

# Foss4g-fr 2014

## Workshop

Créez votre propre fond de plan à partir de données OSM en utilisant TileMill

Sylvain beorchia

[www.makina-corpus.com](http://www.makina-corpus.com)

[sylvain.beorchia@makina-corpus.com](mailto:sylvain.beorchia@makina-corpus.com)



[www.makina-corpus.com](http://www.makina-corpus.com)

**Des fonds de plan personnalisés ?**





Recherche

Où suis-je ?

OK

## Bienvenue dans OpenStreetMap !

OpenStreetMap est une carte du monde, créée par des gens comme vous et libre d'utilisation sous licence libre.

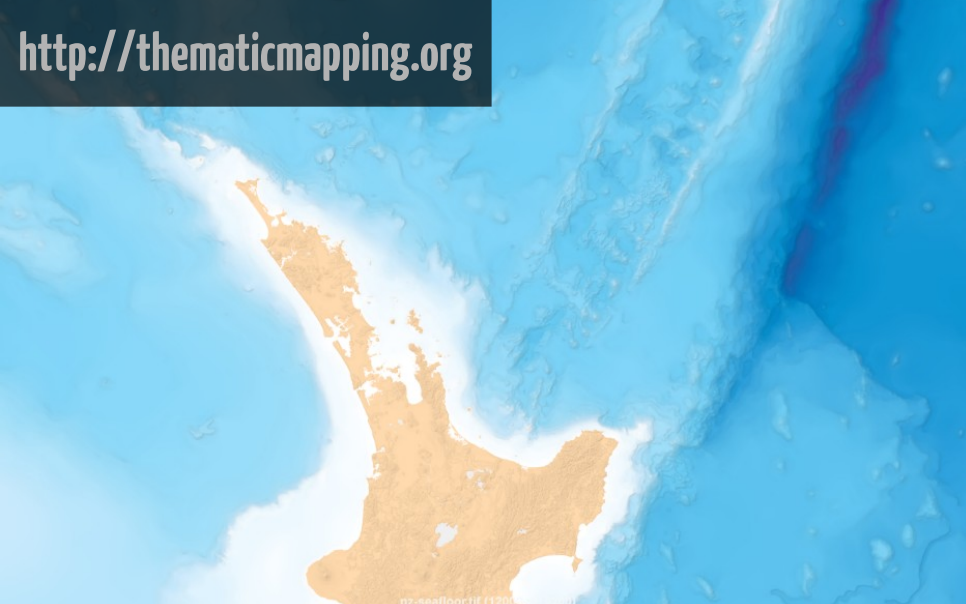
En savoir plus

Commencer à cartographier



openstreetmap.org



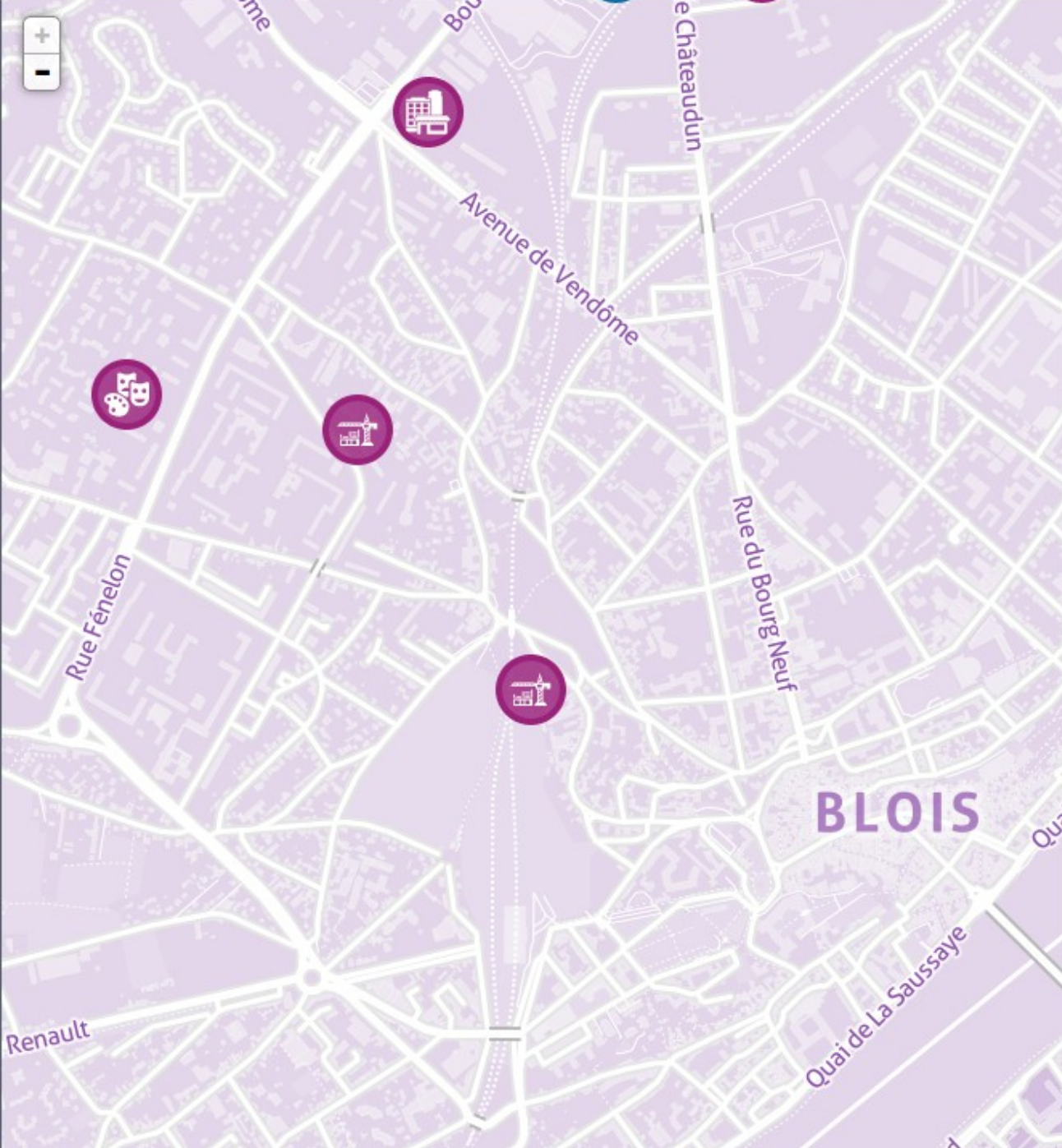


# Styles libres réutilisables



*Développeurs de nouveaux logiciels*





## Une INSA à Blois

L'Institut national des sciences appliquées Centre Val de Loire naîtra de la fusion de l'École nationale d'ingénieurs du Val de Loire de Blois et de l'École nationale supérieure d'ingénieurs de Bourges : il sera donc bi-site.

**Objectifs** : renforcer la visibilité des formations d'ingénieurs dans la région et accroître les effectifs d'étudiants tout en bénéficiant d'une marque forte. 1 700 élèves pourraient être accueillis d'ici à 5 ou 6 ans : 700 à Blois et 985 à Bourges.

Chacune des deux écoles compte actuellement environ 500 étudiants, auxquels s'ajoutent 300 élèves de l'école Hubert Curien de Bourges, (centre de formation supérieure en apprentissage géré par la Chambre de commerce et d'industrie du Cher).

**Ouverture** : rentrée 2013. Les élèves entrés dès septembre 2013 auront,



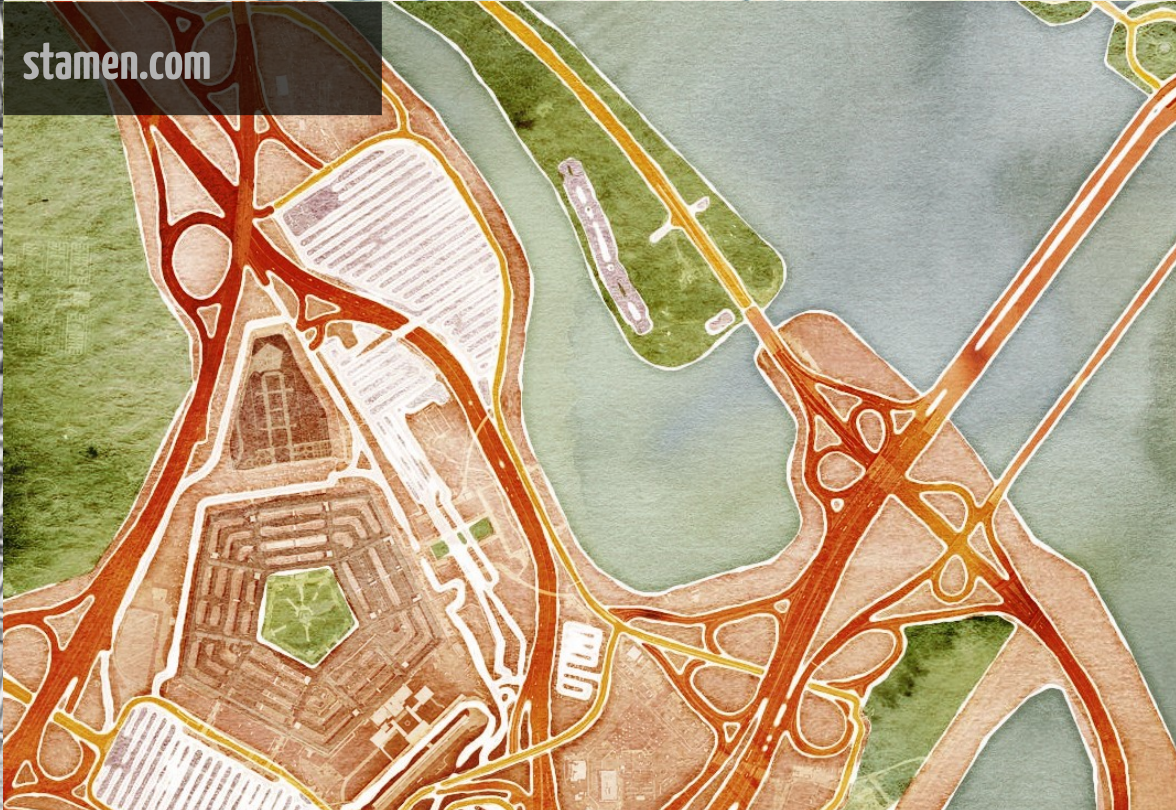
Livraison  
Rentrée 2013



# Charte et carte !

<http://loir-et-cher-2020.makina-corpus.net>









Par *Makina Corpus*

## PLAN DE TOULOUSE En l'An 1680

*Sources*

# Possibilités infinies...

<http://tolosa1680.makina-corpus.com>



*Développeurs de nouveaux horizons*

# Étapes

- **Installer TileMill**
- **Installer Postgres/Postgis**
- **Installer Imposm**
- **Charger les données OSM en base**
- **Créer un projet TileMill à partir du projet OSM bright**
- **Lancer TileMill et customiser le fond de plan**
- **Exporter les données / extraire les tuiles**
- **Tester (websig utilisant les tuiles)**



# Installer TileMill

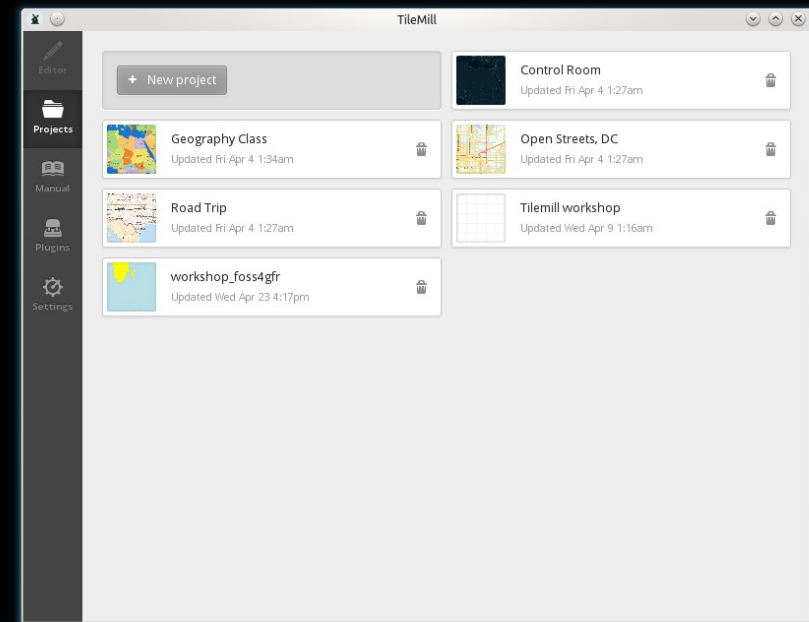
Procédure d'installation :

<https://www.mapbox.com/tilemill/docs/install>

Pour linux :

<https://www.mapbox.com/tilemill/docs/linux-install/>

Suivre la procédure d'installation.





# Installer Postgres/Postgis

Sous linux (debian / ubuntu) :

```
sudo apt-get install postgresql postgresql-9.1-postgis
```

(

Sous windows :

<http://www.postgresql.org/download/windows/>

)





# Installer Imposm

Outil d'import de données OSM vers Postgres/gis  
(<http://imposm.org/>)

En ligne de commande :

```
> sudo aptitude install build-essential python-dev protobuf-compiler  
libprotobuf-dev libtokyocabinet-dev python-psycopg2 libgeos-c1  
> sudo apt-get install python-pip  
> sudo pip install imposm
```

RQ : Il existe d'autres outils pour importer des données OSM, comme osm2pgsql



# Charger les données OSM en base (1/3)

## Créer la base

```
sudo su - postgres
```

```
psql -U postgres -c "create database osm;"
```

```
psql -U postgres -d osm -c "create extension postgis;"
```





# Charger les données OSM en base (2/3)

Récupérer les données OSM :

<http://metro.teczno.com>

<http://download.geofabrik.de/osm/>



Télécharger les données au format .osm.pbf  
(Protocolbuffer Binary Format)



## Charger les données OSM en base (3/3)

### Charger les données dans la base

```
imposm -U gisuser -d osm -m ~/Documents/mapbox-osm-  
bright-86bc63f/imposm-mapping.py --read --write --optimize  
--deploy-production-tables ~/Downloads/basse-normandie-  
latest.osm.pbf
```

Cette commande peut prendre du temps. Sur notre exemple (données du Calvados, 20 minutes de traitement)



# OSM bright comme projet de départ (1/3)

**Télécharger le projet OSM Bright**

<https://github.com/mapbox/osm-bright/zipball/master>

**Enregistrer et décompresser dans un répertoire**

**A télécharger en plus :**

<http://tilemill-data.s3.amazonaws.com/osm/coastline-good.zip>

[http://tilemill-data.s3.amazonaws.com/osm/shoreline\\_300.zip](http://tilemill-data.s3.amazonaws.com/osm/shoreline_300.zip)

<http://mapbox-geodata.s3.amazonaws.com/natural-earth-1.3.0/physical/10m-land.zip>

**A placer dans le répertoire de OSM Bright.**

## OSM bright comme projet de départ (2/3)

Configurer le projet :

Copier `configure.sample.py` vers `configure.py`

Modifier ce fichier :

- `config["importer"] = "imposm"`
- connexion à la base
- nom du projet
- path vers les projets tilemill



# OSM bright comme projet de départ (3/3)

Générer le projet :

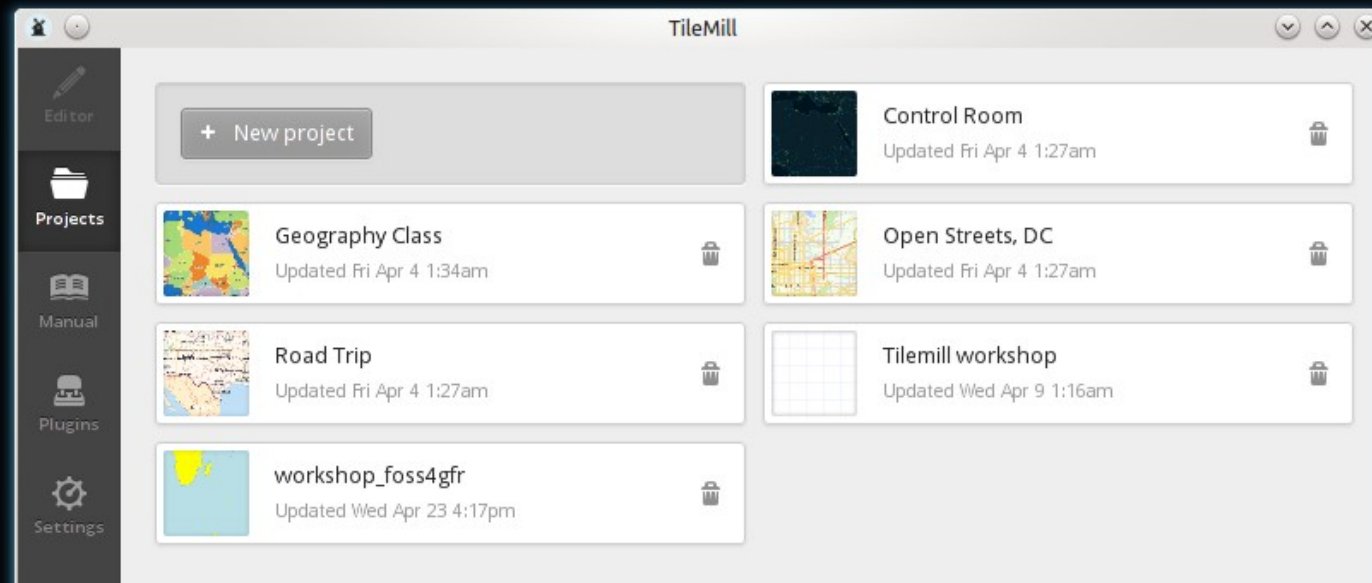
```
cd ~/Documents/mapbox-osm-bright-86bc63f  
./make.py
```

# Customiser le fond de plan : lancer TileMill

## Lancer TileMill

## Sélectionner le projet précédemment créé

Au premier chargement des données complémentaires seront téléchargées (donc un chargement plus long la première fois)





# TileMill

The screenshot displays the TileMill application interface, which is used for creating map tiles. The interface is divided into several main sections:

- Left Sidebar:** Contains navigation icons for Editor, Projects, Manual, Plugins, and Settings.
- Map View:** Shows a street map of a city area. A blue box labeled "Carte" is overlaid on the map.
- Project Settings:** A blue box labeled "Paramètres projets" is overlaid on the top left of the map view.
- Layers / Aides / Fonts:** A blue box labeled "Couches / Aides / Fonts" is overlaid on the bottom left of the map view.
- Styles Panel:** A blue box labeled "Styles" is overlaid on the right side of the map view, showing a list of styles.
- Code Editor:** A blue box labeled "Couleurs" is overlaid on the bottom right of the map view, showing a color palette.
- OSM workshop 2014:** A panel on the right side of the interface showing a list of styles and a code editor with XML-style markup for map features.

The OSM workshop 2014 panel includes a list of styles and a code editor with the following content:

```
348 /* ===== */
349 /* AEROWAYS
350 /* ===== */
351
352 #aeroway[zoom>9] {
353   line-color:@aeroway;
354   line-cap:butt;
355   line-join:miter;
356   [type='runway'] {
357     [zoom=10]{ line-width:1; }
358     [zoom=11]{ line-width:2; }
359     [zoom=12]{ line-width:3; }
360     [zoom=13]{ line-width:5; }
361     [zoom=14]{ line-width:7; }
362     [zoom=15]{ line-width:11; }
363     [zoom=16]{ line-width:15; }
364     [zoom=17]{ line-width:19; }
365     [zoom>17]{ line-width:23; }
366   }
367   [type='taxiway'] {
368     [zoom=10]{ line-width:0.2; }
369     [zoom=11]{ line-width:0.2; }
370     [zoom=12]{ line-width:0.2; }
371     [zoom=13]{ line-width:1; }
372     [zoom=14]{ line-width:1.5; }
373     [zoom=15]{ line-width:2; }
374     [zoom=16]{ line-width:3; }
375     [zoom=17]{ line-width:4; }
376     [zoom>17]{ line-width:5; }
377   }
378 }
379
380 /*****
381
382
```

# Customiser le fond de plan : modifier la symbologie !

Documentation en ligne :

<https://www.mapbox.com/tilemill/docs/crashcourse/styling/>

## Changer la couleur de la mer :

Dans le fichier palette.mss : `@water: #C4DFF6;`

## Modifier la 3D des bâtiments :

Fichier base.mss : `building-height:3.25;`

## Ne pas oublier de sauver le projet pour voir les modifications !



# Customiser le fond de plan : styles conditionnels

On a une couche des batiments

En editant la couche, on voit qu'il s'agit de la table **osm\_buildings**

Dans un terminal :

```
sudo su - postgres
```

```
psql -d osm
```

```
> select distinct(type) from osm_buildings;
```

On va prendre un type et le mettre en valeur

# Customiser le fond de plan : styles conditionnels

Prenons le type : **school**

Dans **base.mss**, on va modifier des lignes concernant l'affichage des buildings par :

```
#buildings[zoom>=17][type != 'school'] {  
    building-fill:@building;  
    building-height:0.25;  
}  
#buildings[zoom>=17][type = 'school'] {  
    building-fill:@wooded;  
    building-height:10.25;  
}
```



Customiser le fond de plan : styles extrêmes

Ajouter un style « dessiné » aux bâtiments :

Télécharger la trame :

<http://foss4g.2tokiislands.com/trame.png>

Et l'enregistrer dans le [projet/img/](#)

Appliquer cette trame aux bâtiments :

[Base.mss](#) , à la fin du bloc des buildings, rajouter :

```
#buildings {  
  polygon-pattern-file: url("img/trame.png");  
}
```

# Customiser le fond de plan : styles extrêmes

**Batiments : avoir des contours « crayonnés »**

```
#buildings {  
    line-smooth:2;  
    line-color: #CCCCCC;  
    line-opacity:0.8;  
}
```

# **Customiser le fond de plan : modifier les données**

**Bien sûr il est possible de sélectionner / filtrer / modifier les données dans postgres, et ajouter de nouvelle couche dans le projet TileMill...**



## Customiser le fond de plan : ajouter des éléments graphiques (1/2)

### Ajouter un effet « papier plié » :

Copier le répertoire **texture** du projet **geography-class** dans notre projet

Copier le fichier **layers/world\_extent\_wgs84.geojson** de **geography-class** dans notre projet

Ajouter une nouvelle couche nommée « **paper** », de type file :

datasource : **layers/world\_extent\_wgs84.geojson**

srs: **wgs84**

id: **paper**

## Customiser le fond de plan : ajouter des éléments graphiques (2/2)

### Ajouter le style dans le CSS :

```
#paper[zoom<2] {  
    polygon-pattern-file:url(textures/paperfolds_256.png);  
}  
#paper[zoom>1] {  
    polygon-pattern-file:url(textures/paperfolds_512.png);  
}
```

# Exporter les données

Utiliser la fonction d'export de TileMill

Choisir le format **.MBtiles**

Attention à la taille ! (réduire les niveaux de zoom si nécessaire)



# Extraire les tuiles

## Utilitaire MButil pour extraire les tuiles

### Installation

```
wget https://github.com/mapbox/mbutil/zipball/master -O mbutil.zip  
unzip mbutil  
cd mapbox-mbutil*  
sudo python setup.py install
```

### Utilisation

```
cd /var/www/data/  
mb-util --scheme=xyz mon_fichier.mbtiles workshop
```

**Tester (websig utilisant les tuiles)**

**Librairie cliente (OpenLayers / Leaflet)**

**Chargement d'une couche de type Tile**

**Cf exemple**

# Quelques liens supplémentaires

<https://www.mapbox.com/tilemill/docs/guides/advanced-map-design/>

<http://makina-corpus.com/blog/metier/2014/dessiner-une-ville-a-la-main-les-batiments>

<http://tolosa1680.makina-corpus.com/>

<http://makina-corpus.com/blog/societe/2014/dessine-moi-une-ville-image-of-the-week-du-wi>

<http://stamen.com/>

<http://www.openstreetmap.org>



**Merci**

