

6) Tracé de bâtiments

6.1) Exploiter les données *Open Street Map*

Afin de les grouper en ensembles cohérents, les bâtiments d'*Open Street Map* ont fait l'objet d'un traitement préalable par vos enseignants :

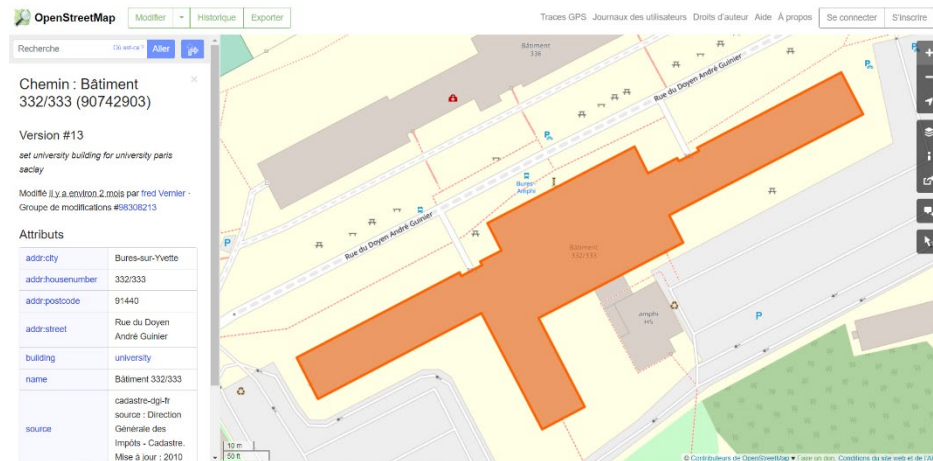


Figure 23 - Bâtiments OSM

Les exports de ces ensembles de bâtiments vous sont fournis dans les fichiers « *buildings_... .geoJson* » du répertoire « *data* ».

Dupliquez votre classe *Roads* pour créer votre nouvelle classe *Buildings* pour gérer les bâtiments. Le constructeur de cette classe appelé par *this.buildings = new Buildings(this.map)*; construira une *PShape buildings* de type *GROUP* qui sera laissée vide à la construction.

Une fois l'objet construit, vous ajouterez les ensembles de bâtiment en spécifiant la couleur souhaitée pour chaque ensemble, afin de constituer votre groupe de formes :

```
// Prepare buildings
this.buildings = new Buildings(this.map);
this.buildings.add("buildings_city.geojson", 0xFFaaaaaa);
this.buildings.add("buildings_IPP.geojson", 0xFFCB9837);
this.buildings.add("buildings_EDF_Danone.geojson", 0xFF3030FF);
this.buildings.add("buildings_CEA_algorithmes.geojson", 0xFF30FF30);
this.buildings.add("buildings_Thales.geojson", 0xFFFF3030);
this.buildings.add("buildings_Paris_Saclay.geojson", 0xFFee00dd);
```

6.2) Transformation de polygones en bâtiments 3D

Cette fois-ci, les fichiers *geoJSON* contiennent des *features* dont la géométrie est de type « *polygon* ».

Ces éléments sont des tableaux de polygones (un bâtiment peut être défini par plusieurs polygones) qui marquent l'emplacement au sol du bâti.

Vous devrez donc, à partir des coordonnées au sol, construire une *PShape walls* qui décrira les murs, ainsi une *PShape roof* qui décrira le toit. Comme pour vos voies, vous veillerez à ne prendre que les coordonnées de bâtiments à l'intérieur du terrain, la méthode *ObjectPoint.inside()* de *Map3D* vous permettant de vérifier cette condition

Vous déterminerez une hauteur approximative du bâtiment en tenant compte du nombre d'étages s'il est indiqué dans les propriétés. Par exemple, on élèvera chaque mur de 3 mètres par étage à partir du point d'élévation maximale en faisant :

```
int levels = properties.getInt("building:levels", 1);
float top += Map3D.heightScale * 3.0f * (float)levels;
```

Afin que vos murs paraissent plus sombres que les toits, vous pourrez soit diminuer la luminosité de la couleur passée en paramètre, soit jouer sur l'émissivité des matériaux (i.e. *walls.emissive(0x30)* et *roof.emissive(0x60)*).

Vous pourrez enfin affiner le calcul des normales des murs, en calculant le produit vectoriel de chaque segment de tracé au sol avec sa verticale associée. Pour les toits, la normale sera constante si vous les tracez à plat sur un plan $z = top$.

Vous ajouterez à votre classe les méthodes *update()* et *toggle()* pour gérer l'affichage et la visibilité des bâtiments, que vous affecterez à la touche « *B* » du clavier.

Félicitations, vous disposez d'une vue relativement réaliste¹⁵ de l'implantation des bâtiments sur votre terrain :

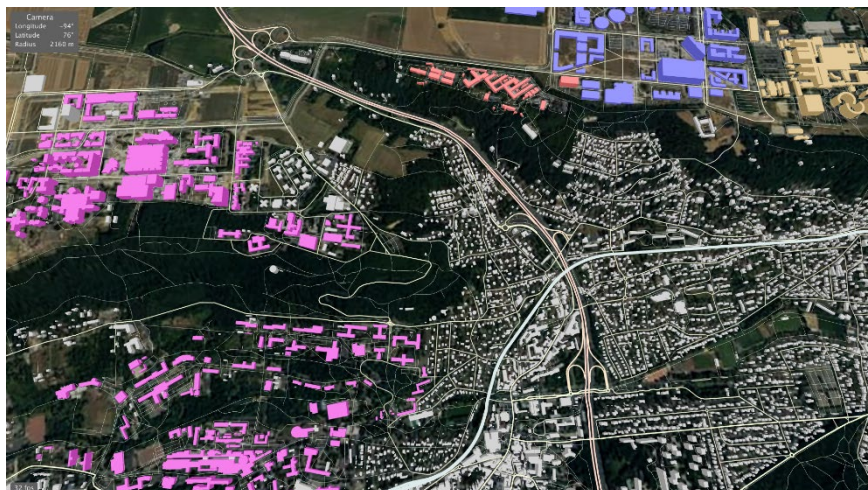


Figure 24 - Tracé des bâtiments

¹⁵ Vous n'avez pas à traiter les découpes intérieures (extrusions) de certains bâtiments complexes.