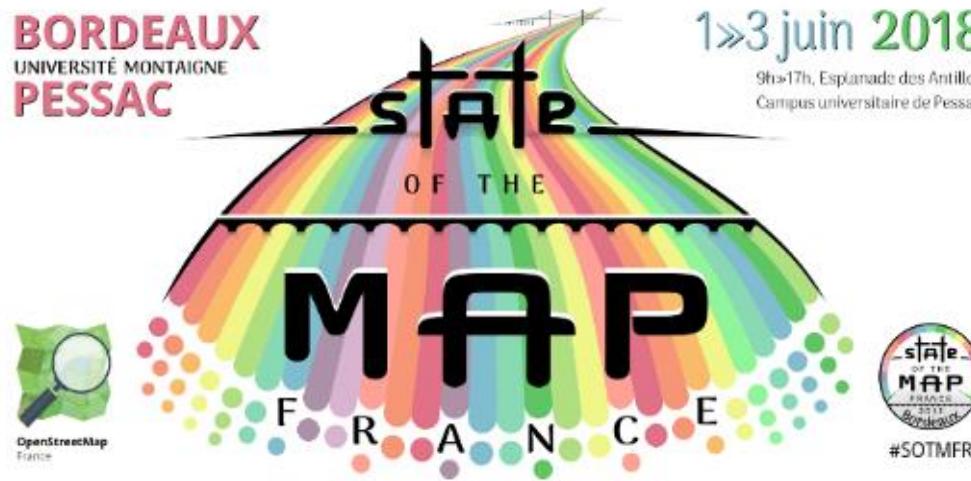


BORDEAUX
UNIVERSITÉ MONTAIGNE
PESSAC

1 > 3 juin 2018

9h > 17h, Esplanade des Antilles
Campus universitaire de Pessac



Atelier : Prise en main de l'API Overpass



@Boris Mericskay



overpass turbo



+

Overpass turbo



API Overpass



API Overpass permet d'interroger la base de données OSM

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

Overpass turbo est une application Web d'exploration de données pour OpenStreetMap

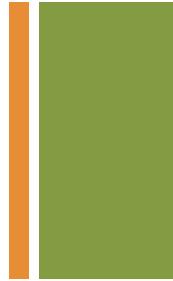
- Ce site permet d'exécuter toutes sortes de requêtes de l' API Overpass et présente le résultat sur une carte interactive.

The screenshot shows the overpass-turbo interface. On the left, a code editor displays a Overpass query: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["landuse"="forest"]((bbox));
11);
12 // print results
13 out skel qt;On the right, a map of a forested area in France is displayed, with several airports labeled: "Aérodrome de Ploufrancé Lorient", "Aérodrome des Coteaux Molinaires de Saint-Cyr Corseguin", and "Aérodrome de L'Île-d'Yeu".

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



Modèle de données OSM

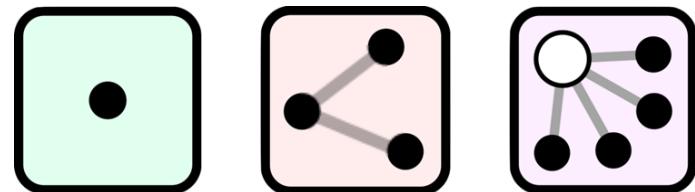


Dans un SIG, les données cartographiques sont représentées de trois façons différentes

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

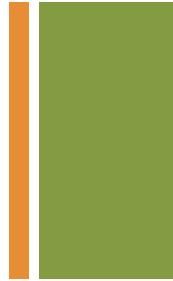
Dans OpenStreetMap, ces trois concepts sont modélisés différemment -> 3 objets primitifs (**Eléments**) :

- **Noeuds** (*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)



Modèle de données OSM

Clip slide

OSM Data Model

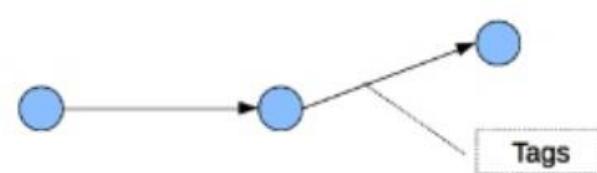
Node



Way



Open polyline



Closed polyline



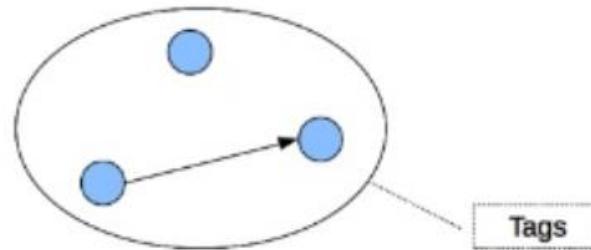
Area



Relation



Tag





La folksonomie OSM



Une sémantique bien particulière

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

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

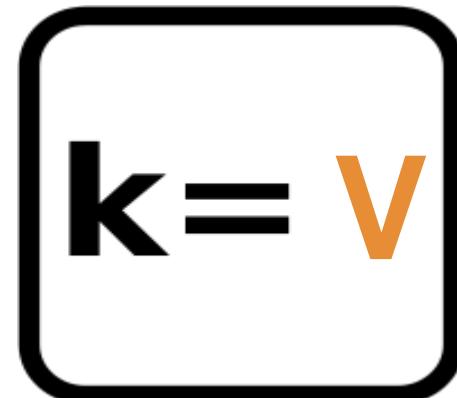


Le schéma des tags repose sur le fonctionnement
`<clé>=<valeur>` (`key=value`)

→ Possibilité d'associer plusieurs tags à un objet

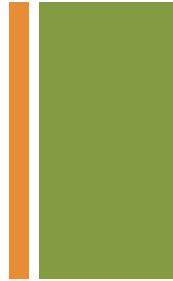
Par exemple pour indiquer qu'un trait correspond à une route secondaire, en sens unique, de vitesse maximale 90 km/h et munie d'une bande cyclable, on utilisera :

- highway=secondary
- oneway=yes
- maxspeed=90
- cycleway=lane





API Overpass



Routes

- Documentation

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

Key = **highway**

Value =

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

Key	Value	Element	Comment	Rendering	Photo
Roads					
These are the principal tags for the road network. They range from the most to least important.					
highway	motorway	<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.					

+

Requêtes classiques



Requêtes classiques



Extraire les routes selon une valeur (hierarchie)

- Les routes principales (trunk) <https://overpass-turbo.eu/s/z8s>

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide Enjoy OpenStreetMap data overpass turbo

Type Key Value

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

way highway trunk

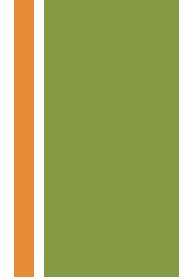
```
8  // query part for: route
9  way["highway"="trunk"]({{bbox}});
10 );
11
12 // print results
13 out body;
14 >;
15 out skel qt;
```

gagner résultats

Emprise



Requêtes classiques



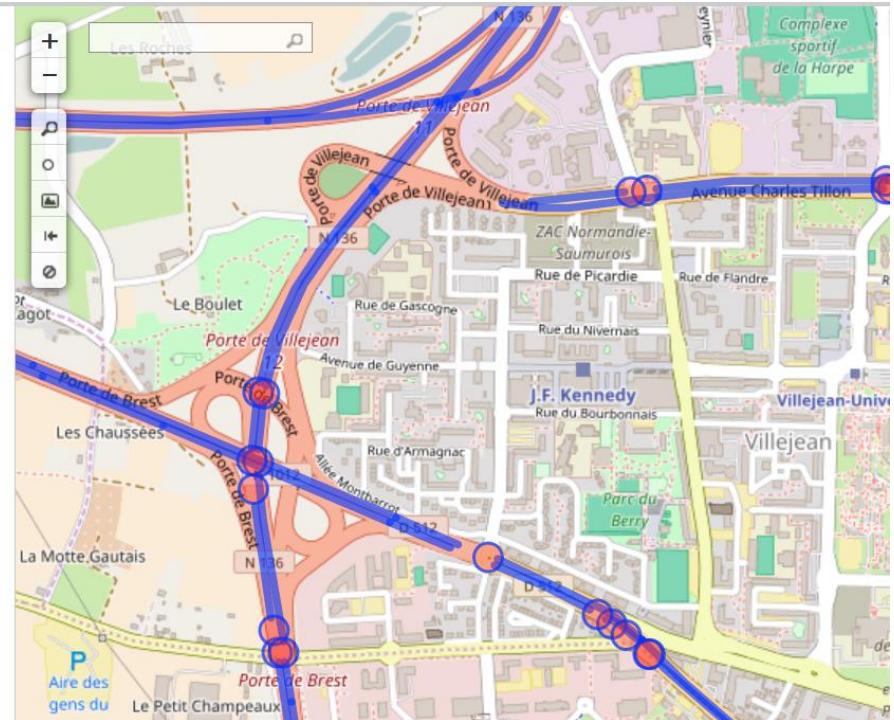
Combiner 2 critères

- 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   // query part for: "route"
9   way["highway"="trunk"]({{bbox}});
10  way["highway"="primary"]({{bbox}});

11  );
12  // print results
13  out body;
14  >;
15  out skel qt;
```





Requêtes classiques



Ajouter un critère de vitesse

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

Enjoy OpenStreetMap data? Support OSM's 2016 donation drive!

overpass turbo

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide

```
/*
This has been generated by the overpass-turbo wizard.
The original search was:
"route"
*/
[out:json][timeout:25];
// gather results
{
    // query part for: "route"
    way["highway"]["maxspeed"="30"]({{bbox}});
};

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

300 m

Chargé – noeuds: 400, ch...

Affiché – points d'intérêt – POIs: 0, li...



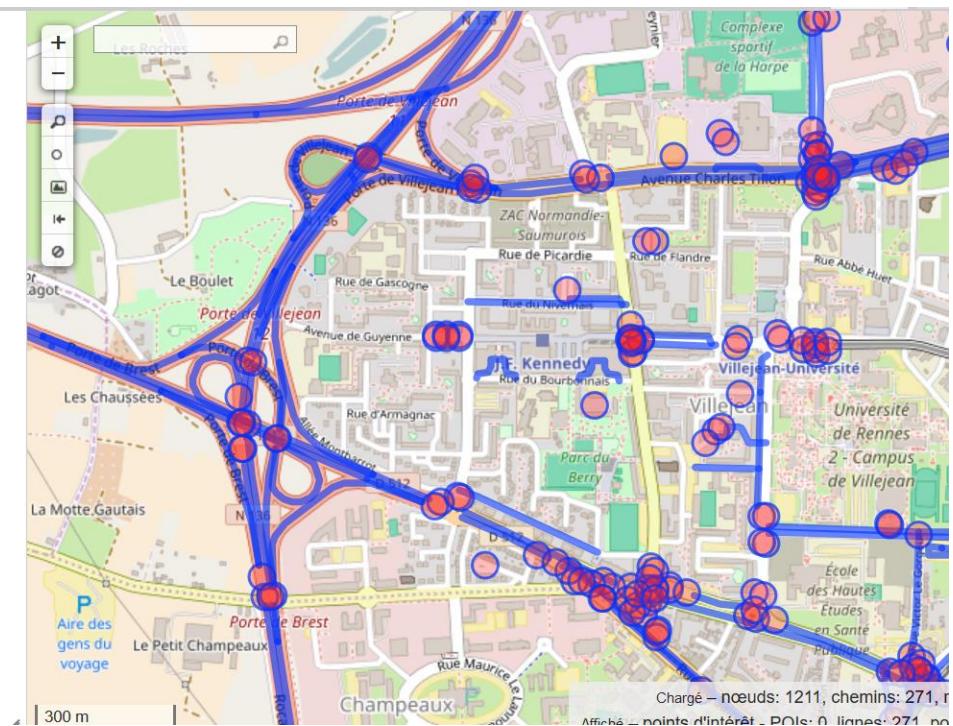
Requêtes classiques



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);
```



Requêtes classiques

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

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

```
/*
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;



|



Requêtes classiques



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

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

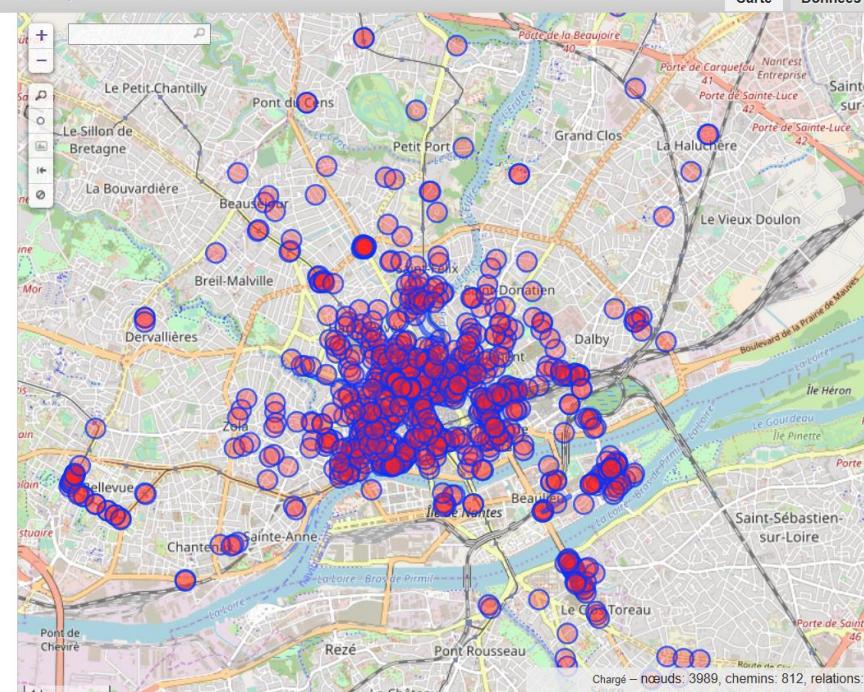
Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aideoverpass turboCarte Données

```
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;
```

/*
This has been generated by the overpass-turbo wizard.
The original search was:
"route in rennes"
*/
[out:json][timeout:25];
// fetch area "rennes" to search in
({geocodeArea:nantes})->.searchArea;
// gather results
(
// query part for: "route"
way ["highway"] ["maxspeed"="30"] ["oneway"="yes"] (area.searchArea);
);
// print results
out body;
>;
out skel qt;

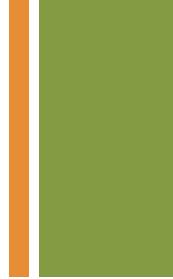
Carte Données

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





Requêtes classiques



Amenity

■ Documentation

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

Key = **highway**

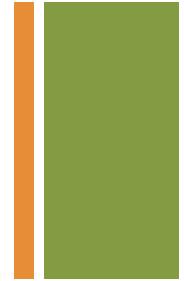
Value =

- 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.		

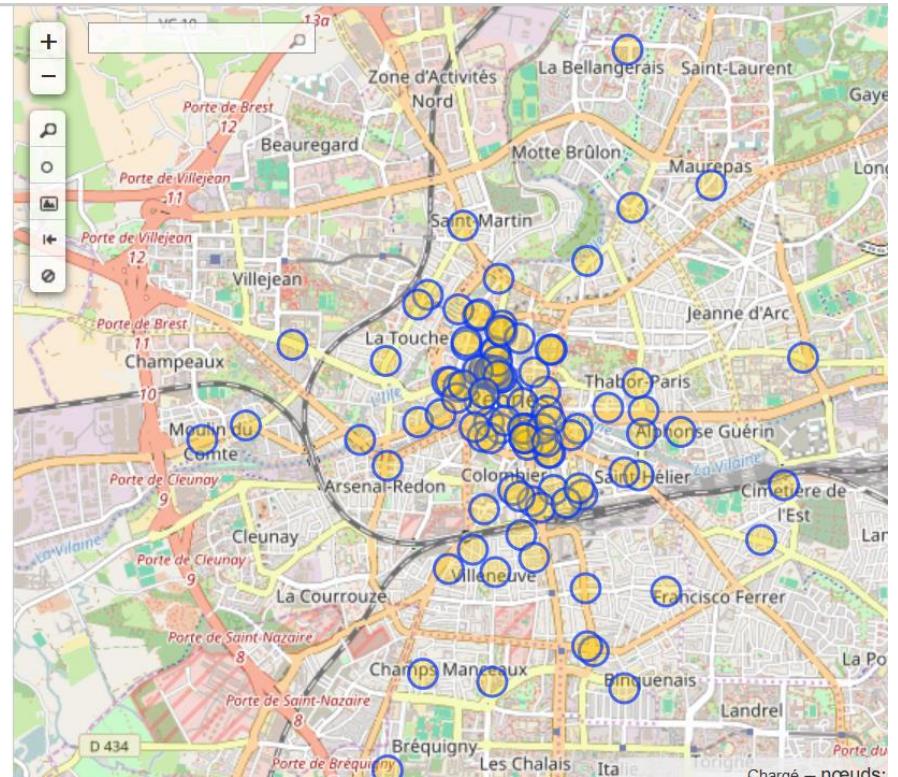


Requêtes classiques



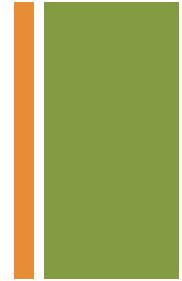
→ Extraire les bars

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





Requêtes classiques

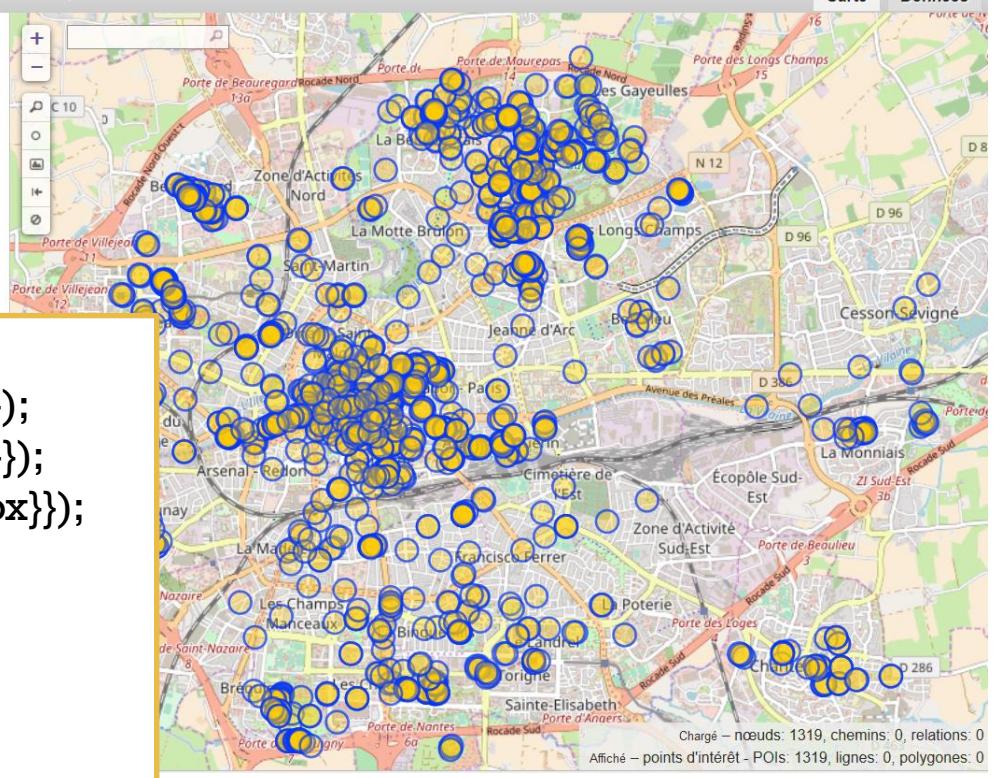


Extraire plusieurs type d'objets

```
Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo ⚙  
Carte Données  
  
1  (   
2    node["amenity"]=="bar"]({{bbox}});   
3    node["amenity"]=="cafe"]({{bbox}});   
4    node["amenity"]=="bench"]({{bbox}});   
5    );   
6    
7  out body;   
8  >;   
9  out skel qt;  
  
10  
  


(   
node["amenity"]=="bar"]({{bbox}});   
node["amenity"]=="cafe"]({{bbox}});   
node["amenity"]=="bench"]({{bbox}});   
);   
  
out body;   
>;   
out skel qt;

  
https://overpass-turbo.eu/#cl
```

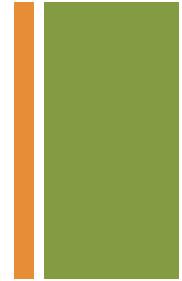


Chargé – noeuds: 1319, chemins: 0, relations: 0
Affiché – points d'intérêt – POIs: 1319, lignes: 0, polygones: 0

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



Requêtes classiques



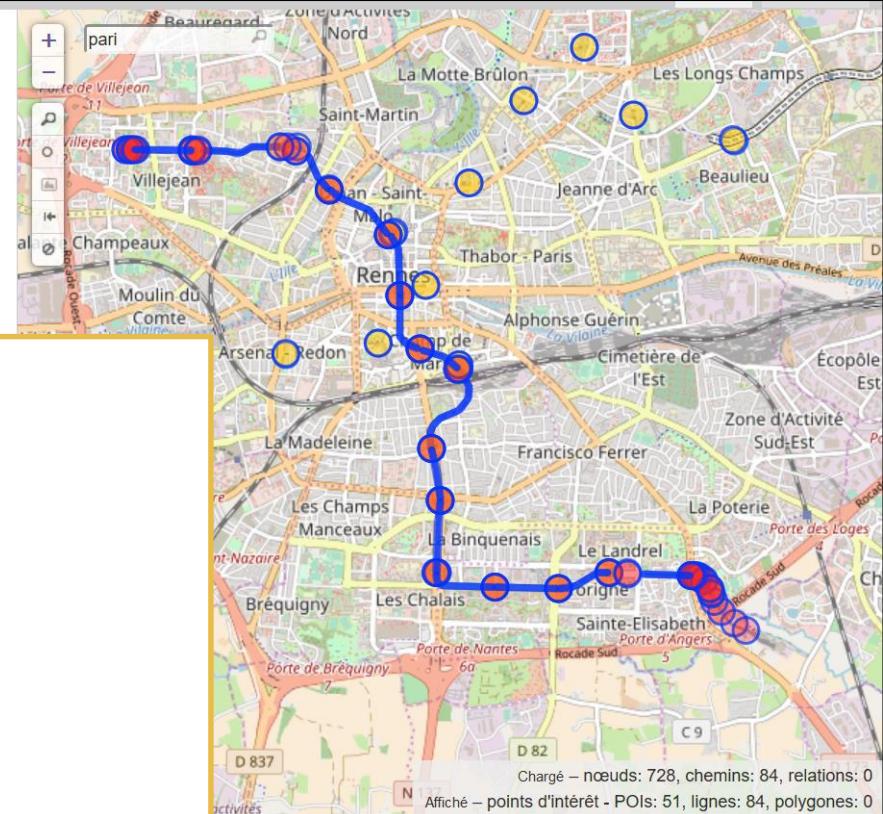
Extraire plusieurs type d'objets

```
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);
);

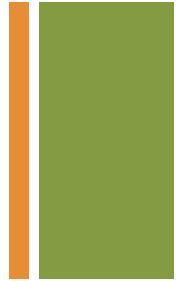
out body;
>;
out skel qt;
```



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



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=coast`)

`boundary=political`

is approved, should be documented in each country where they are used. Can be used for countries, states, regions, etc.

`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 protected areas.

`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

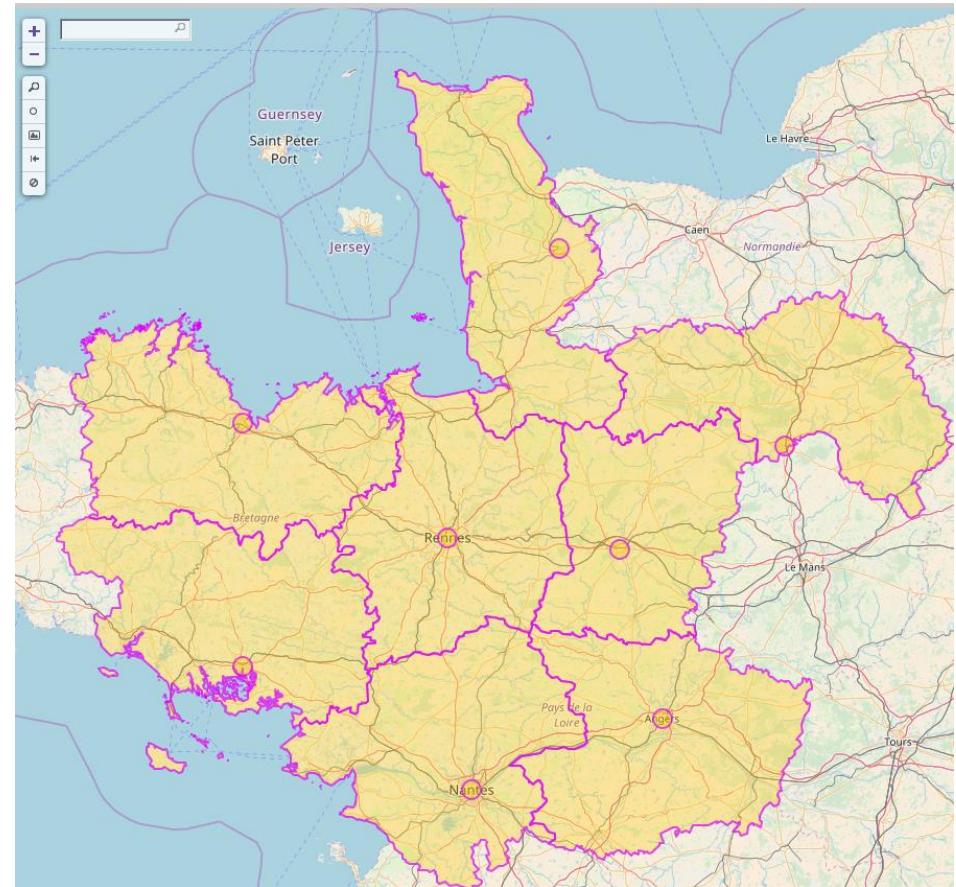


Requêtes classiques



Limites administratives

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





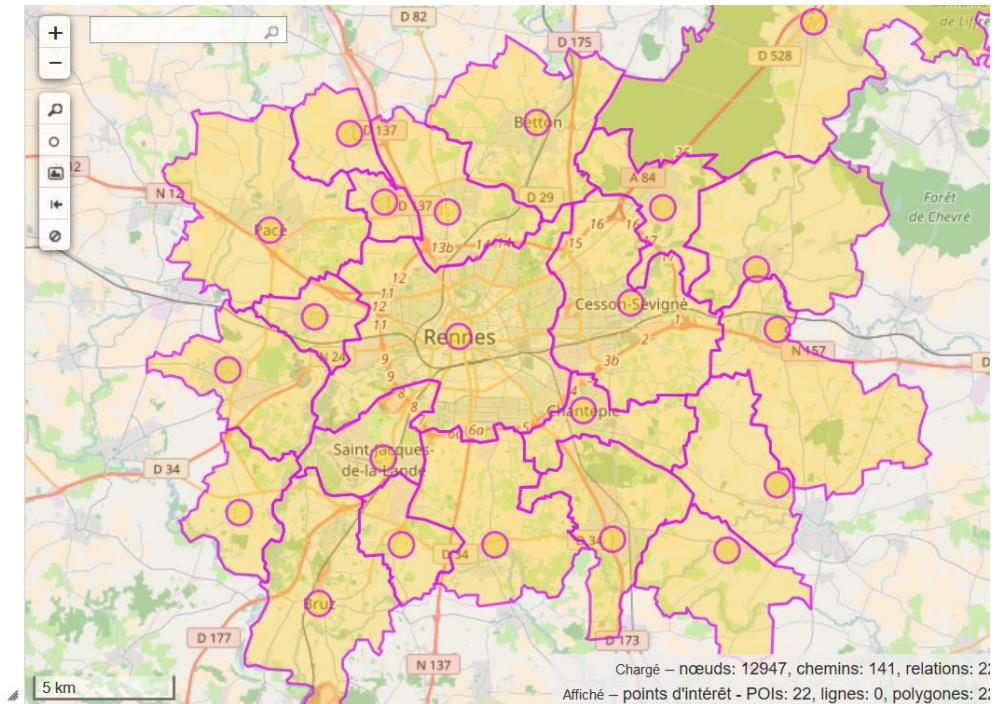
Requêtes classiques



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



Landuse

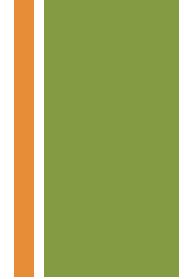
■ Documentation

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

landuse	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.		
landuse	brownfield		Zone où des anciens bâtiments ont été rasés. La construction de nouveaux bâtiments est planifiée, mais pas encore en cours.		
landuse	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)		

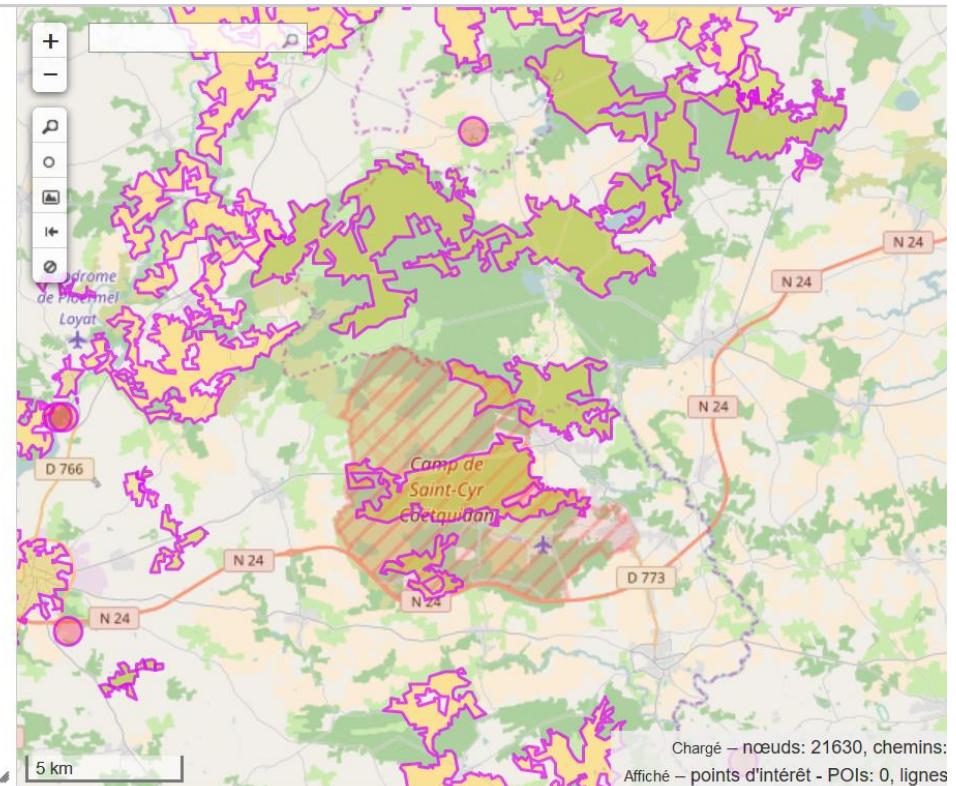


Requêtes classiques



Extraire les zones renseignées sur l'occupation des sols

```
/*
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"]({{bbox}});
);
// print results
out body;
>;
out skel qt;
```



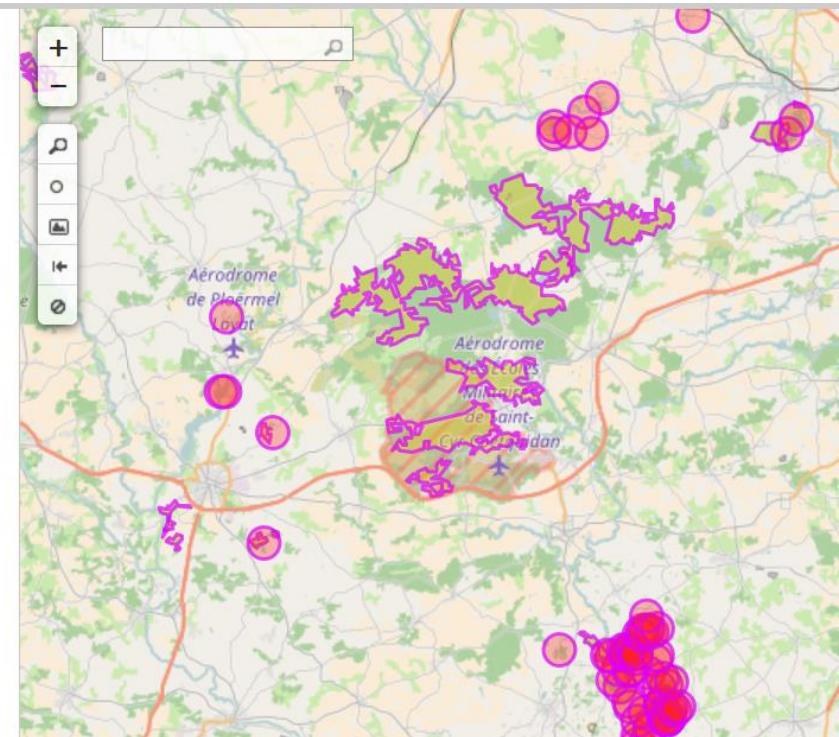


Requêtes classiques



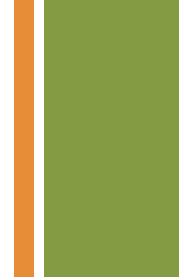
Extraire les zones renseignées comme forêt

```
/*
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



Sélectionner les lignes grandes vitesses

```
1 [out:json][timeout:25];
2 way["railway"="rail"]["highspeed"="yes"]({{bbox}});
3 // print results
4 out body;
5 >;
6 out skel qt;
```



+

Requêtes complexes



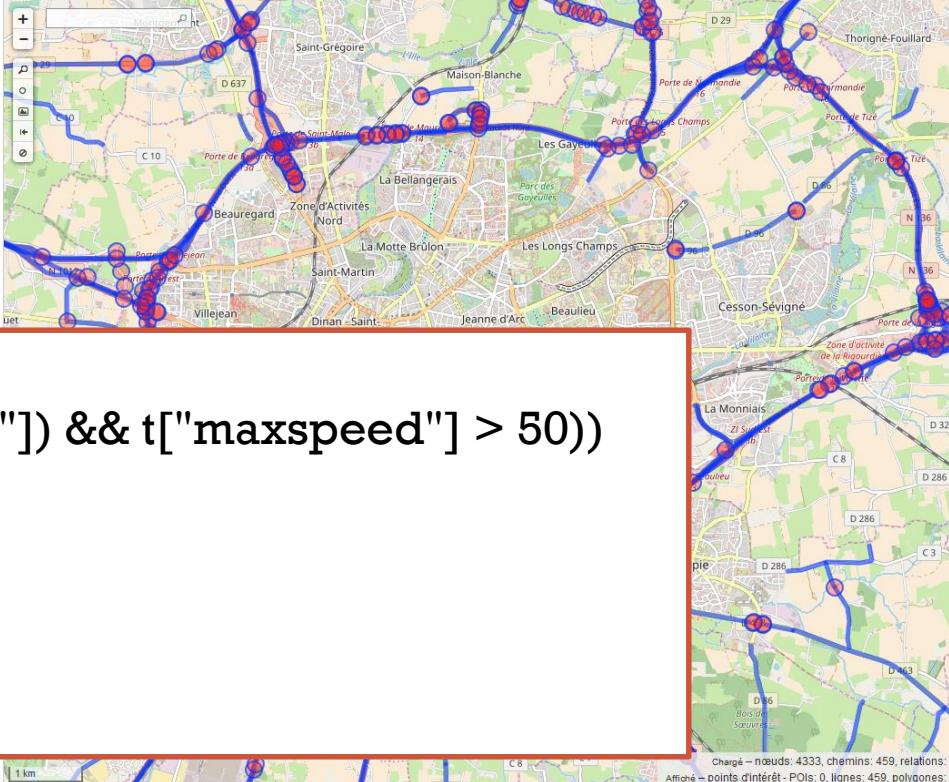
Requêtes complexes



Critère minimum ou maximum

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

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



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



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 |
```

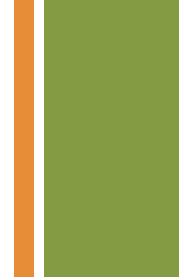
```
(way["highway"]
  (if: (is_number(t["lanes"])) && t["lanes"] > 1) ({{bbox}}));
(way["highway"]
  (if: (is_number(t["maxspeed"])) && t["maxspeed"] > 50) ({{bbox}}));
);
out body;
>;
out skel qt;
```

2 km

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

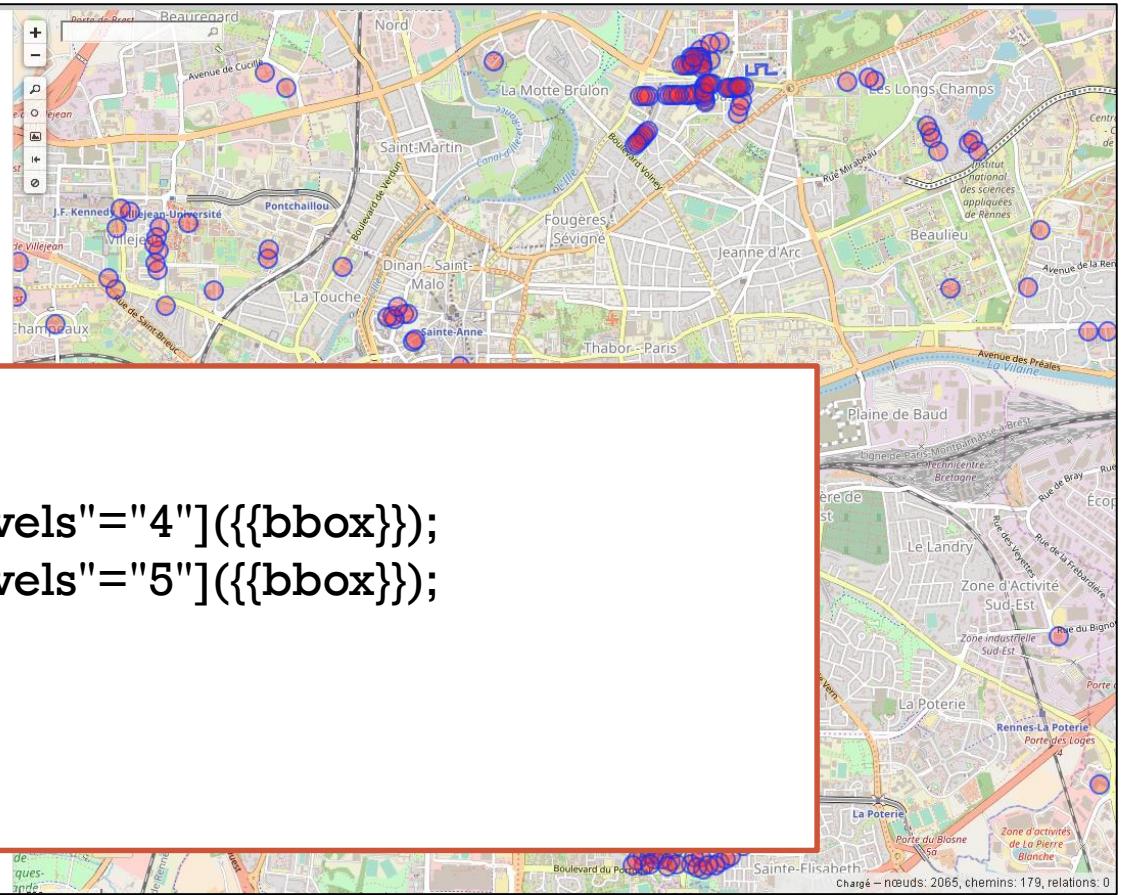


Requêtes complexes



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;
```

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

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		



Contrôle données



Eléments sans tags

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

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

Carte Données

way({{bbox}})(if:count_tags() == 0);
out geom;

Chargé – noeuds: 0, chemins: 14, relations: 0
Affiché – points d'intérêt - POIs: 0, lignes: 14, polygones: 0



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;
```

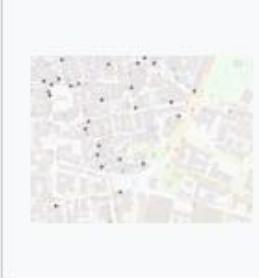
```
way
["highway"~"primary|secondary|tertiary|trunk|service|residential|primary_link|
secondary_link|tertiary_link|unclassified"]["access"!~"no|private"]![!area]
({{bbox}});
(way(around:0)["building"~"."]);
node(w););
out meta;
```



line styles



color coding



icons



a simple thematic map



markers with text

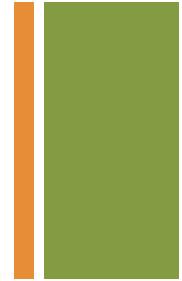
+

Requêtes et style

https://wiki.openstreetmap.org/wiki/Overpass_turbo/MapCSS



Requête et style



Afficher des étiquettes

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

```
1
2
3 node["public_transport"]=="stop_position" ["subway"]=="yes"]({{bbox}});
4
5 {{style:
6   node, way, relation { text: name; }
7 }}
8
9
10 out body;
11 >;
12 out skel qt;
```

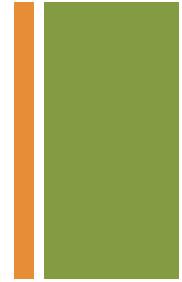
Carte Données

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

Chargé – noeuds: 59, chemins: 0, relations: 0
Affiché – points d'intérêt - POIs: 59, lignes: 0, polygones: 0
22-24



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;
```

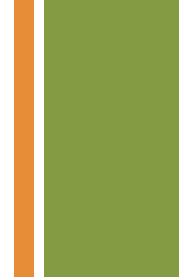
Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo ☰

```
1 [out:json][timeout:25];
2
3 {
4   way[building="apartments"]
5   { color:blue; fill-color:blue; }
6
7   way[building="residential"]
8   { color:blue; fill-color:blue; }
9
10
11   way[building="house"]
12   { color:red; fill-color:red; }
13
14   way[building="school"]
15   { color:red; fill-color:green; }
16
17 }
18
19 // print results
20 out body;
21
22 out skel qt;
```





Requête et style



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

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

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

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

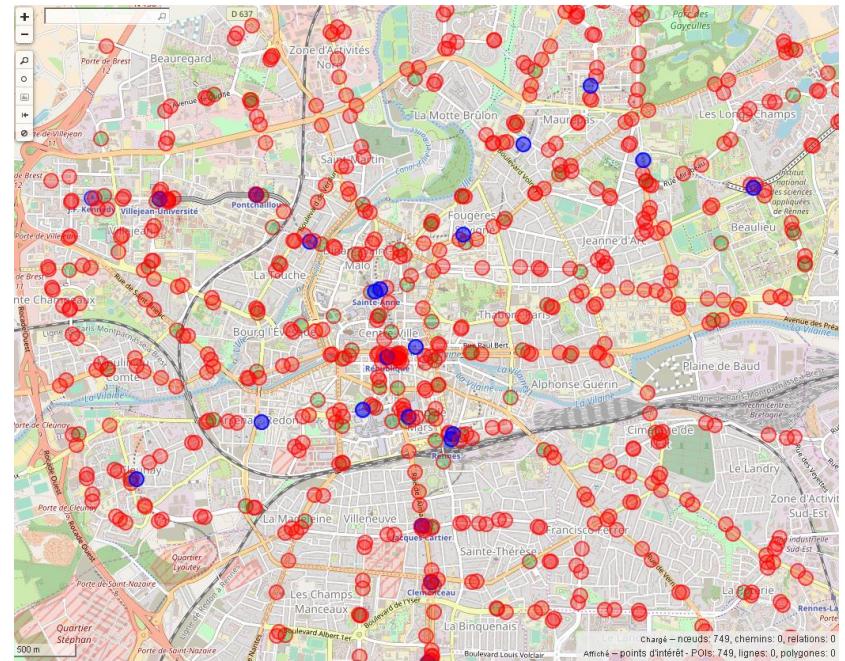
{{style:

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

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

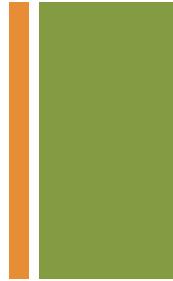
node[amenity=bicycle_rental]
{ color:red; fill-color:green; }
}};

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





Requête et style



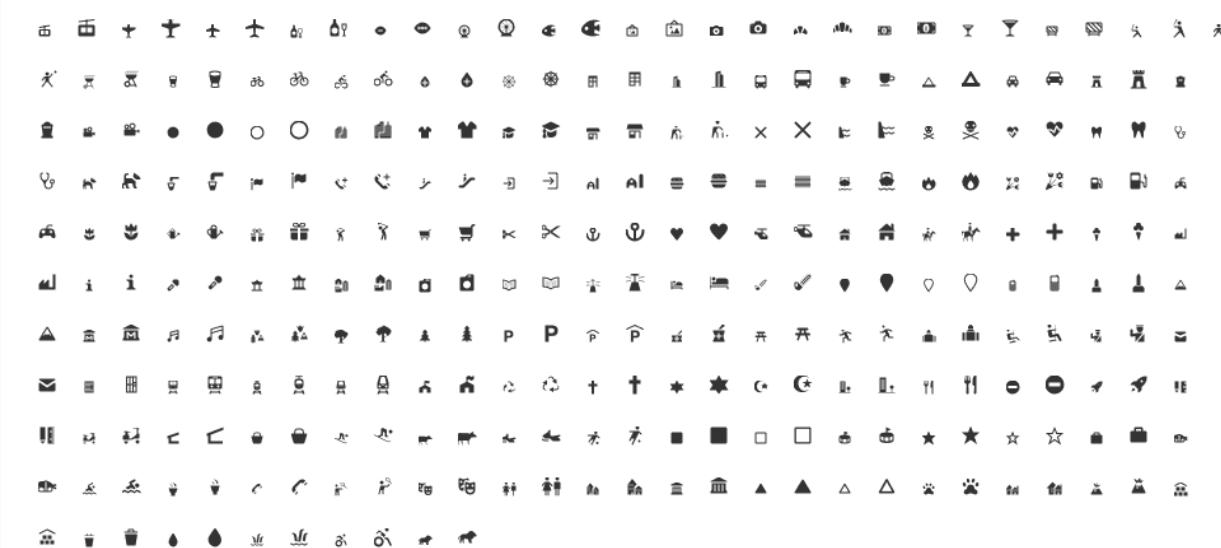
Utiliser des icônes (maki)

<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)

Exécuter Partager Exporter Assistant Enregistrer Charger Paramètres Aide overpass turbo

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

Carte **Données**

The map shows the city of Rennes with its street network, including Avenue Charles Tillon, Rue de la Bascule, and Rue de la Gare. Landmarks like the 'Complexe sportif de la Harpe' and 'Université de Rennes 2 - Campus de Villejean' are visible. Public transport stops, particularly bus stops, are clearly marked with large, black, bus-shaped icons, making them stand out from other map elements.

{}{style:
node[highway=bus_stop] {
icon-image: url('icons/maki/bus-18.png');
icon-width: 18;}
}

- nœuds: 745, chemins: 0, relations: 0
- POIs: 745, lignes: 0, polygones: 0



Requête et style



Mise en forme des sentiers

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

```
[bbox:{bbox}];  
  
(relation[route=hiking][network~"^wn$"];way(r);>);out;  
  
{  
  style:  
  
    relation[network=lwn] way { color:blue; fill-color:cyan; }  
    relation[network=iwn] way { color:red; fill-color:red; }  
    relation[network=nwn] way { color:green; fill-color:green; }  
    relation[network=rwn] way { color:yellow; fill-color:yellow; }  
}  
}
```



Requête et style



Mise en forme des sentiers

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
```

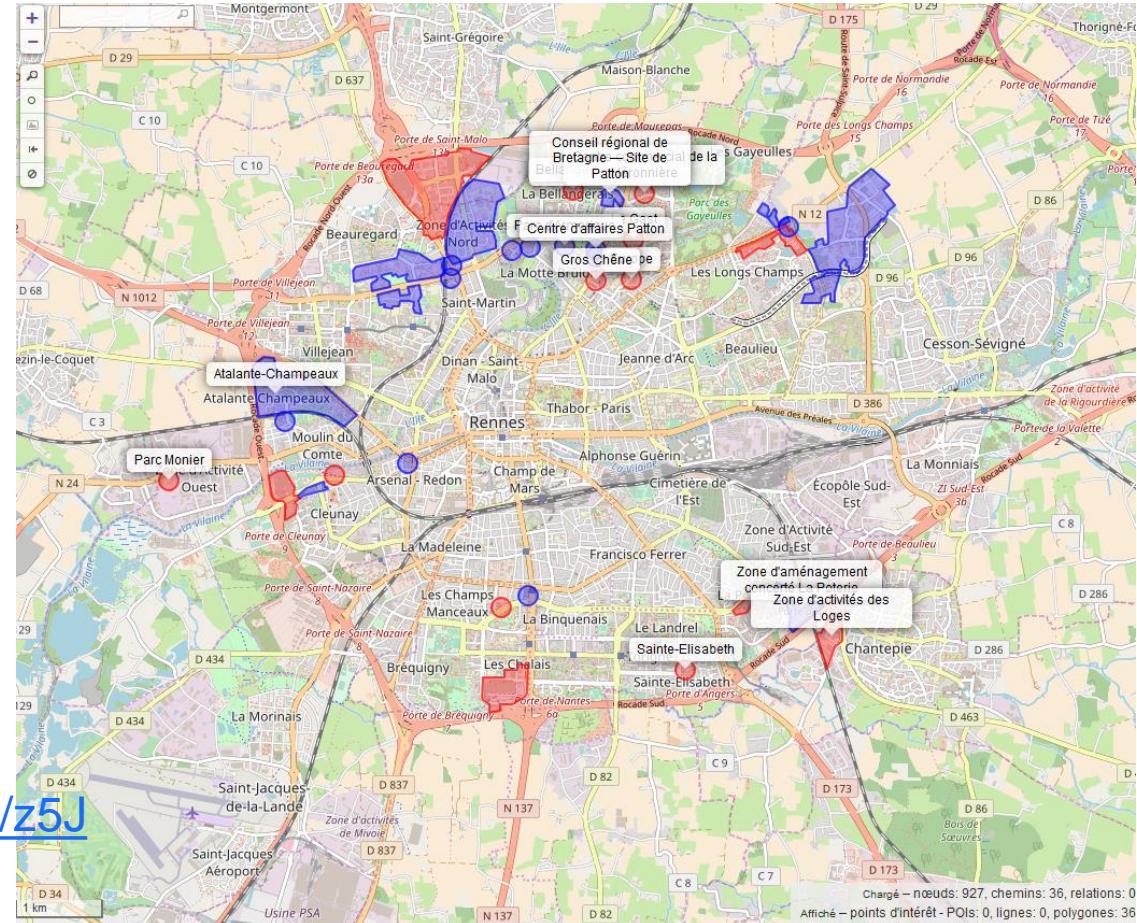
Chargé – noeuds: 75447, chemins: 3024, relations: 65
Affiché – points d'intérêt - POIs: 490, lignes: 3089, polygones: 0



Requête et style



Sélectionner les zones commerciales avec un affichage stylisé et des étiquettes



**+ Requêtes
contributors / date**



Extraction par contributeur

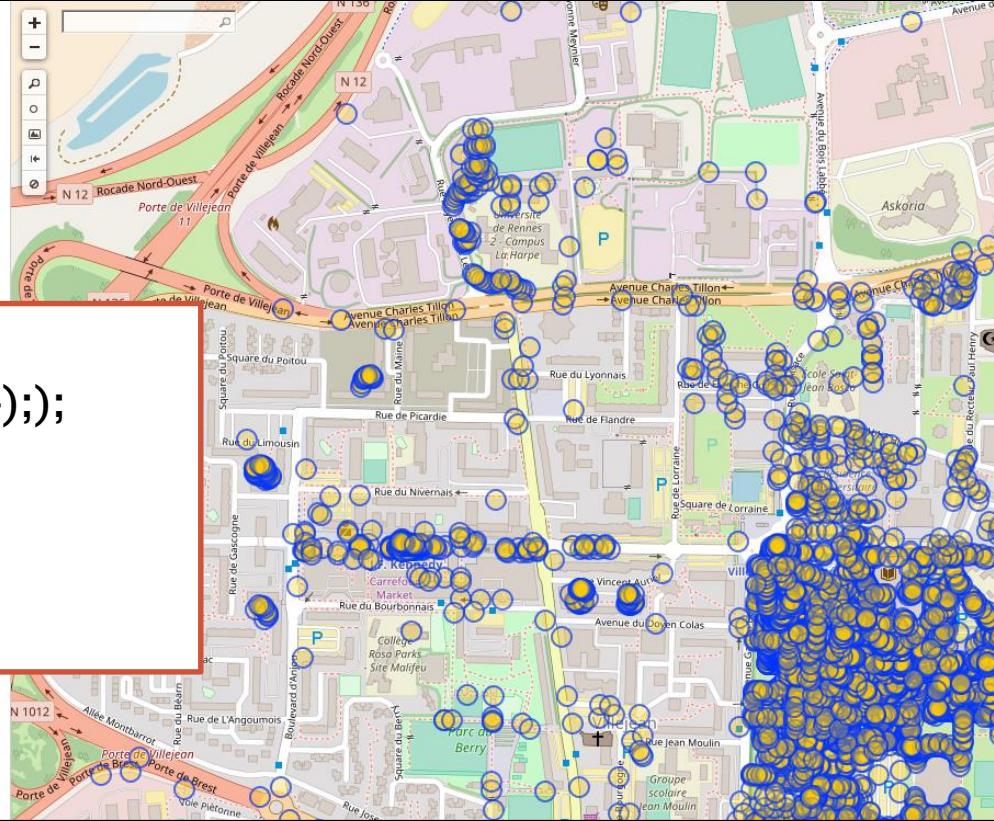


```
[out:json][timeout:25];
(node(user:PanierAvide)({{bbox}}));;
```

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

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

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





Extraction par date



Sélectionner les nouvelles aménités depuis le 1^{er} mai 2018

```
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:"2018-05-01T00:00:00Z")({{bbox}});
);
out body;
>;
out skel qt;
```



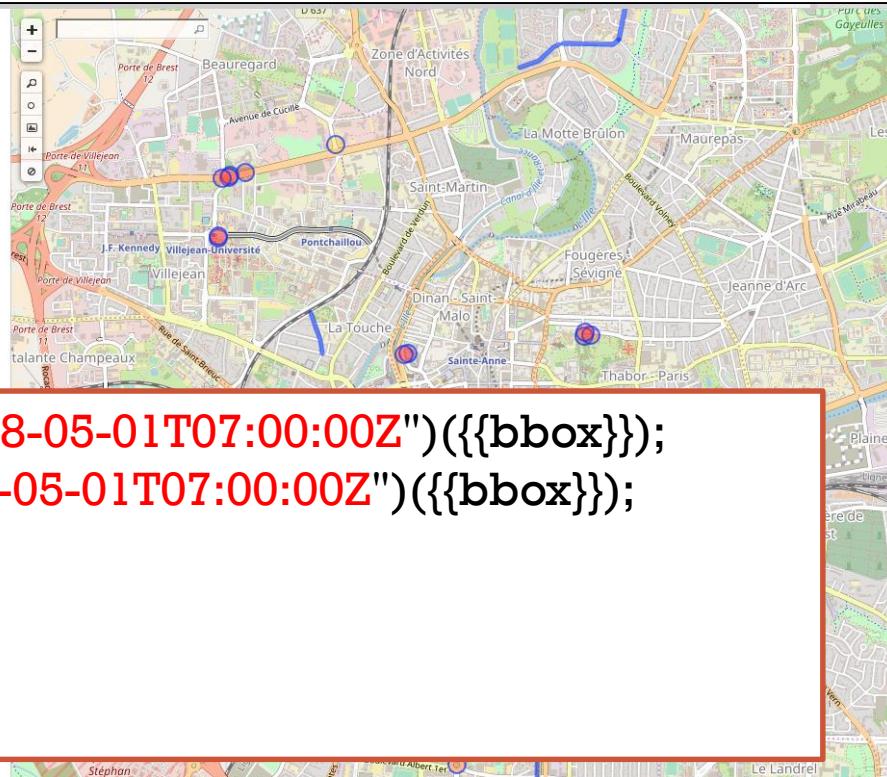


Extraction par date



Sélectionner les routes mises à jour depuis le 1^{er} mai 2018

```
[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;
```



```
( node["highway"] (changed:"2018-05-01T07:00:00Z")({{bbox}});
way["highway"] (changed:"2018-05-01T07:00:00Z")({{bbox}});
);
out body;
>;
out skel qt;
```

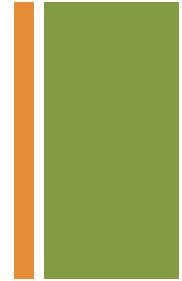


+

Requêtes spatiales

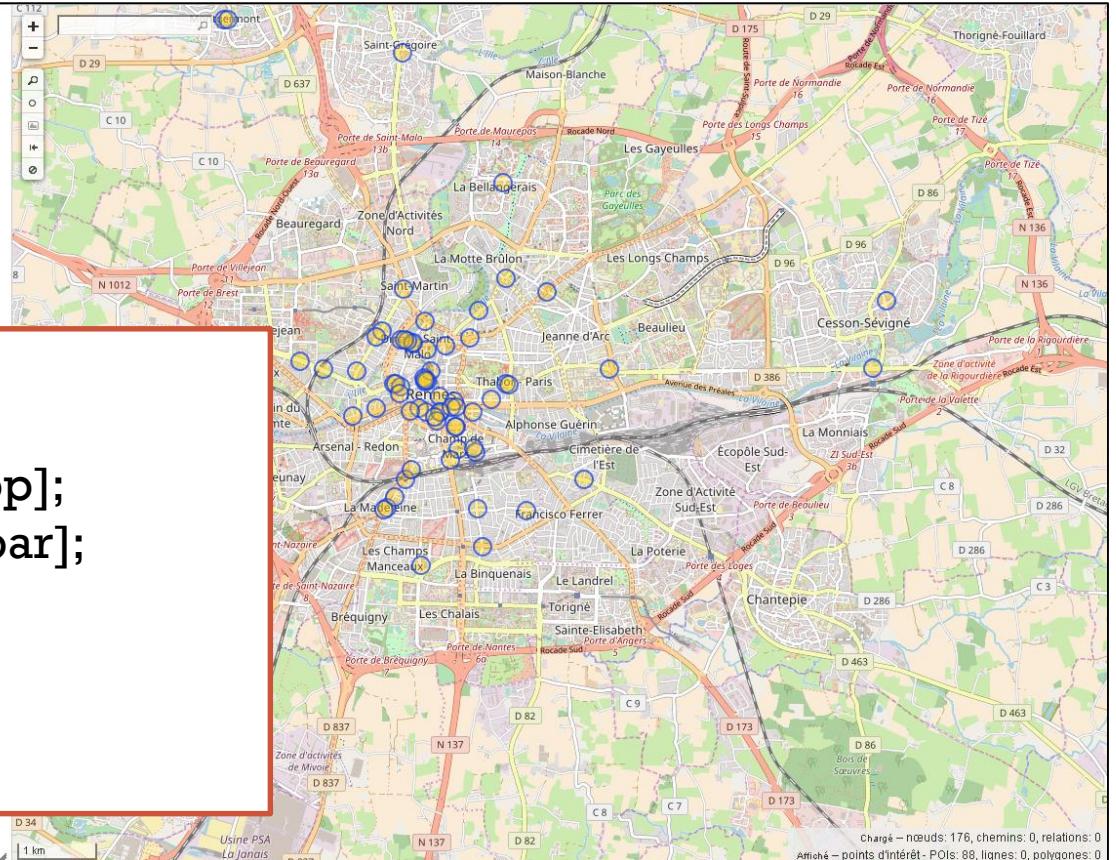


Sélection spatiale



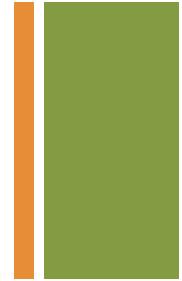
Sélectionner tous les bars à moins de 100m d'un arrêt de bus

```
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 neighborhoods labeled such as Villejean, Pontchaillou, La Touche, and Centre. Numerous blue circles of varying sizes are scattered across the map, representing bars located within 200 meters of subway stations. The map also shows the network of roads, rivers (Vilaine and 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. The map includes labels for neighborhoods like Villejean, Pontchâtel, La Motte-Bruzon, and Beaulieu, as well as major roads and landmarks.

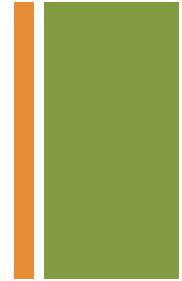
Carte Données

[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

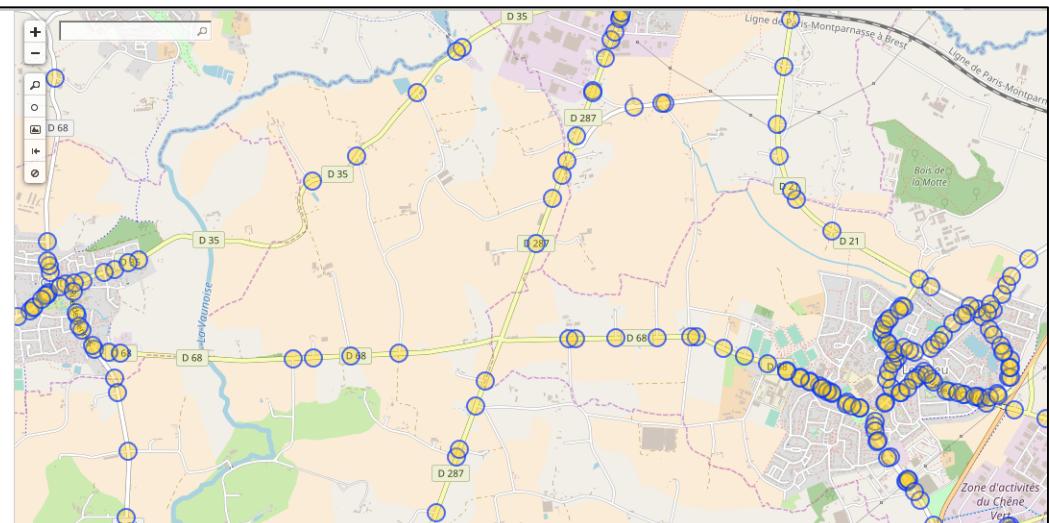


Sélection spatiale



Sélectionner les intersections entre routes principales et petites routes

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



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

