 TLC Informatique computer "Mo 14:00-18:45; Tu-Sa 09:30-12:30,14:00-18:45; Sa 09:30-
8 Christian Janvier bakery Mo-Fr 07:00-14:00; Sa 07:30-14:00; PH closed no
9 Aven Tour travel_agency +33 2 99 30 22 94
10 Diagonal supermarket +33 2 99 22 07 25 yes
11 Total Access convenience;gas 24/7 +33 2 99 36 15 89
12 Christine Carnet bakery Tu-Sa 06:30-20:00; Su 06:00-19:00 +33 2 99 31 87 23 no
13 Diagonal supermarket
14 U Express supermarket Mo-Sa 08:30-19:30 +33 2 99 63 14 02 yes
15 Le Fournil de pierre bakery
16 Brioche Dorée bakery Mo-Fr 08:00-19:00; Sa 09:00-19:00 +33 2 23 42 41 20 yes
17 King Jouet toys Mo-Sa 10:00-19:00 +33 2 99 51 32 32
18 vacant
19 Christian Janvier bakery Tu-Sa 07:30-19:45; Su 07:30-13:30
```

```
[out:csv(name, shop, opening_hours, phone, wheelchair)];
area[admin_level=8][name="Rennes"];
nwr[shop](area);
out;
```

```
33 Kaboul butcher
34 Boulangerie du Gros Chêne bakery "Tu-Su,PH 07:00-20:30"
35 Thas Phone - Swarnalaya convenience "Mo-Su,PH 09:00-22:00" +33 7 60 67 35 35 no
```

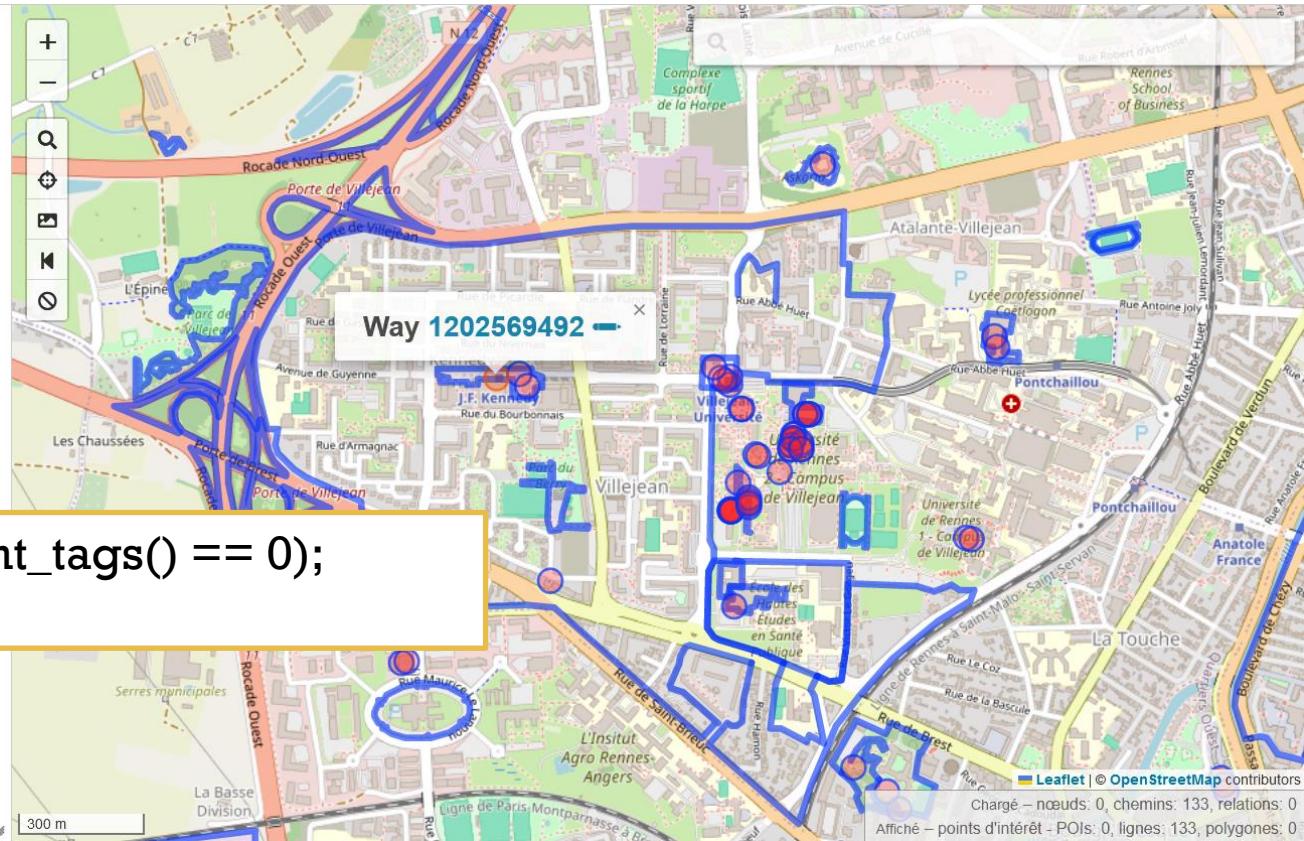


# Contrôle données



## Eléments sans tags

```
1 way({{bbox}})(if:count_tags() == 0);  
2   out geom;  
3 |
```



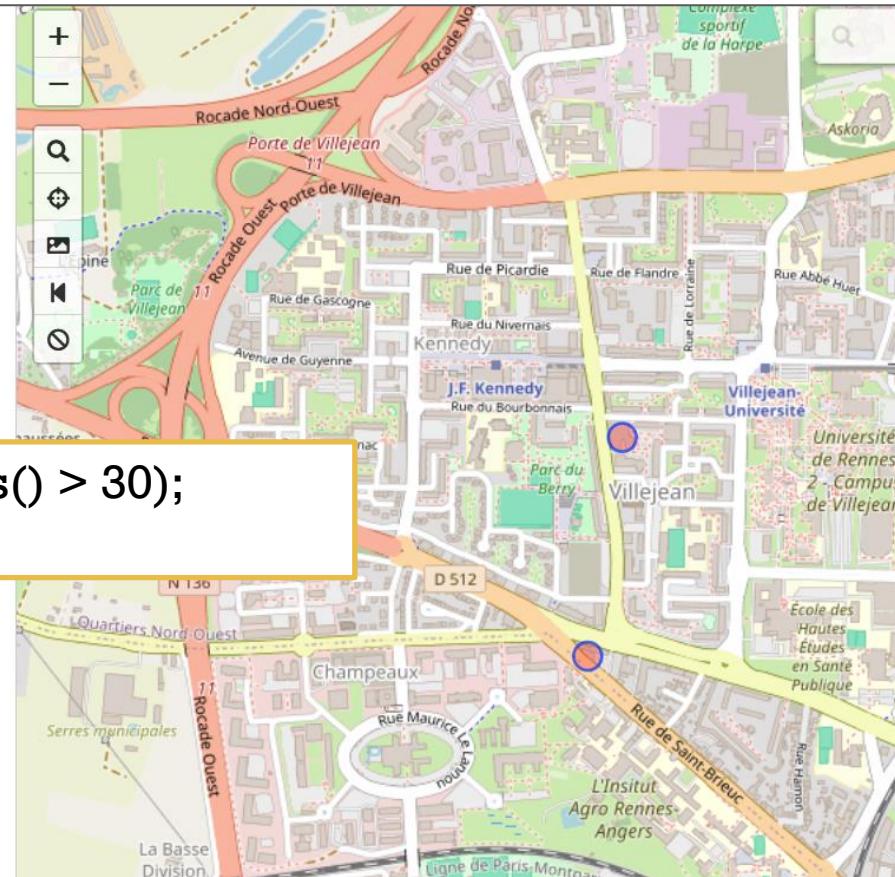


# Contrôle données



Objets avec plus de 30 tags

```
1 nwr({{bbox}})(if:count_tags() > 30);  
2 out geom;  
3
```



```
nwr({{bbox}})(if:count_tags() > 30);  
out geom;
```

<https://overpass-turbo.eu/s/1Nrp>



# Contrôle données



Montrer tous les bâtiments dont la hauteur est renseignée

The screenshot shows the Overpass Turbo interface with a query editor on the left and a map view on the right. The map displays the city of Nantes, France, with many buildings represented by red circles of varying sizes, indicating they have been identified as having height information. The query in the editor is:

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



# Contrôle données



## Superposition bâtiments / routes

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo ⚙

```
1 way[!covered][!tunnel]
2
3     ["highway"~"primary|secondary|tertiary|trunk|service|residential|primary_link|secondary_link|tertiary_link|unclassified"]["access"!~"no|private"]
4     [!area]
5         ({{bbox}});
6         (way(around:0) ["building"~"."]);
           node(w););
       out meta;
```

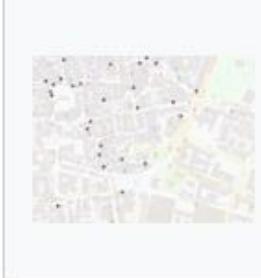
```
way
["highway"~"primary|secondary|tertiary|trunk|service|residential|primary_link|
secondary_link|tertiary_link|unclassified"]["access"!~"no|private"]![!area]
({{bbox}});
(way(around:0)["building"~"."]);
node(w););
out meta;
```



line styles



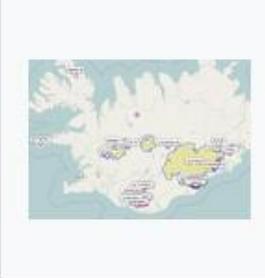
color coding



icons



a simple thematic map



markers with text

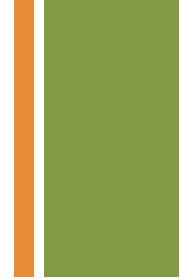
+

## Requêtes et style

[https://wiki.openstreetmap.org/wiki/Overpass\\_turbo/MapCSS](https://wiki.openstreetmap.org/wiki/Overpass_turbo/MapCSS)



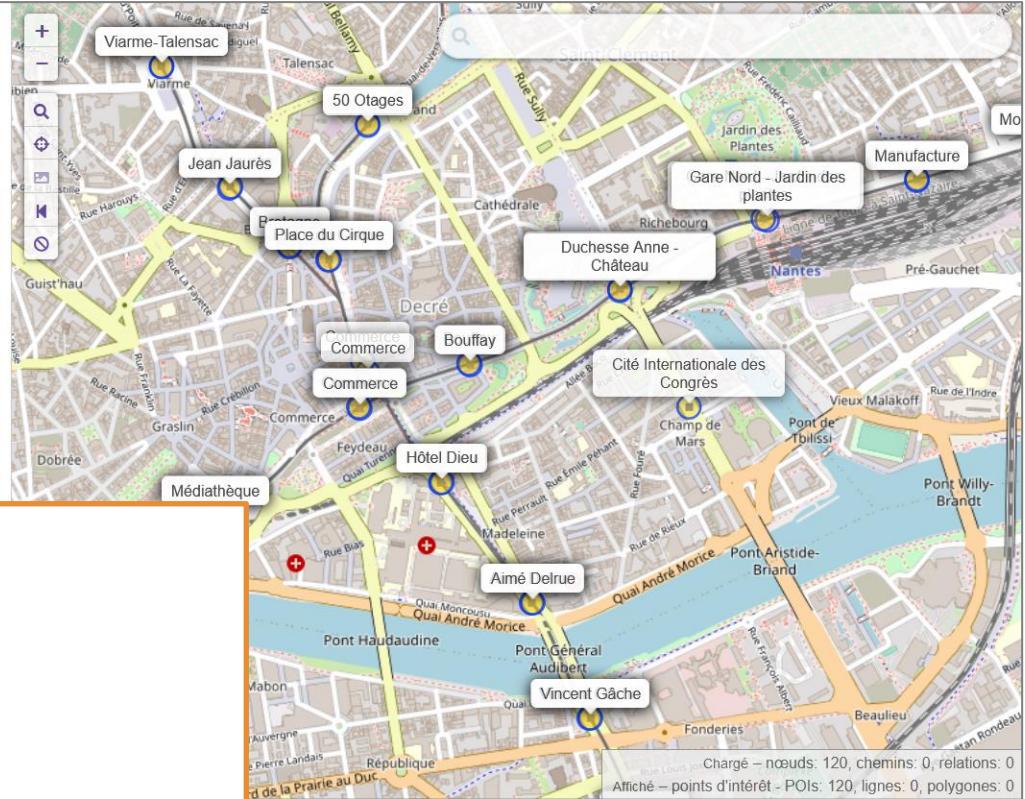
# Requête et style



## Afficher des étiquettes

```
1 [out:json][timeout:25];
2 {{geocodeArea:nantes}}->.searchArea;
3
4 (
5   node["public_transport"]="stop_position" ["tram"]="yes"](area.searchArea);
6 );
7
8 {{style:
9
10 node { text: name; }
11
12 }}
13
14
15 out body;
16 >;
17 out skel qt;
```

```
  {{style:
    node { text: name; }
}}
```



<https://overpass-turbo.eu/s/1j8i>



# Requête et style

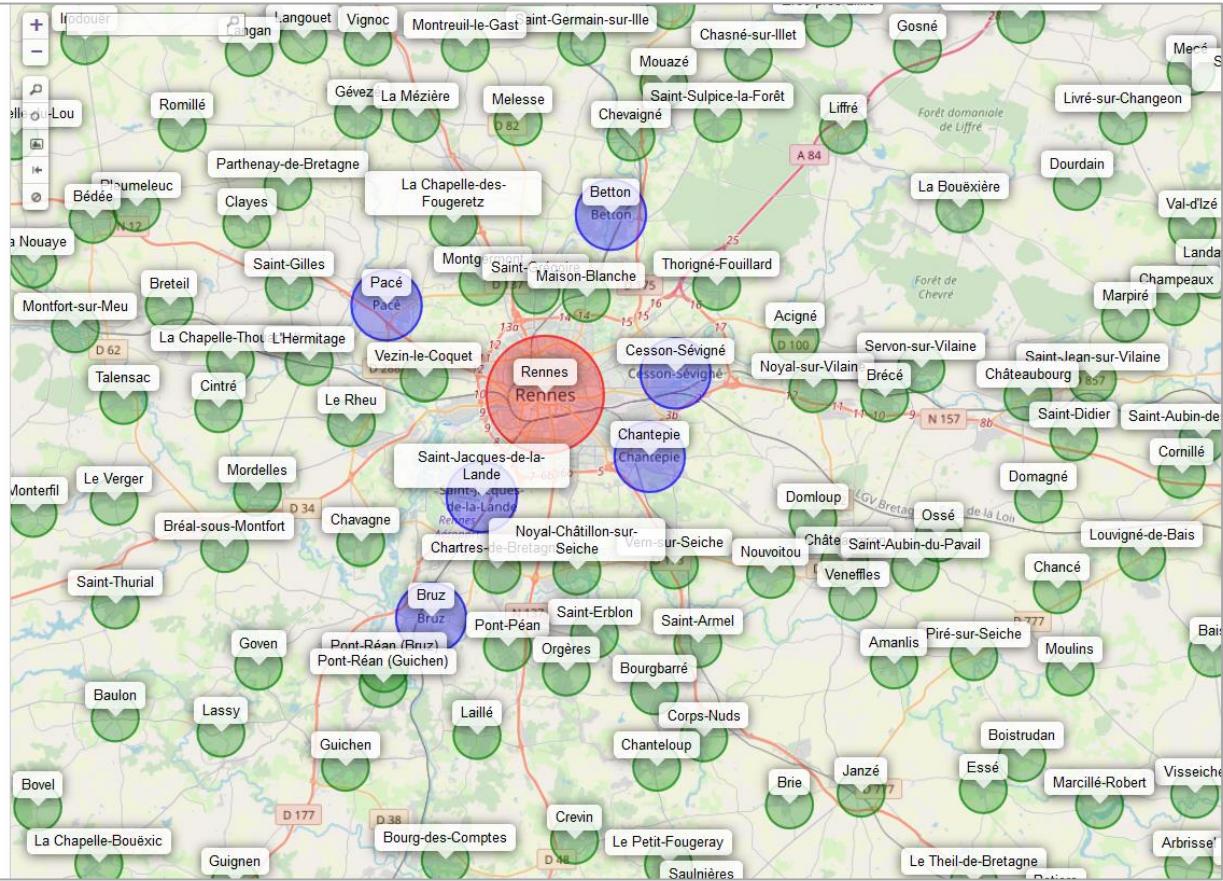


## Afficher des étiquettes en fonction du type de ville

```
[bbox: [{"bbox"}]];
(node[place=city];
node[place=town];
node[place=village];
);

{{style:
node [place=city]{text:name ; color:red; fill-color:red;
symbol-size:50; }
node [place=town]{text:name ; color:blue; fill-
color:blue; symbol-size:30;}
node [place=village]{text:name ; color:green; fill-
color:green; symbol-size:20;
}}}

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



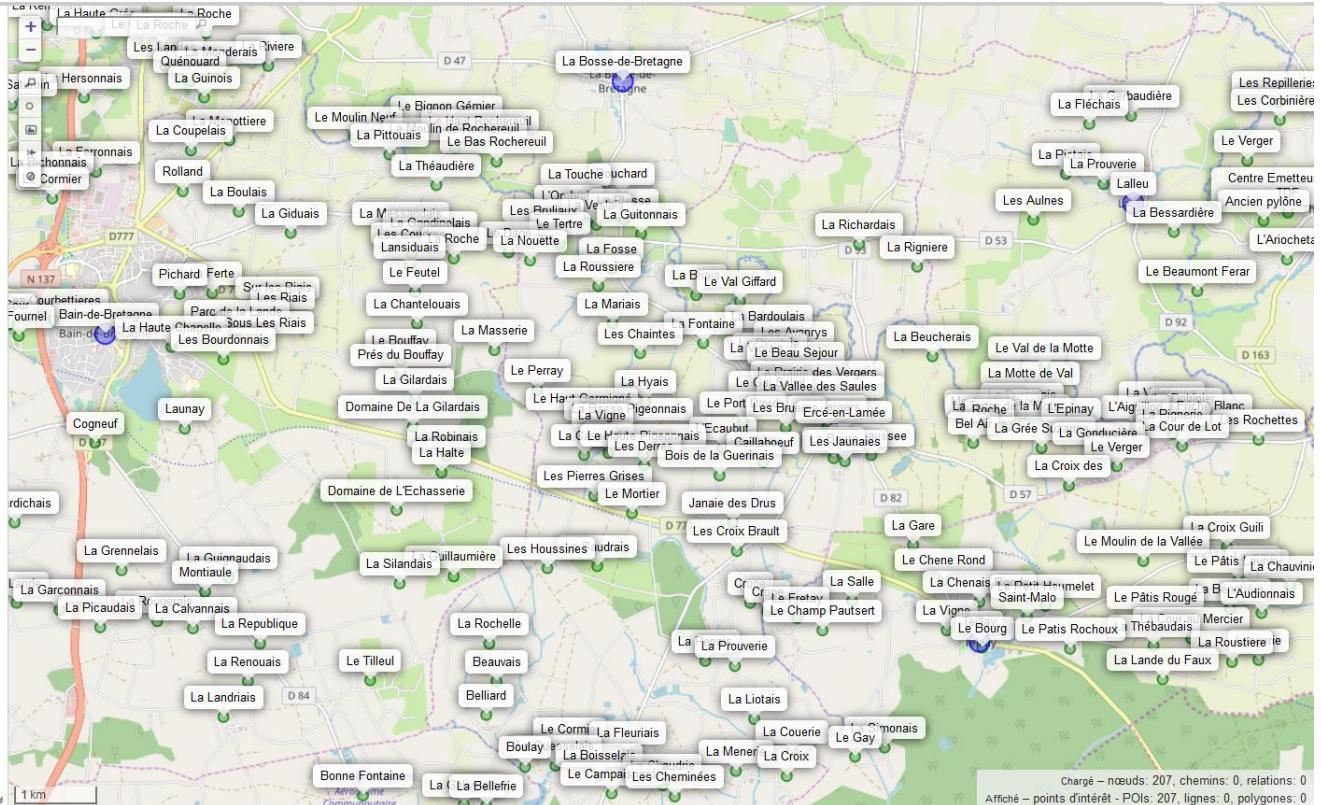


# Requête et style



Afficher des étiquettes en fonction du type de ville et hameau

```
1 [bbox:[{bbox[]}]];
2 (node[place=village];
3 node[place=hamlet];
4 );
5
6 {{style:
7 node [place=hamlet]{text:name ; color:green; fill-
8 color:green; symbol-size:5; font-size:2;}
9 node [place=village]{text:name ; color:blue; fill-
10 color:blue; symbol-size:10;}}
11
12 out body;
13 >;
14 out skel qt;
```

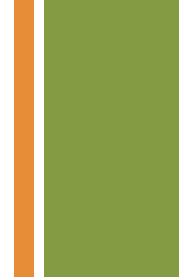


<https://overpass-turbo.eu/#close>

<https://overpass-turbo.eu/s/1j8j>



# 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; }

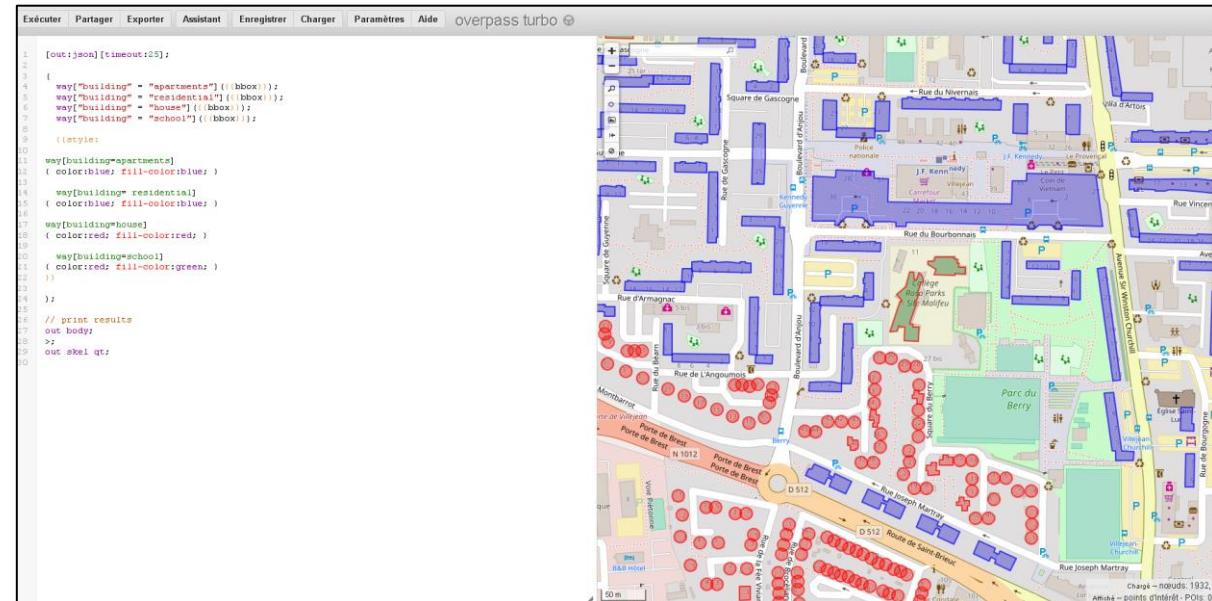
way[building= residential]
{ color:blue; fill-color:blue; }

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

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

);

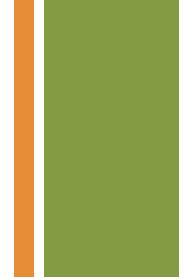
// print results
out body;
>;
out skel qt;
```



<https://overpass-turbo.eu/s/1j8k>

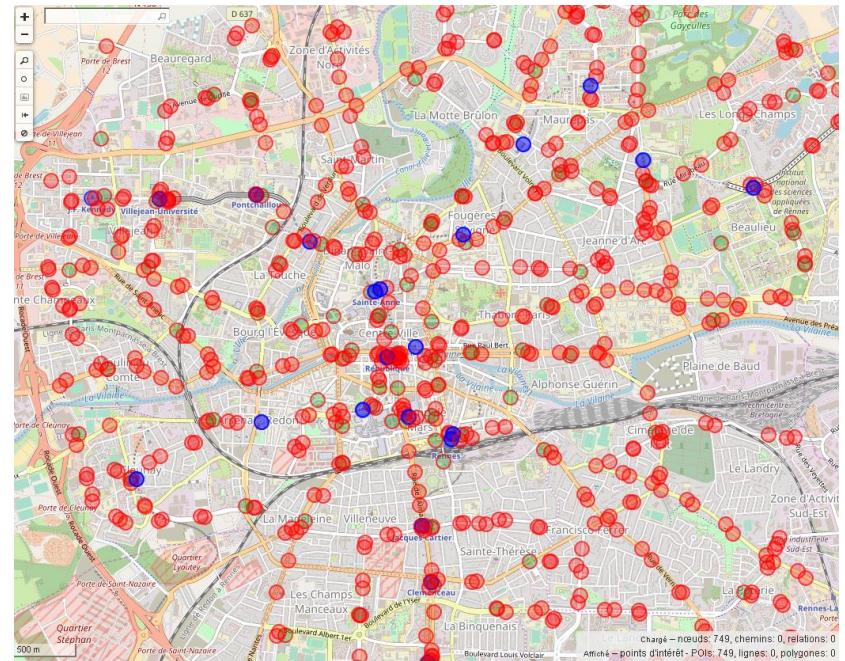


# Requête et style



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

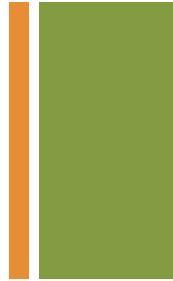
```
[out:json][timeout:25];  
  
{{geocodeArea:rennes}}->.searchArea;  
  
( node["public_transport"]="stop_position" ["subway"]="yes"  
(area.searchArea);  
  node["highway"]="bus_stop"](area.searchArea);  
  node["amenity"]="bicycle_rental"](area.searchArea);  
  
{{style:  
  
  node[public_transport=stop_position]  
  { 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;
```



<https://overpass-turbo.eu/s/1j8l>



# Requête et style



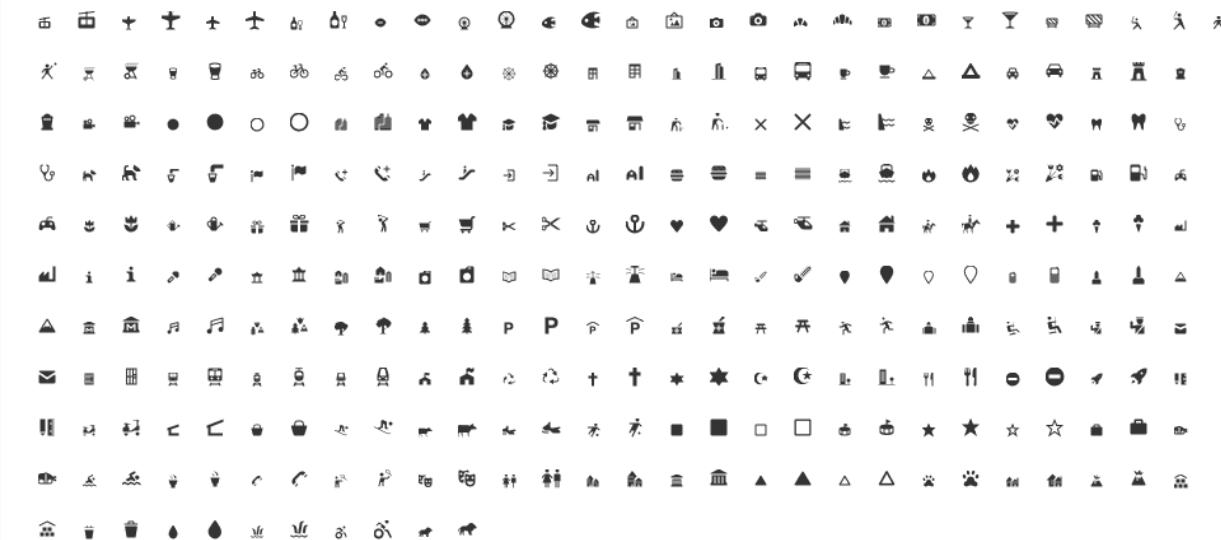
## Utiliser des icônes (maki)

<https://www.mapbox.com/maki-icons/>

### MAKI

Maki is an icon set made for map designers. Maki includes icons for common points of interest like parks, museums, and places of worship. Each icon is available as an SVG in two sizes: 11px by 11px and 15px by 15px. Maki is open source and CC0 licensed.

[Download Maki](#)



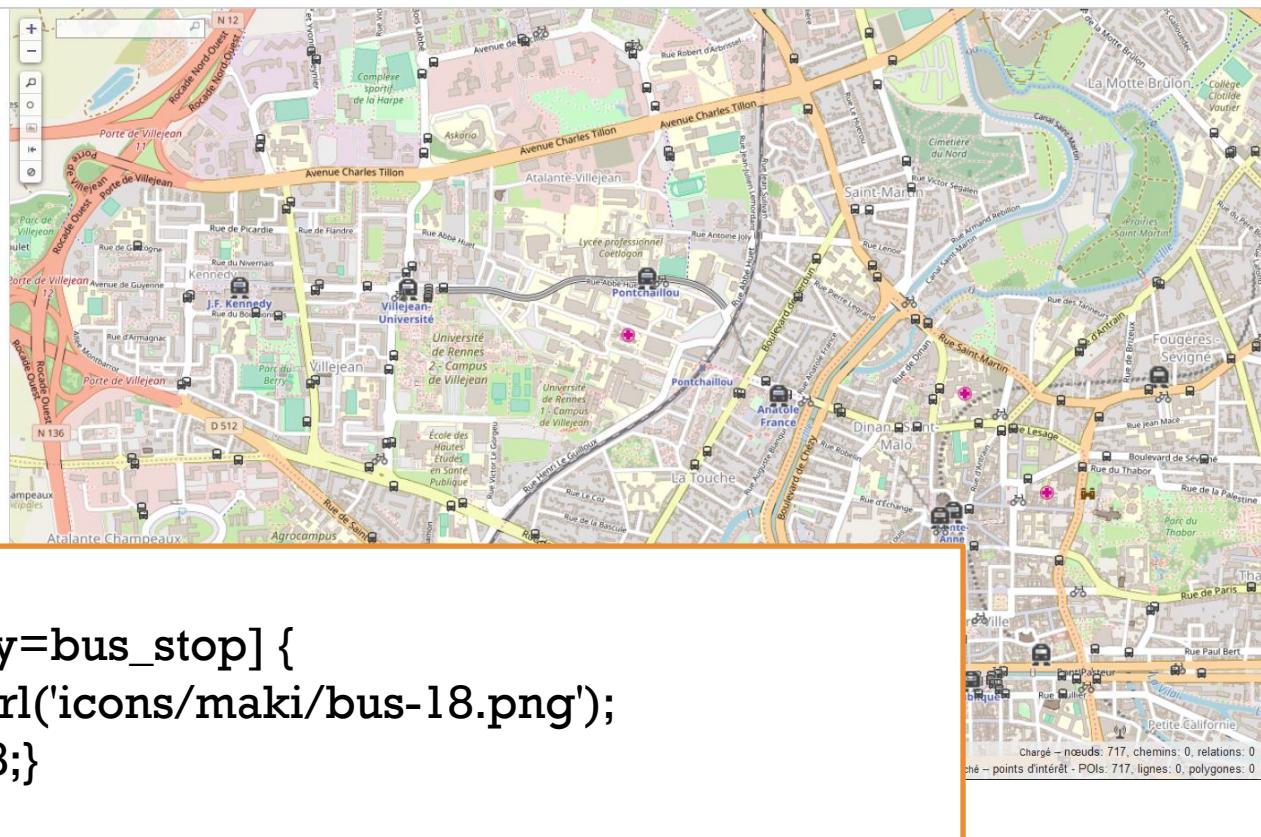


# Requête et style



## Utiliser des icônes (maki)

```
1 [out:json][timeout:25];
2
3 {{geocodeArea:rennes}}>.searchArea;
4
5 ( node["public_transport"]="stop_position"
6 ["subway"="yes"] (area.searchArea);
7 node["highway"]="bus_stop" (area.searchArea);
8 node["amenity"]="bicycle_rental" (area.searchArea);
9
10 {{style:
11   node[highway=bus_stop] {
12     icon-image: url('icons/maki/bus-18.png');
13     icon-width: 15;
14   }
15
16   node[public_transport=stop_position]
17   { icon-image: url('icons/maki/rail-18.png');
18     icon-width: 30;
19   }
20
21   node[amenity=bicycle_rental]
22   { icon-image: url('icons/maki/bicycle-18.png');
23     icon-width: 20;
24   }
25 };
26 out body;
27 >;
28 out skel qt;
29
30
31
32
```



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



# Requête et style



## Mise en forme des sentiers

network	iwn nwn rwn lwn	International walking network: long distance paths that cross several countries National walking network: long distance paths Regional walking network: used for walking routes that cross regions Local walking network: used for small local walking routes
---------	--------------------------	--

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo ⚙

Carte Données

```
1 [bbox:[{bbox}]];
2
3 (relation[route=hiking][network~".wn$"];way(r);>);out;
4
5 {{style:
6
7 relation[network=lwn] way { color:blue; fill-color:cyan; }
8 relation[network=iwn] way { color:red; fill-color:red; }
9 relation[network=nwn] way { color:green; fill-color:green; }
10 relation[network=rwn] way { color:yellow; fill-color:yellow; }
11
12
13 }}
```

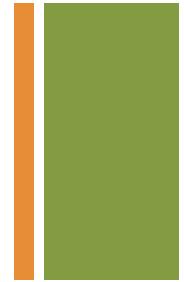
Chargé – noeuds: 75447, chemins: 3024, relations: 65  
Affiché – points d'intérêt - POIs: 490, lignes: 3089, polygones: 0

2 km

<https://overpass-turbo.eu/s/1Nrq>



# Requête et style



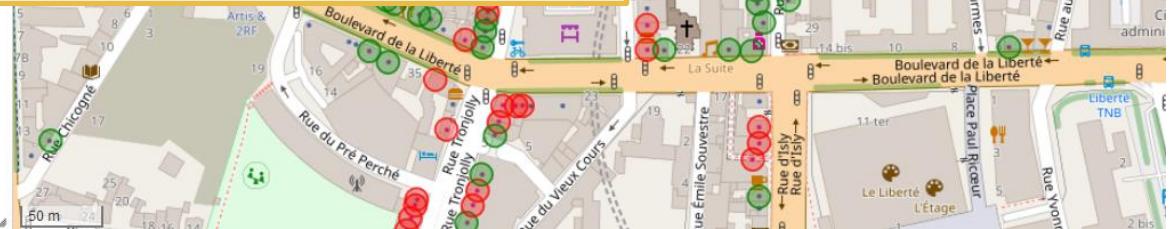
Mise en forme des magasins selon si l'information des horaires est remplie ou non

```
1 node[shop]
2   ({{bbox}});
3   out;
4
5 {{style:
6   node
7   { color:red; fill-color: red;}
8
9   node[opening_hours]
10  { color:green; fill-color:green; }
11 }}
```



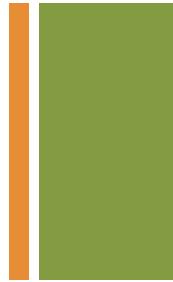
```
node[shop]({{bbox}});
out;

{{style:
node {color:red; fill-color: red;}
node[opening_hours] { color:green; fill-color:green; }
}}
```



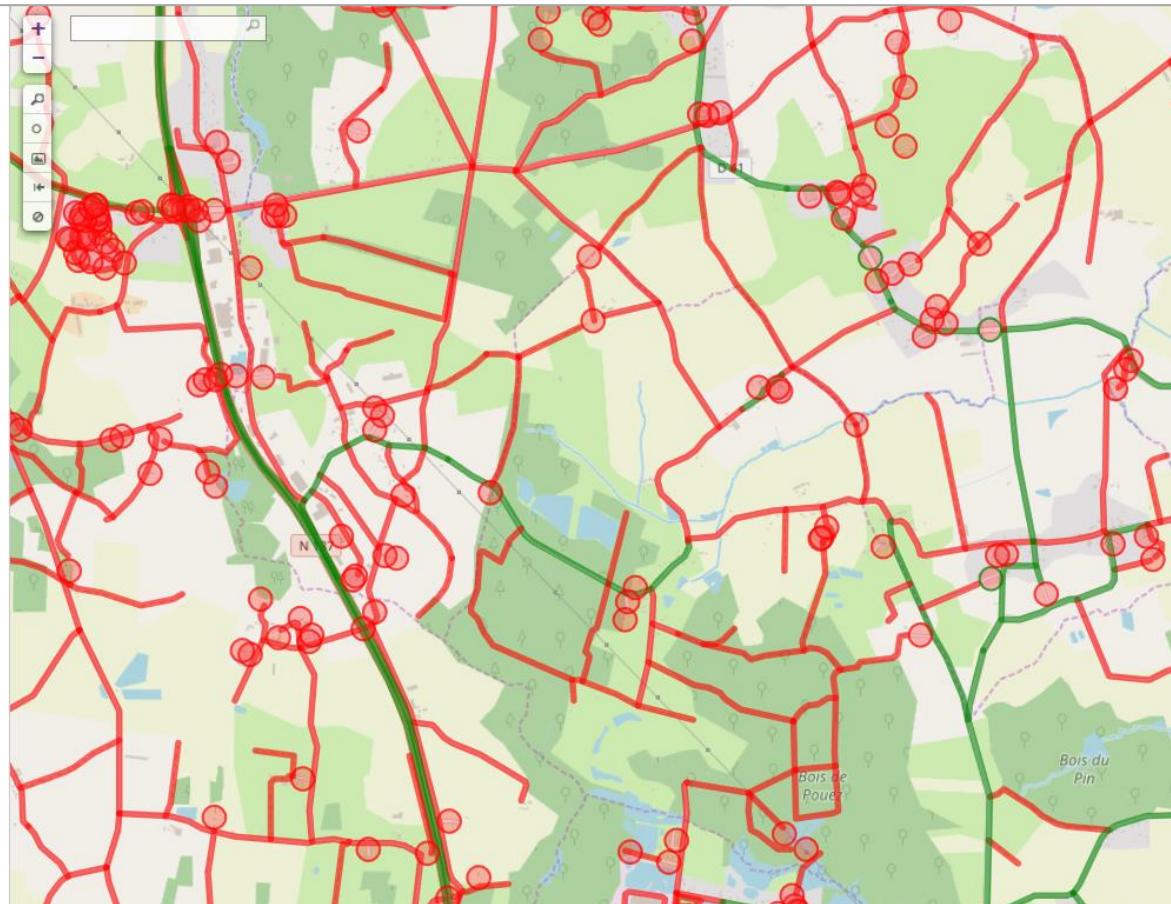


# Requête et style



Mise en forme des routes selon si l'information de vitesse est remplie ou non

```
way ["highway"]({{bbox}});
{{style:
way{color: red;}
way[maxspeed]{ color:green; }
}}
// print results
out body;
>;
out skel qt;
```



**+ Requêtes  
contributors / date**



# Extraction par contributeur

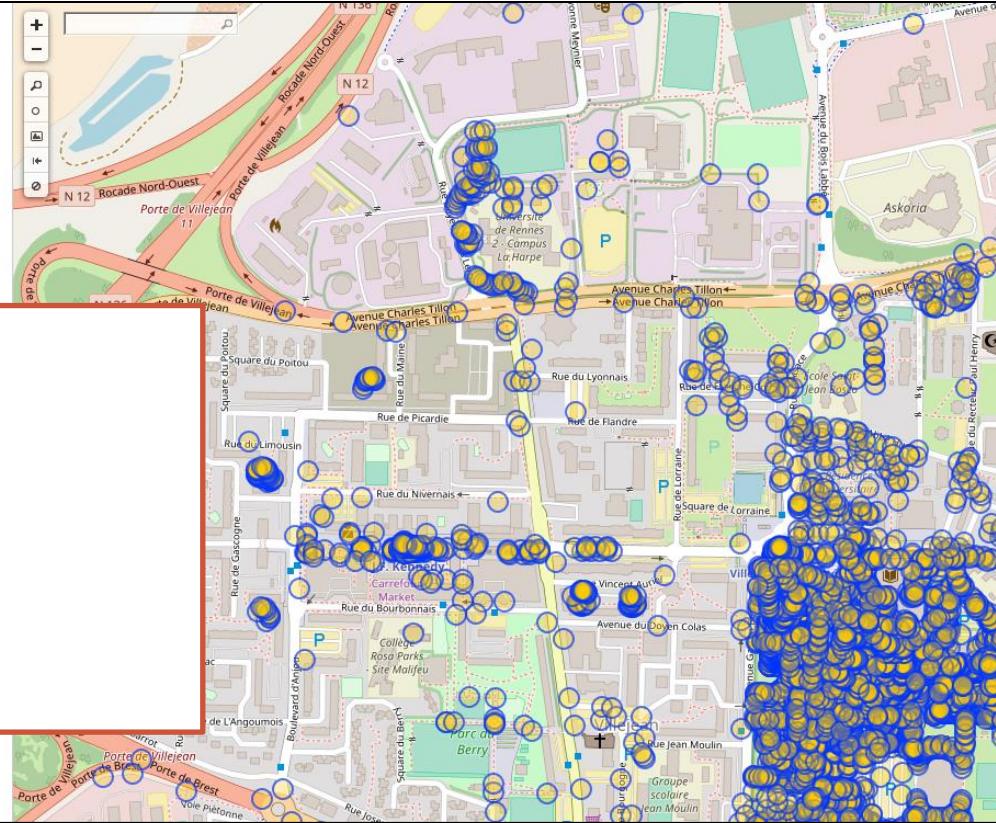


```
[out:json][timeout:25];
(node(user:PanierAvide)({{bbox}}));;
```

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

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

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



<https://overpass-turbo.eu/s/1Ns5>

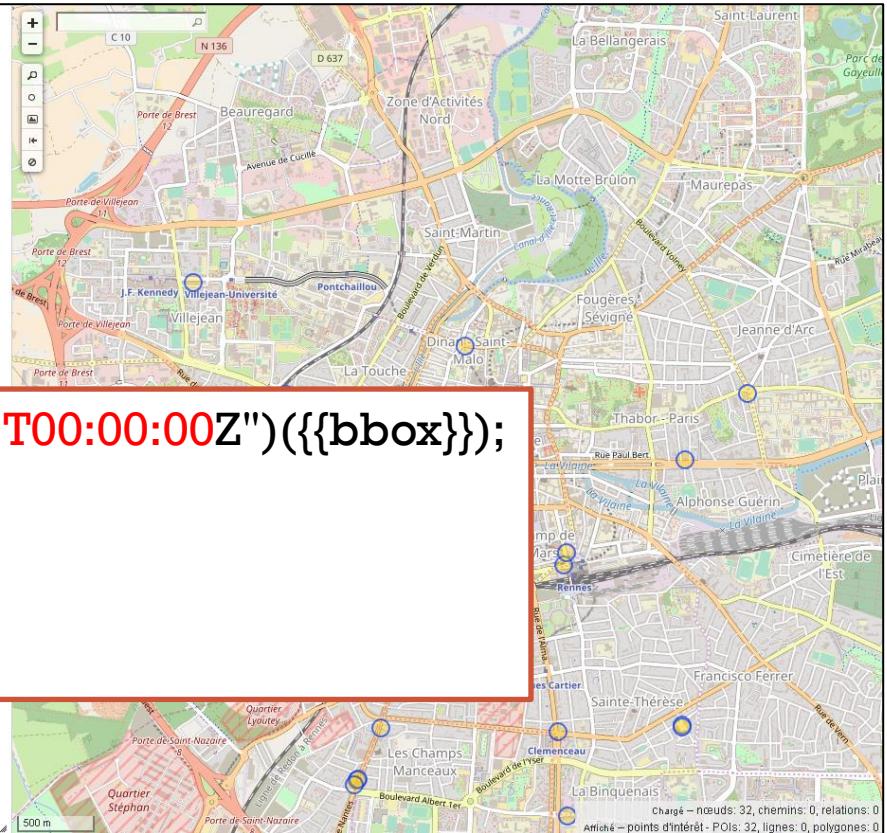


# Extraction par date



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

```
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 |
```



```
( node["amenity"](newer:"2024-06-01T00:00:00Z")({{bbox}});
);
out body;
>;
out skel qt;
```

<https://overpass-turbo.eu/s/1Ns6>

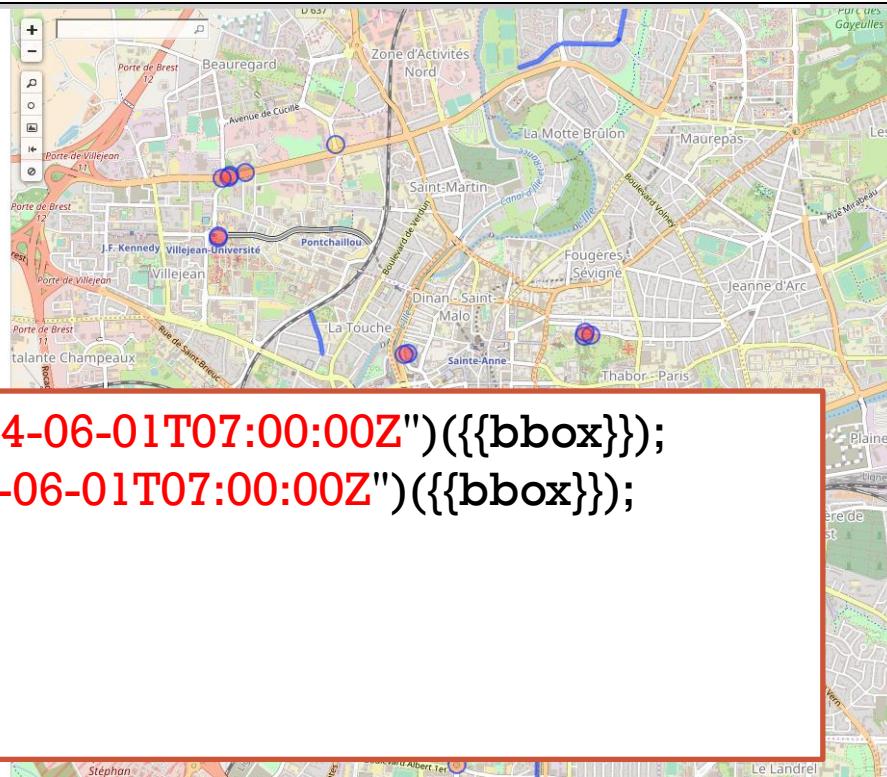


# Extraction par date



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

```
[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;
```

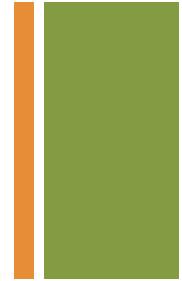


+

# Requêtes spatiales

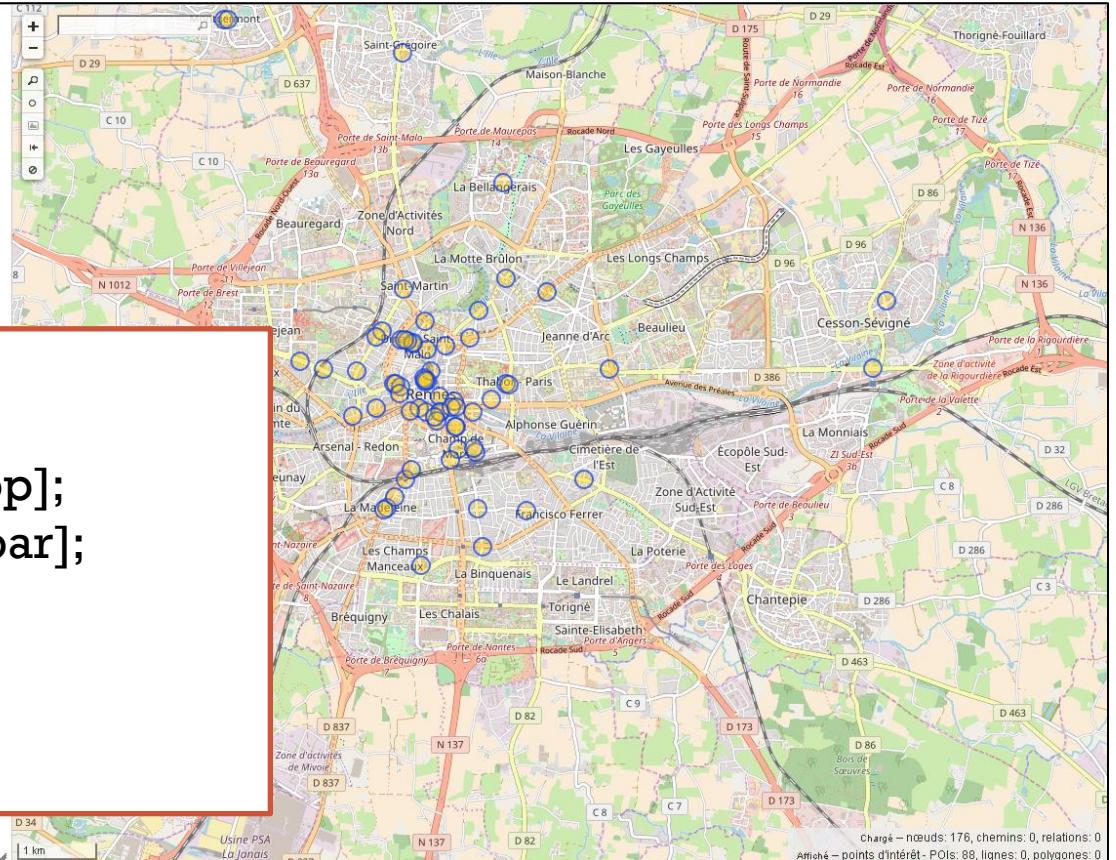


# Sélection spatiale



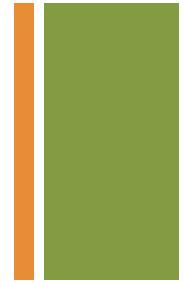
Sélectionner tous les bars à moins de 100m d'un arrêt de bus

```
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 |
```





# 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;
```

The map displays the city of Rennes with various districts labeled: Villejean, J.F. Kennedy, Pontchaillou, La Touche, Saint-Martin, Fougères-Sévigné, Jeanne d'Arc, Thabor-Paris, and others. Numerous blue circles of varying sizes are scattered across the map, representing bars located within 200 meters of a subway station. The map also shows the network of roads, rivers (Vilaine, Rance), and the Rennes Métro system.



# Sélection spatiale



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

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo ⚙

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

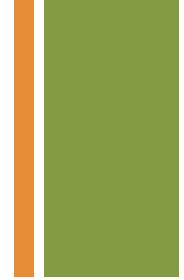
The map displays the city of Rennes with various transportation infrastructure. Numerous blue circles, representing bus stops, are scattered across the city, with many located near or around metro stations. Labels on the map include Beurregard, Pontchaillou, Villejean, Villejean-Université, Pontchaillou, La Motte Brûlon, Maurepas, Les Longs Champs, Institut national des sciences appliquées de Rennes, Beaulieu, Fougères, Cézanne, Jeanne d'Arc, La Touche, Malo, Sainte-Anne, and Porte de l'Ille. The map also shows several roads, including the N 136, D 82, and Boulevard du Portugal, along with green spaces and buildings.

[out:json][timeout:25];  
area[name="Rennes"];  
node(area)["public\_transport"="stop\_position"] ["subway"="yes"] ;  
node(around:200)["highway"="bus\_stop"];  
out;  
out body;  
>;  
out skel qt;

Chargé – nœuds: 280, chemins: 0, relations: 0  
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 ⚙️

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

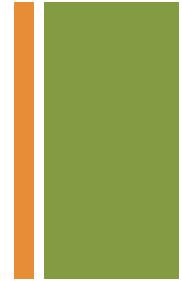
Carte Données

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

Chargé – noeuds: 0, chemins: 6, relations: 0, points d'intérêt: 0, nœuds: 0, lignes: 0, polygones: 0



# Sélection spatiale



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

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

Carte Données

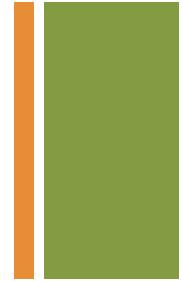
```
1 [out:json][timeout:25];
2 area[name="Rennes"];
3 node(area)[public_transport=stop_position][subway=yes];
4 node(around:200)[amenity];
5 out;
6 out body;
7 >;
8 out skel qt;
```

The map displays the city of Rennes with numerous blue circles of varying sizes scattered across the urban area. These circles represent points of interest (POIs) located within a 200m radius of subway stations. The map includes labels for various neighborhoods, roads, and landmarks. A legend in the top right corner indicates that the blue circles represent POIs.

Chargé – noeuds: 3258, chemins: 0, relations: 0  
Affiché – points d'intérêt - POIs: 1629, lignes: 0, polygones: 0



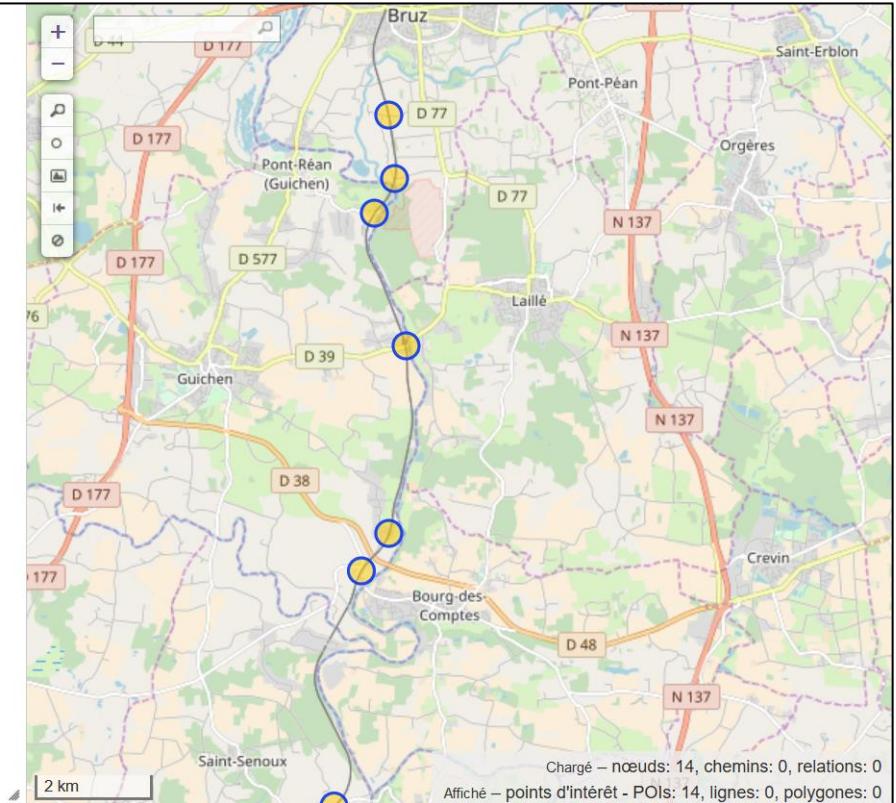
# Sélection spatiale



Sélectionner les intersections entre routes et voies ferrées

```
1 [bbox:{bbox}];  
2 way["railway"="rail"]->.major;  
3 way["highway"]->.minor;  
4 node(w.major)(w.minor);  
5 out body;  
6 >;  
7 out skel qt;
```

```
[bbox:{bbox}];  
way["railway"="rail"]->.major;  
way["highway"]->.minor;  
node(w.major)(w.minor);  
out body;  
>;  
out skel qt;
```



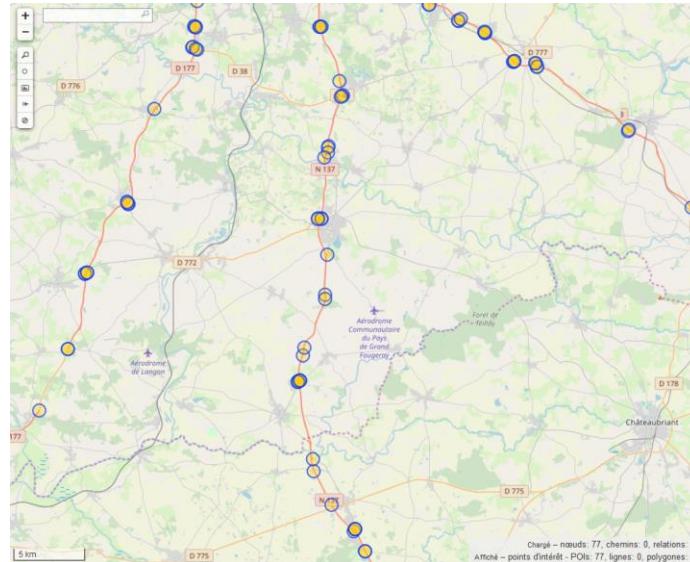


# Sélection spatiale



Sélectionner les sorties des voies rapides

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





# Sélections spatiales



Rechercher toutes communes qui n'ont pas d'école

```
// Output in CSV format
[out:csv(::id, name)];

area[admin_level=6]["name"="Ille-et-Vilaine"][boundary=administrative]-
>.boundaryarea;

// Echelle d'analyse la commune
rel(area.boundaryarea)[admin_level=8];
// map relations to their respective area
map_to_area -> .all_level_8_areas;

// chercher toutes les écoles
( node(area.boundaryarea)["amenity"="school"];
  way(area.boundaryarea)["amenity"="school"];>);

// qui sont situées dans des communes
is_in;
area._[admin_level=8] -> .level_8_areas_with_school;

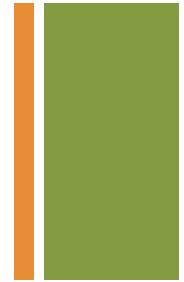
// Toutes les communes qui ne possèdent pas d'école
(.all_level_8_areas; - .level_8_areas_with_school;);

// Calculate and output relation
rel(pivot);
out geom;
```

id	name
1	@id name
2	89002 Dourdain
3	106684 Arbrissel
4	106686 Availles-sur-Seiche
5	106687 Amanlis
6	107908 Balazé
7	107909 Bais
8	107911 Boistrudan
9	107912 Brielles
10	107915 Bruc-sur-Aff
11	107918 Brie
12	108429 Châtillon-en-Vendelais
13	108430 Chelun
14	108431 Chancé
15	108434 Champeaux
16	108437 Cornillé
17	108440 Coësmes
18	111020 Drouges
19	111021 Domalain
20	111055 Erbrée
21	111056 Eancé
22	111057 Essé
23	116110 Gahard
24	123371 La Bazouge-du-Désert
25	123374 Lourmais
26	123375 Longaulnay
27	123380 Le Tiercent
28	123382 La Baussiane
29	123383 Le Châtellier
30	123385 La Bousac
31	123386 Le Ferré
32	123387 Luitré
33	128198 La Bosse-de-Bretagne
34	134221 Lanrigan
35	134223 Lalleu
36	134316 Lieuron
37	139656 Marcillé-Robert
38	139657 Montreuil-sous-Pérouse
39	139658 Moulin
40	139663 Moutiers
41	139664 Montreuil-des-Landes
42	139666 Moncéé



# Deux outils complémentaires



QuickOSM

**Successful query, but no result.**

Help with key/value    Reset

**Key**: name  
**Value**: Umweltzone Heidelberg

In    Germany    1000 m  
 Extent of the map canvas  
 Extent of a layer

**Advanced**

Node <input checked="" type="checkbox"/>	Points <input checked="" type="checkbox"/>
Way <input checked="" type="checkbox"/>	Lines <input checked="" type="checkbox"/>
Relation <input checked="" type="checkbox"/>	Multilinestrings <input checked="" type="checkbox"/>
	Multipolygons <input checked="" type="checkbox"/>

Timeout: 25  
Directory: Save to temporary file    Browse  
File prefix:

Show query    Run query

No result

100%

