Maputnik

Making your own map with open source tools

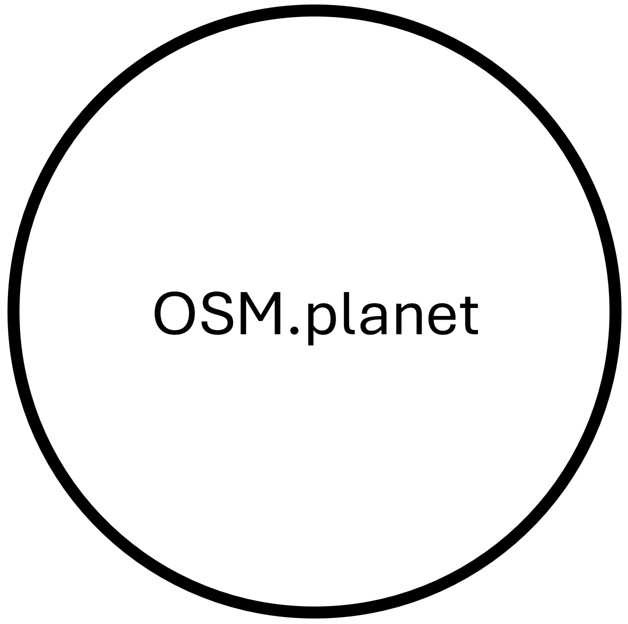
What is **Maputnik?**

“A free and open visual editor for the MapLibre GL styles targeted at developers and map designers.”

Put **simply** its an easy way to make your own maps with an **editor** you can **easily** interact with in your browser

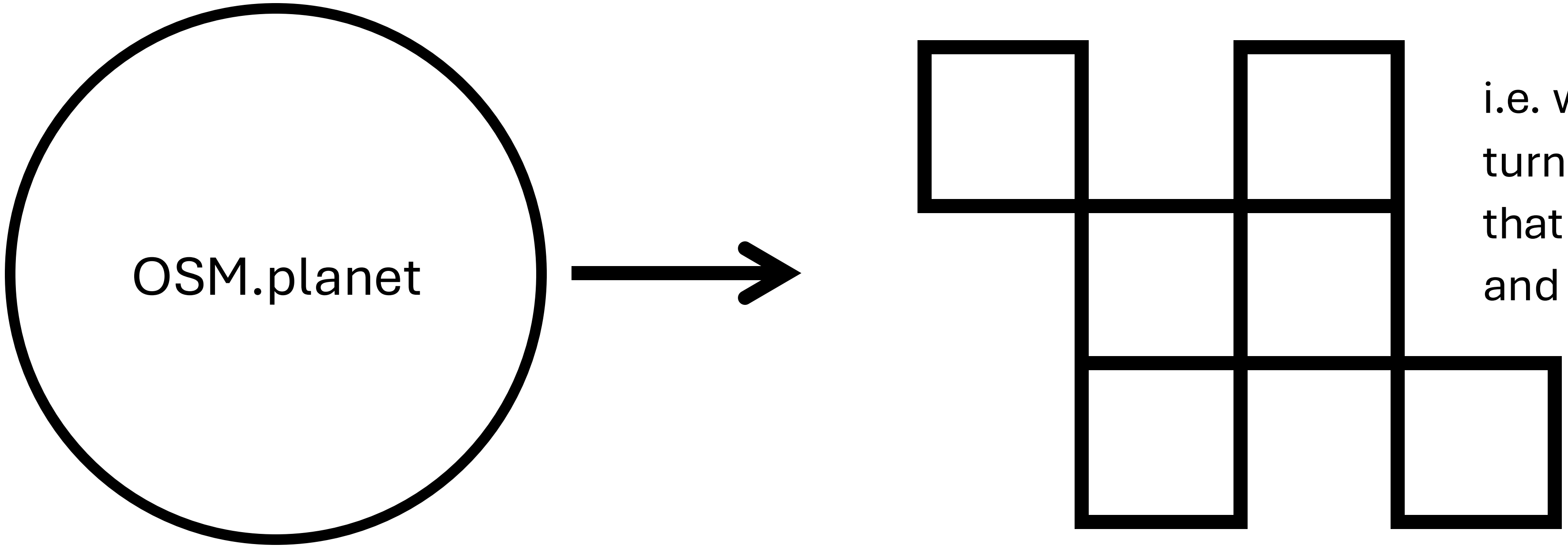
But what is **Maputnik?**

That’s a little more complicated…



But what is **Maputnik?**

It used to be ...

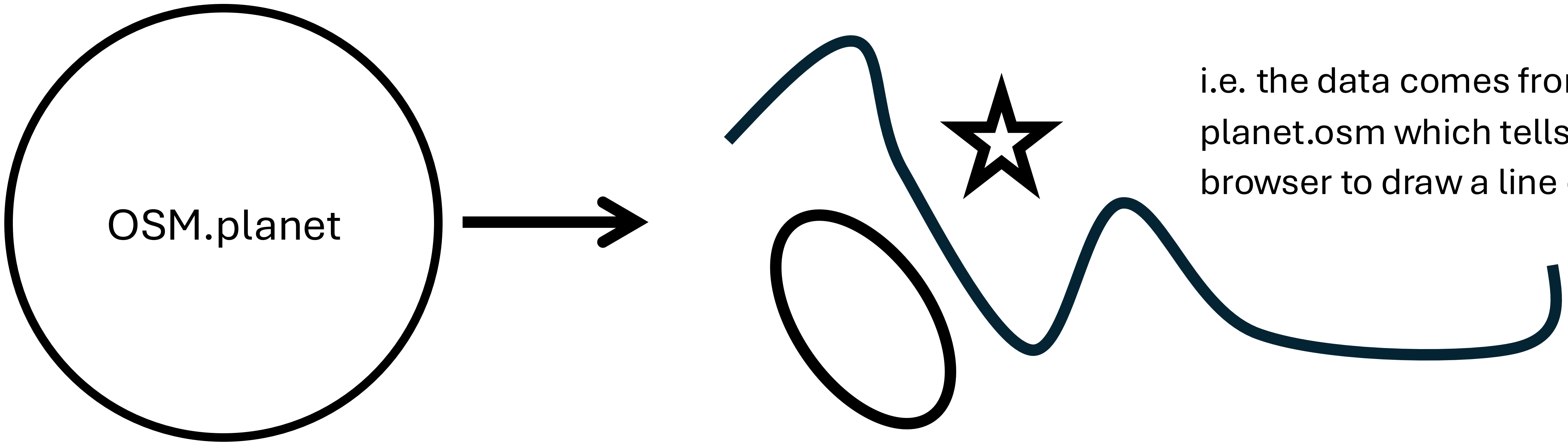


i.e. we would run code to turn osm.planet into images that you could then zoom in and out of

But what is **Maputnik?**

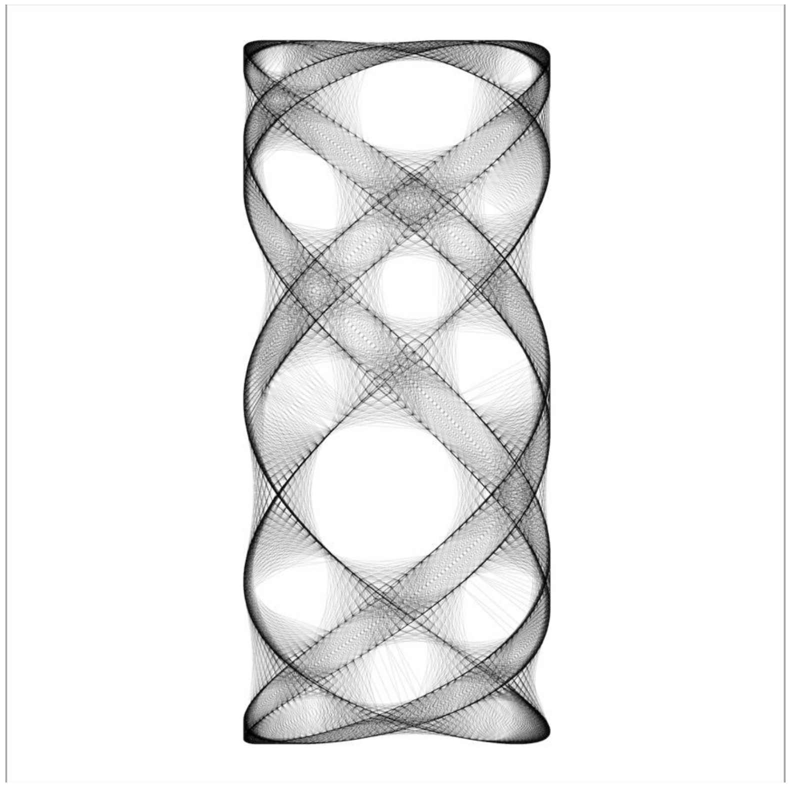
Now …

Vectors rendered by your browser



i.e. the data comes from planet.osm which tells your browser to draw a line etc

In principle this is the same as vectors in design software where you can keep zooming in



vector

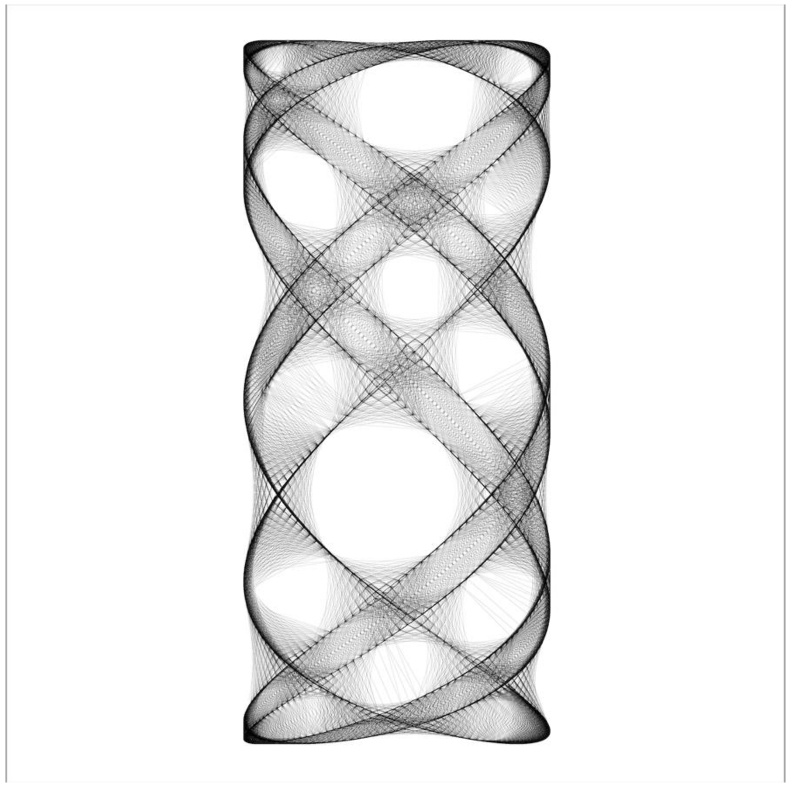
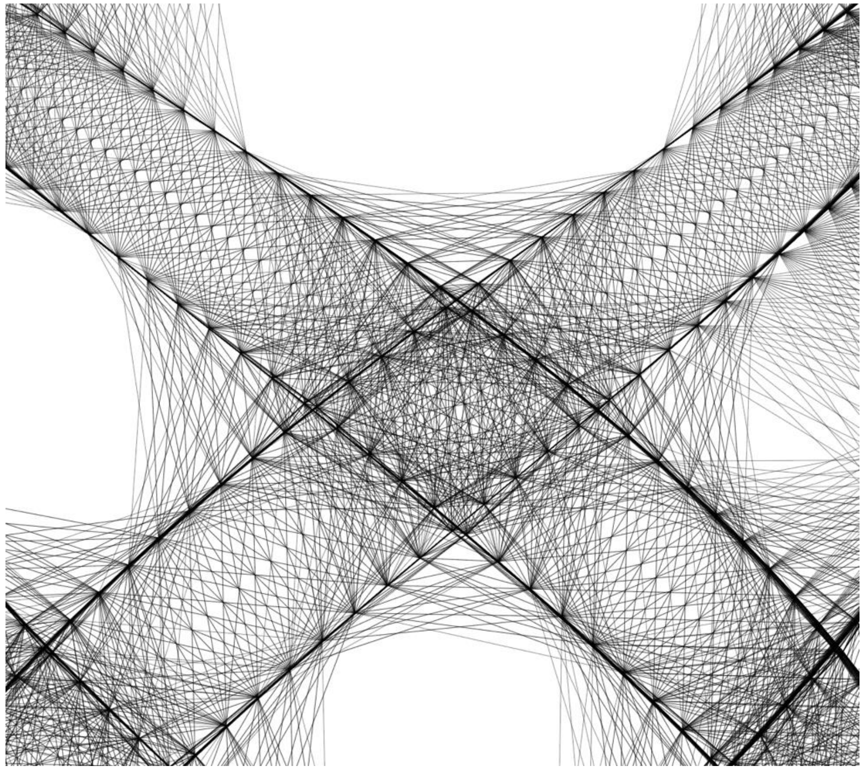


image file

In principle this is the same as vectors in design software where you can keep zooming in



vector

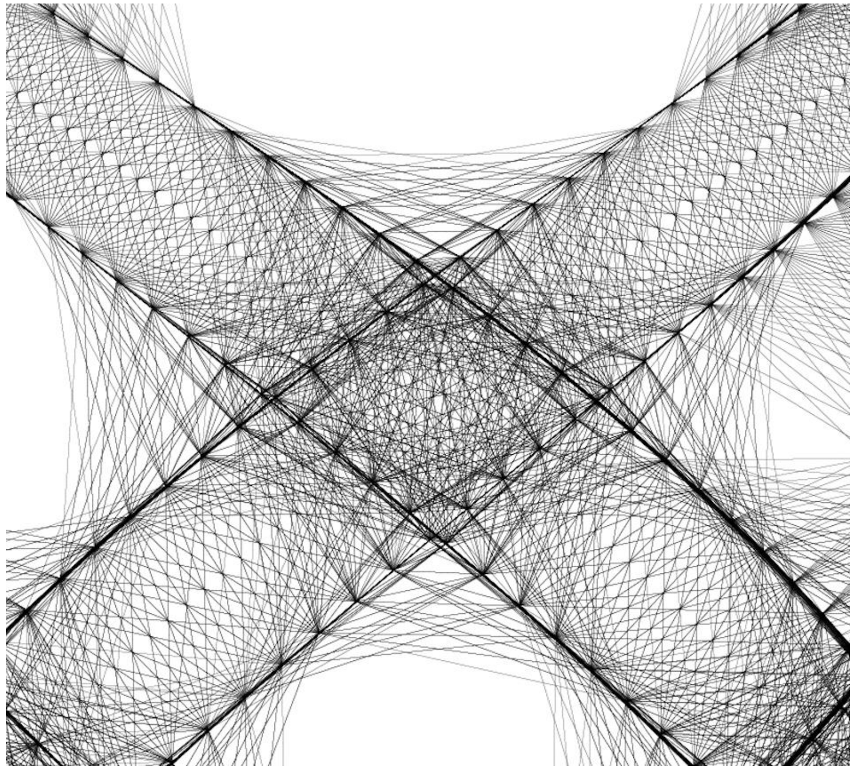
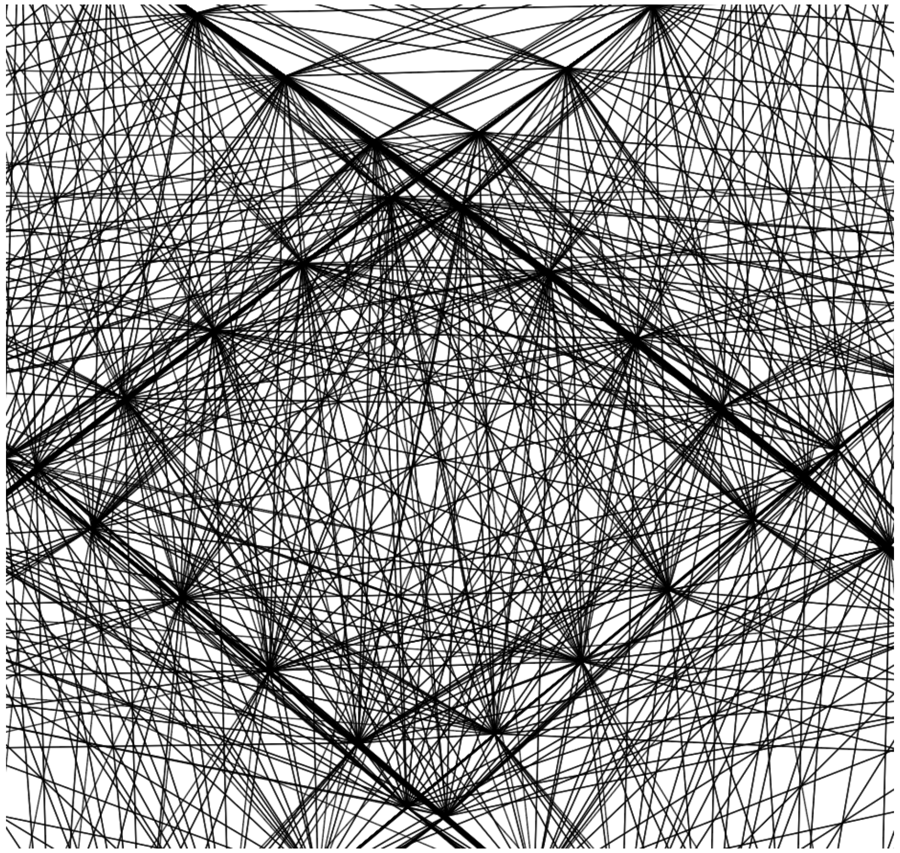


image file

In principle this is the same as vectors in design software where you can keep zooming in



vector

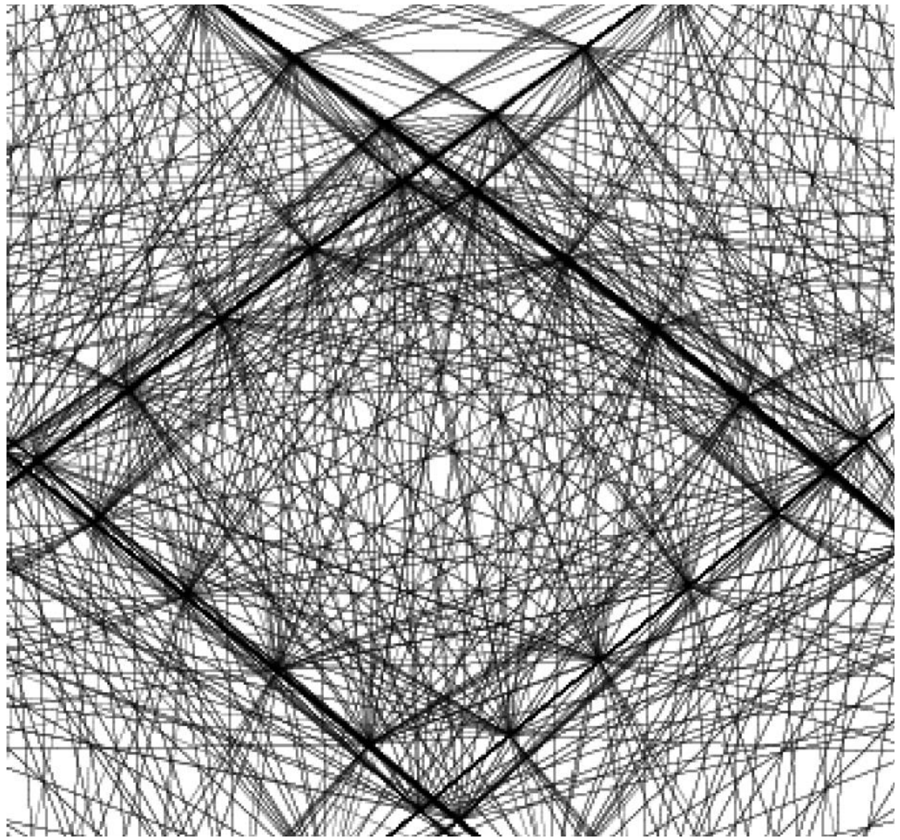
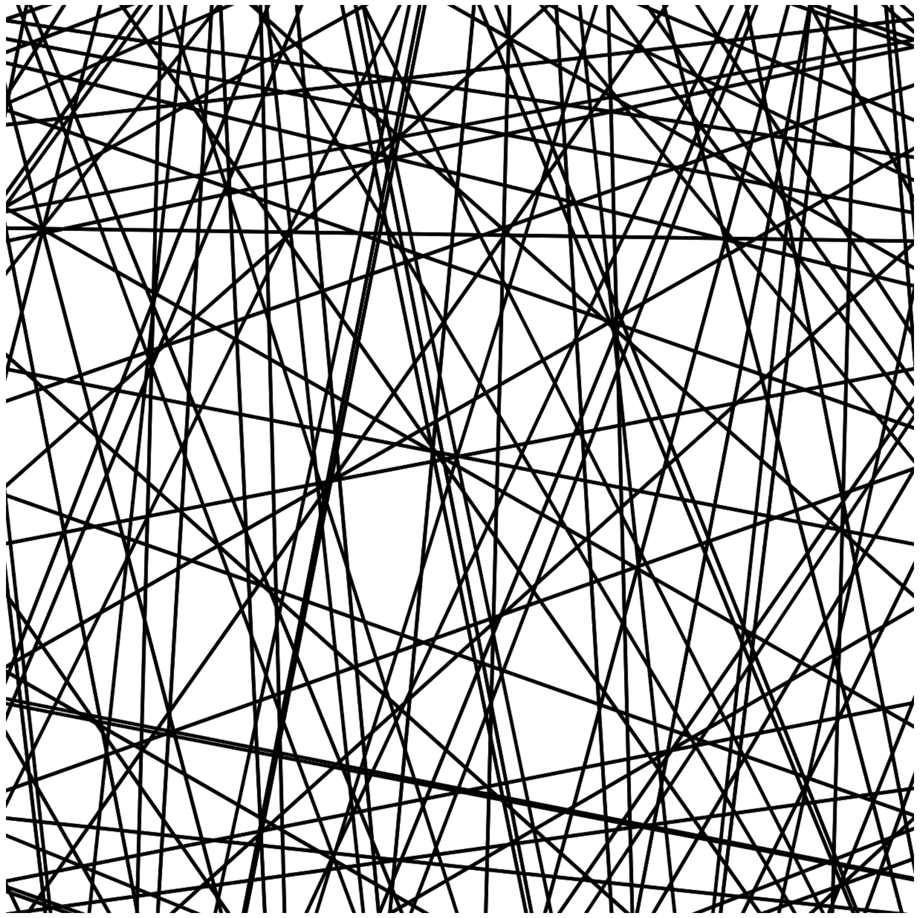


image file

In principle this is the same as vectors in design software where you can keep zooming in



vector

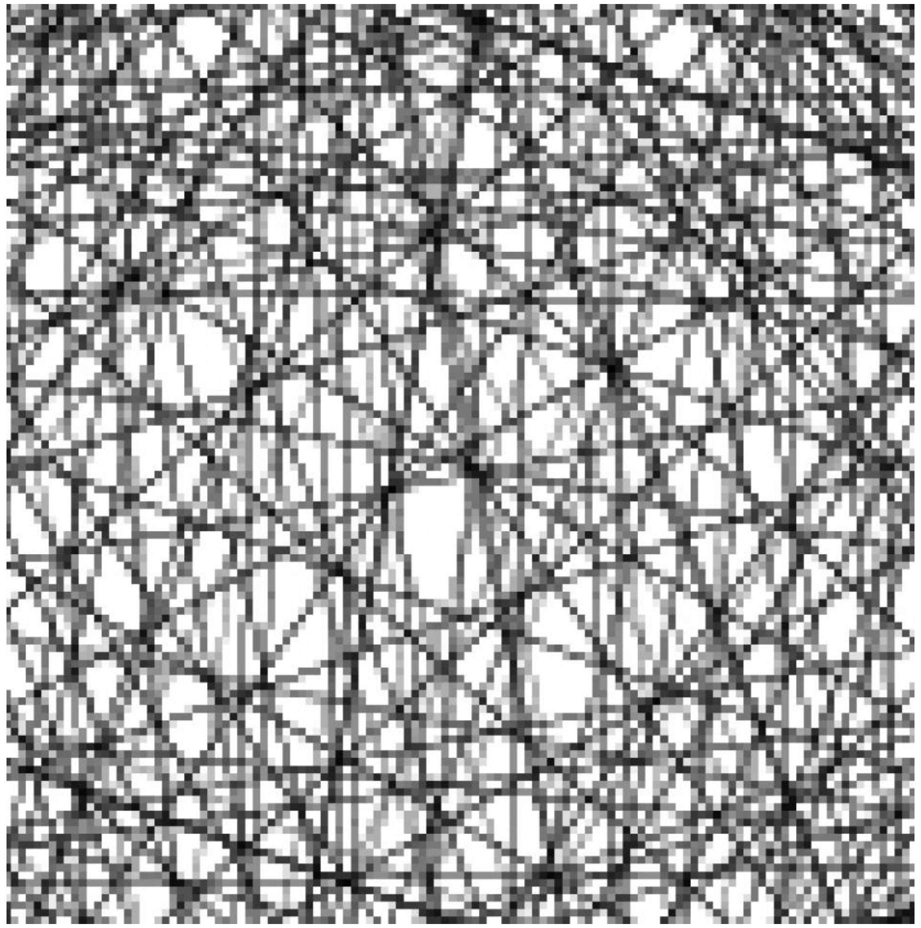
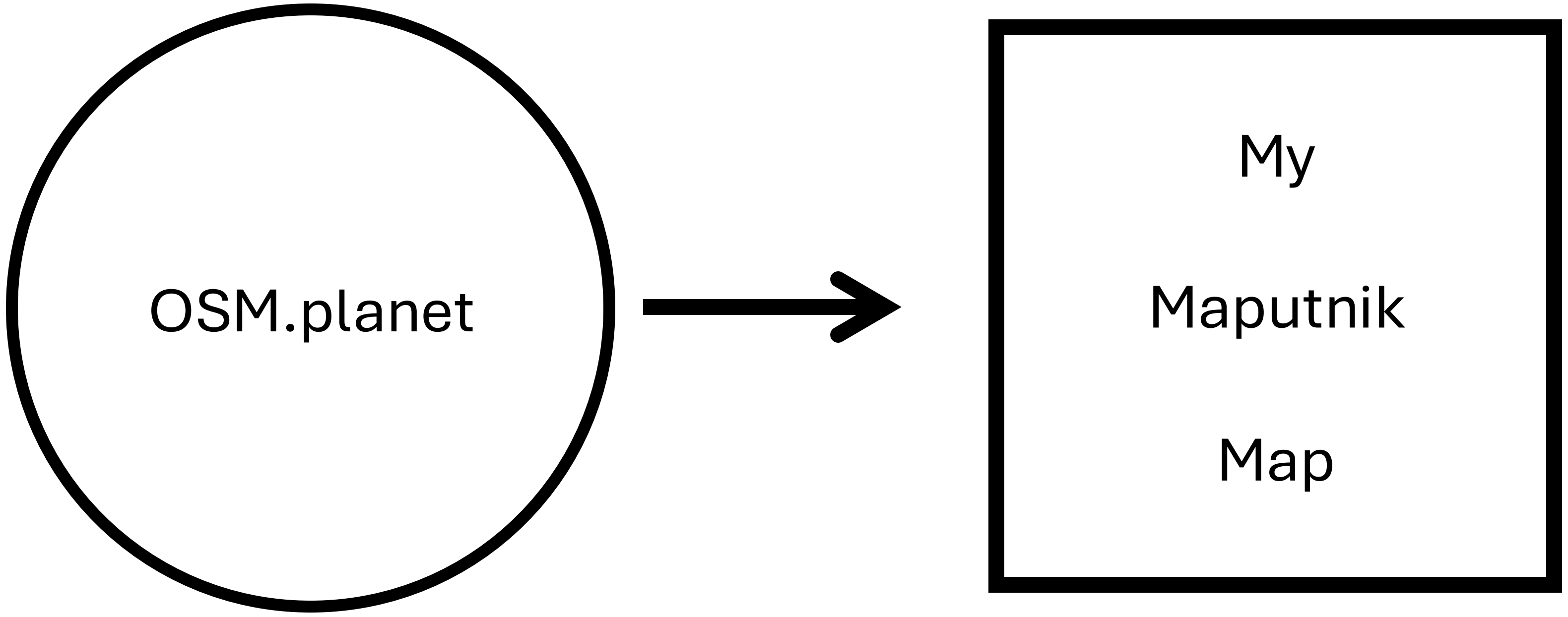


image file

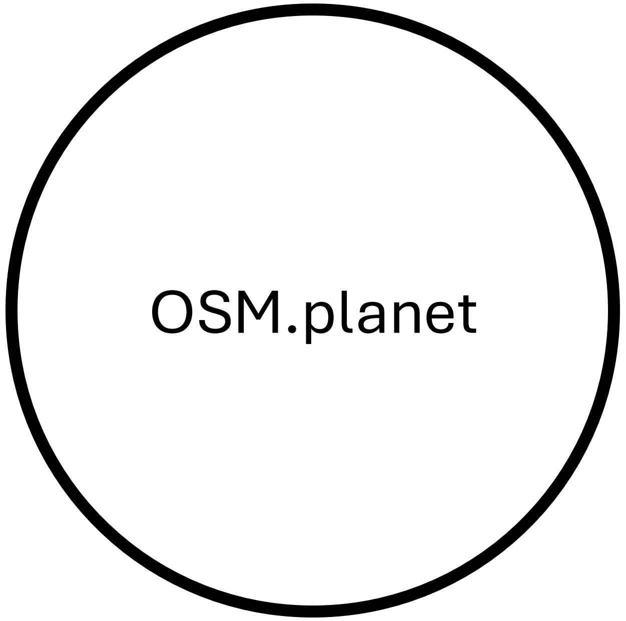
But what is **Maputnik?**

This means it is easier to edit and customize or maps by directly accessing the data we want from OSM and telling the browser how we want it to be displayed



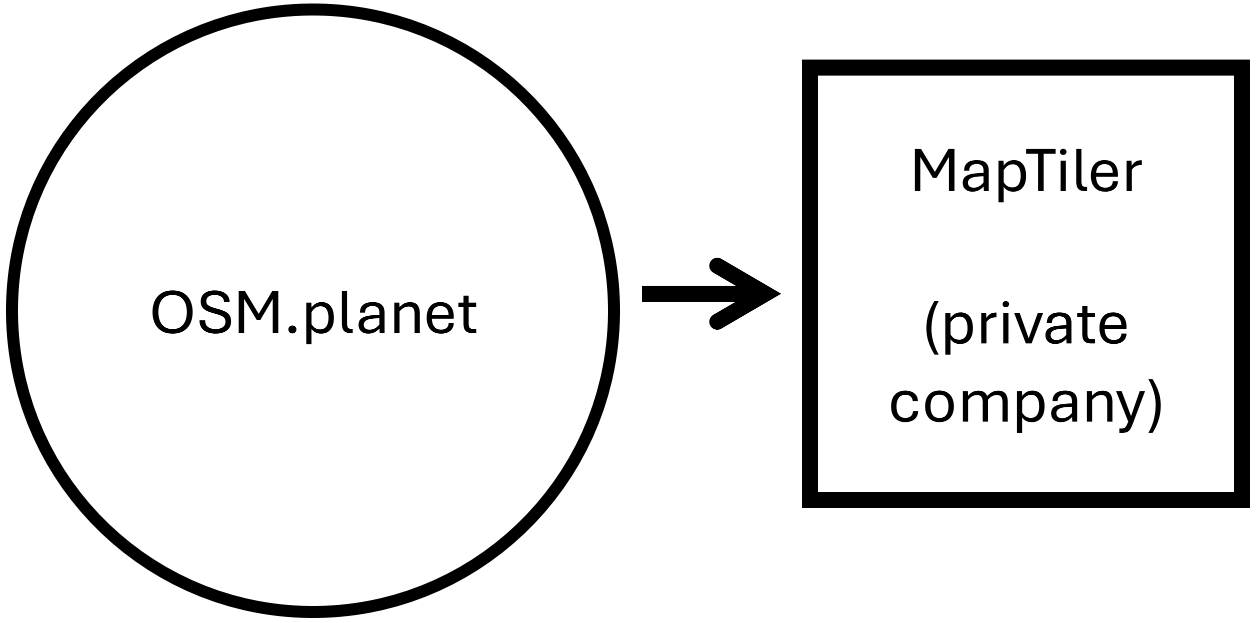
But what is **Maputnik?**

Actually it’s a little bit more complicated than that …



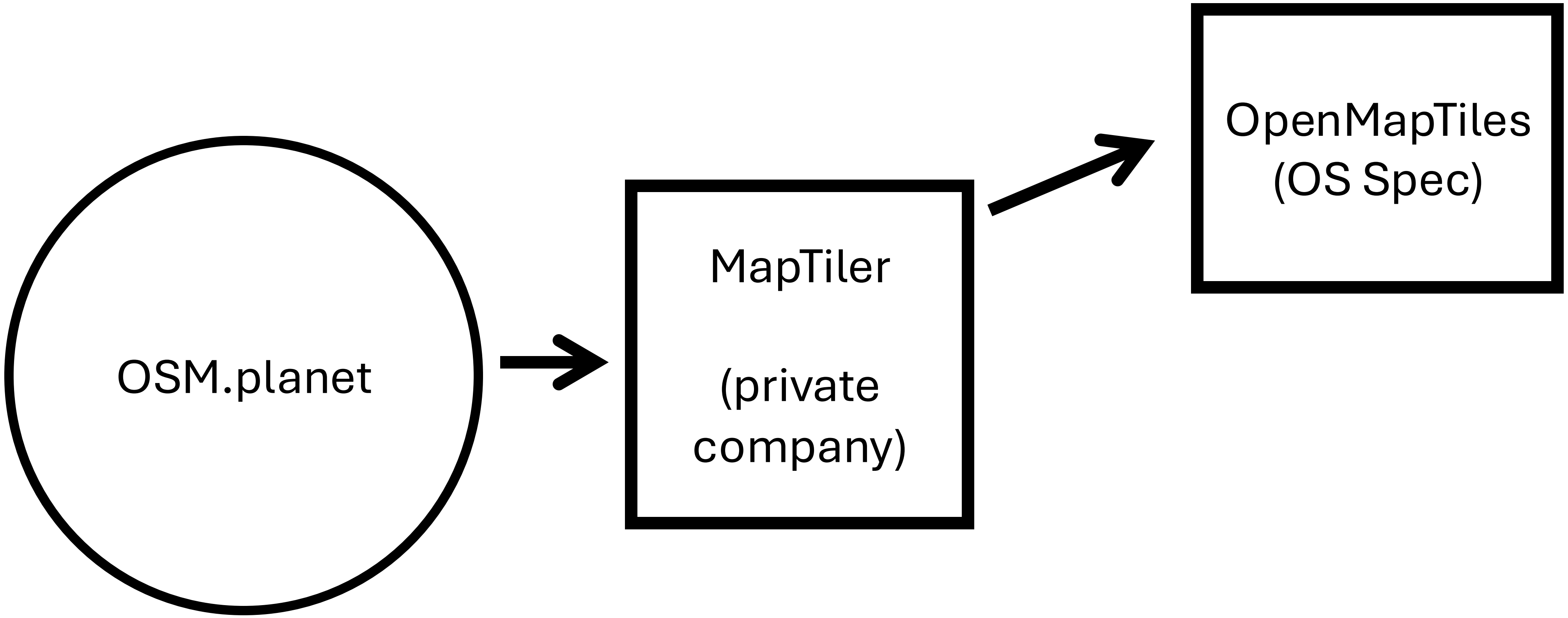
But what is **Maputnik?**

Actually it’s a little bit more complicated than that ...



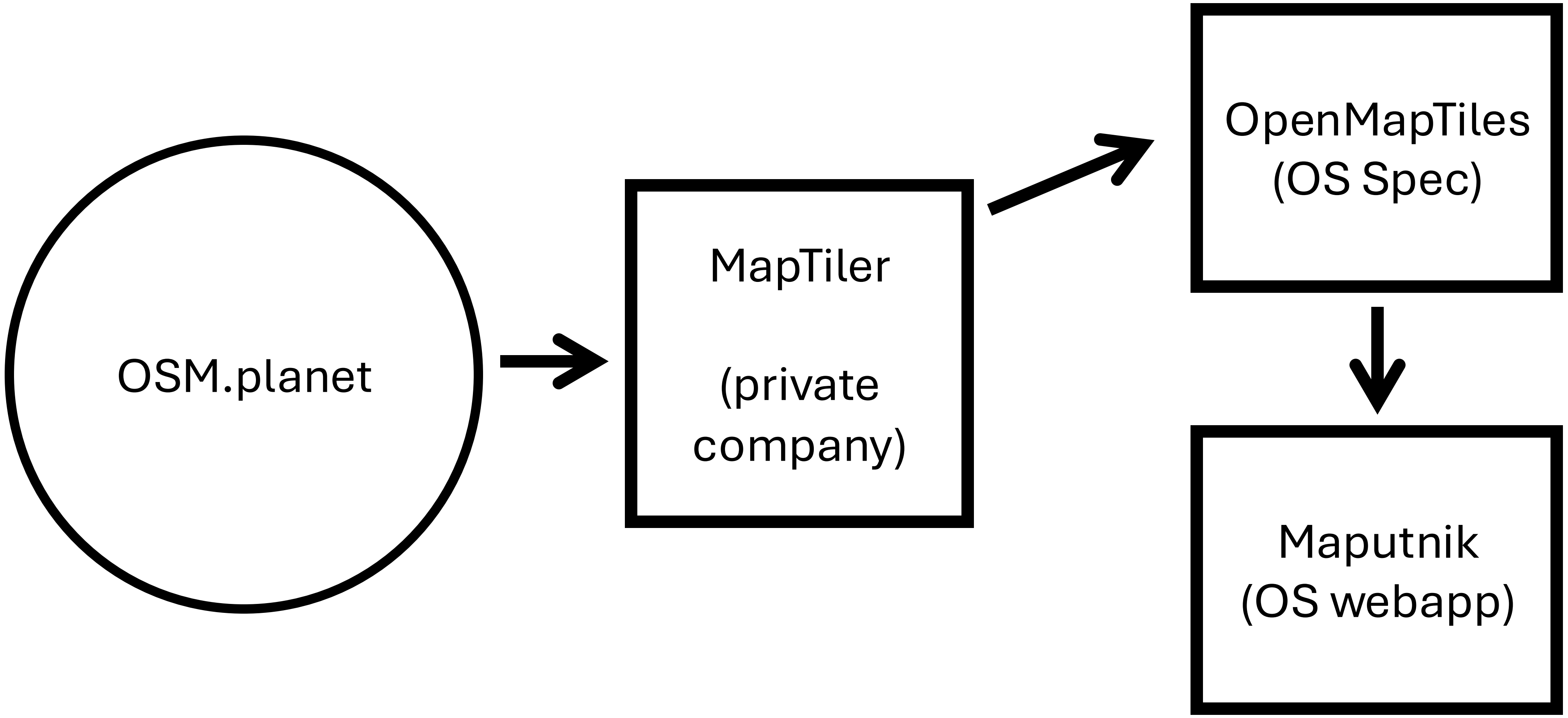
But what is **Maputnik?**

Actually it’s a little bit more complicated than that ...

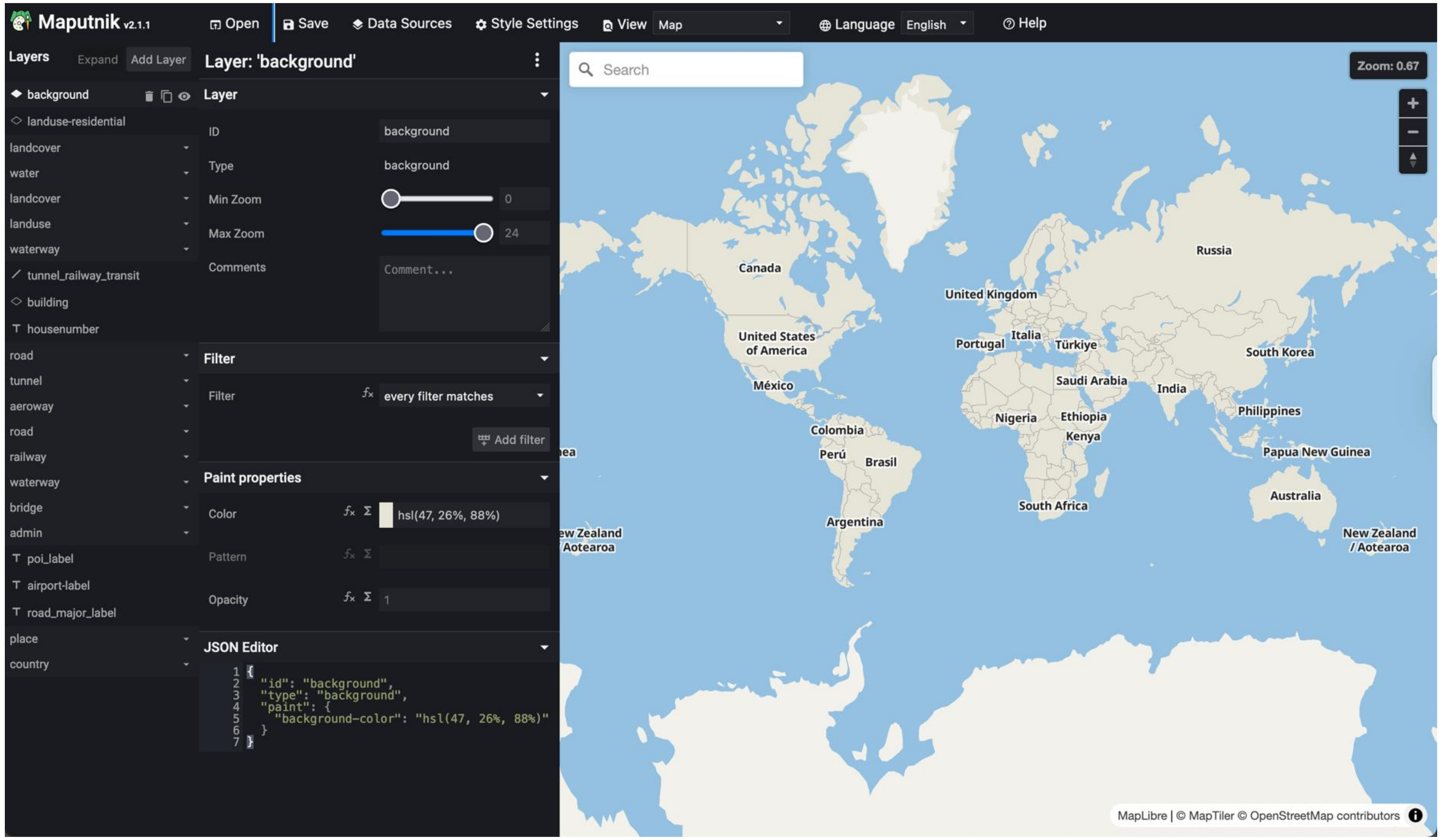


But what is **Maputnik?**

Actually it’s a little bit more complicated than that …



Enough! Lets make a map



Layer: 'background'

Layer

ID	background
Type	background
Min Zoom	0
Max Zoom	24
Comments	Comment...

Filter

Filter	every filter matches

Add filter

Paint properties

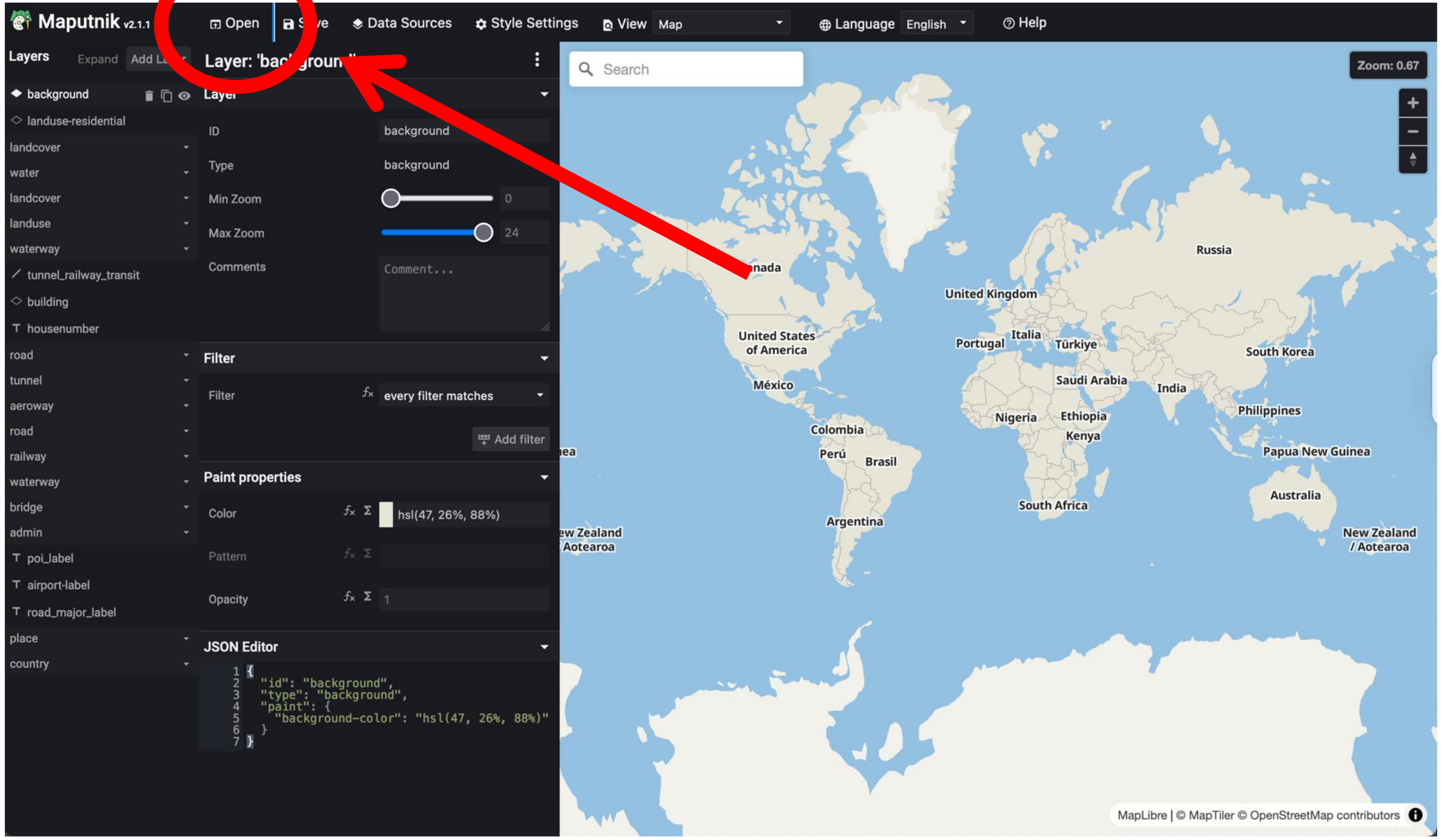
Color	hsl(47, 26%, 88%)
Pattern	
Opacity	1

JSON Editor

```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "hsl(47, 26%, 88%)"
  }
}
```

https://maplibre.org/maputnik/

Lets make a **NEW** map

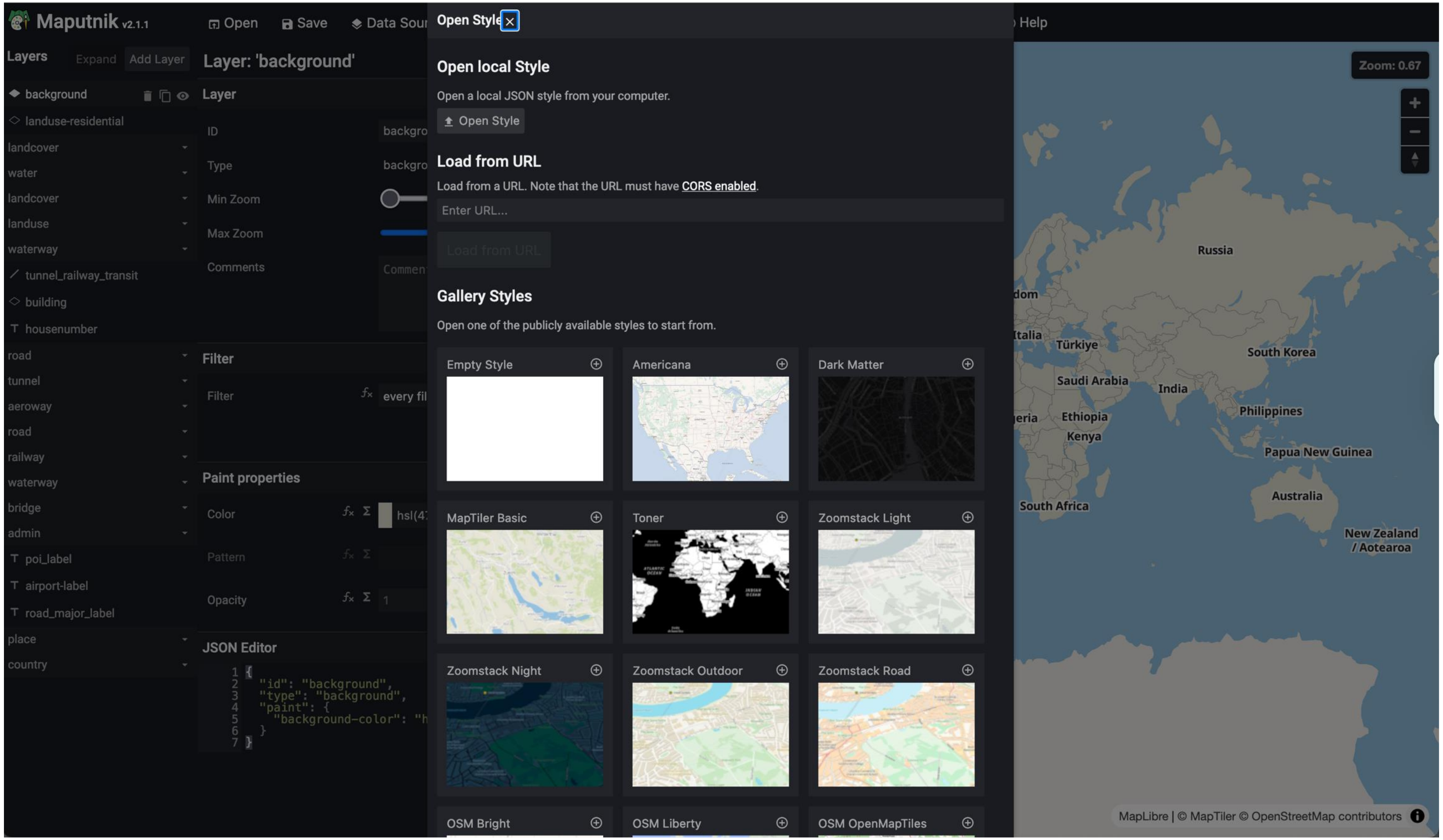


Layer: 'background'

ID	background
Type	background
Min Zoom	0
Max Zoom	24
Comments	Comment...
Filter	every filter matches
Color	hsl(47, 26%, 88%)
Pattern	
Opacity	1

```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "hsl(47, 26%, 88%)"
  }
}
```

https://maplibre.org/maputnik/



Open Style

Open local Style

Open a local JSON style from your computer.

Open Style

Load from URL

Load from a URL. Note that the URL must have CORS enabled.

Enter URL...

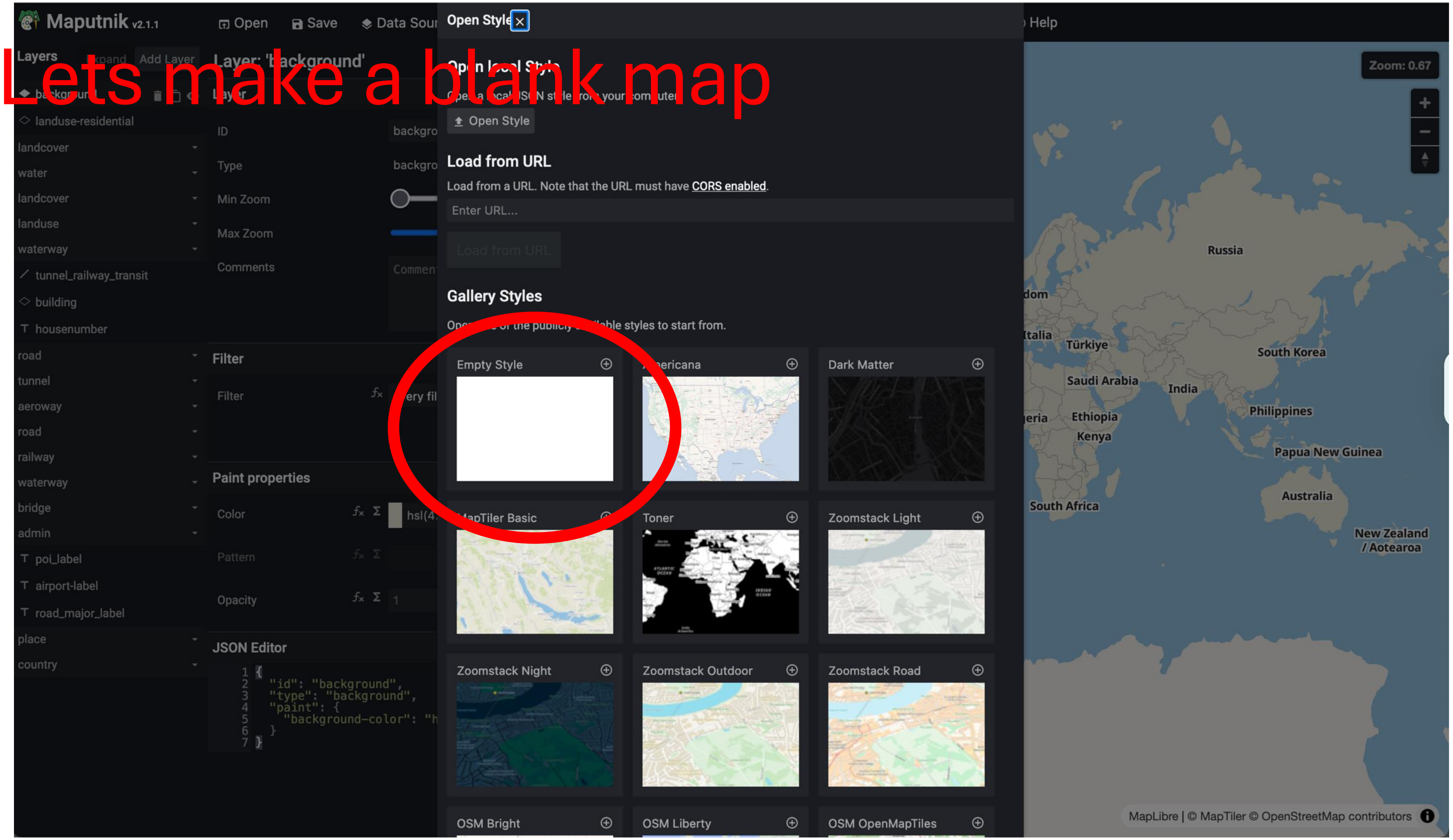
Load from URL

Gallery Styles

Open one of the publicly available styles to start from.

```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "h
  }
}
```

Lets make a blank map



Open Style

Open local Style

Open a local JSON style from your computer.

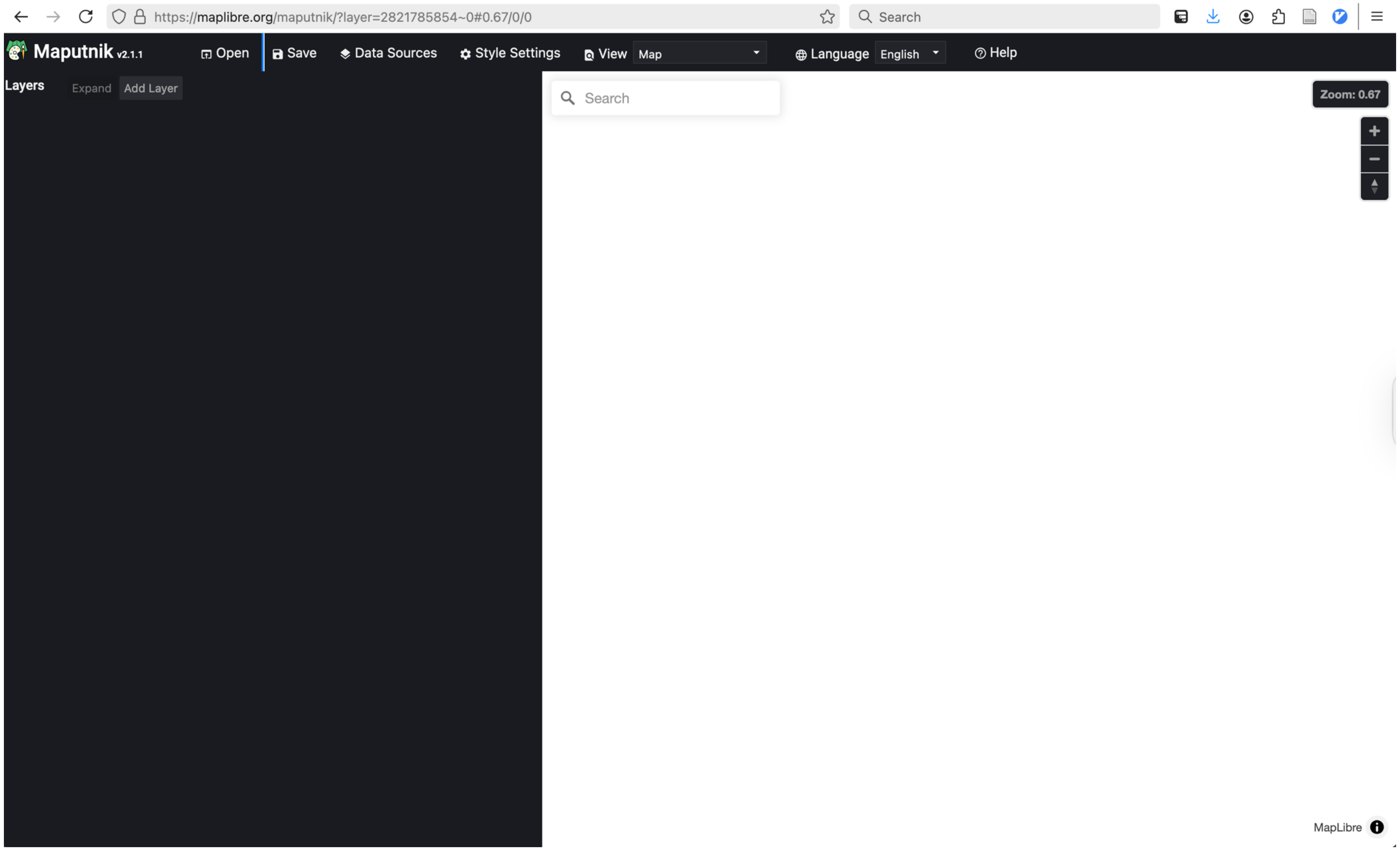
Load from URL

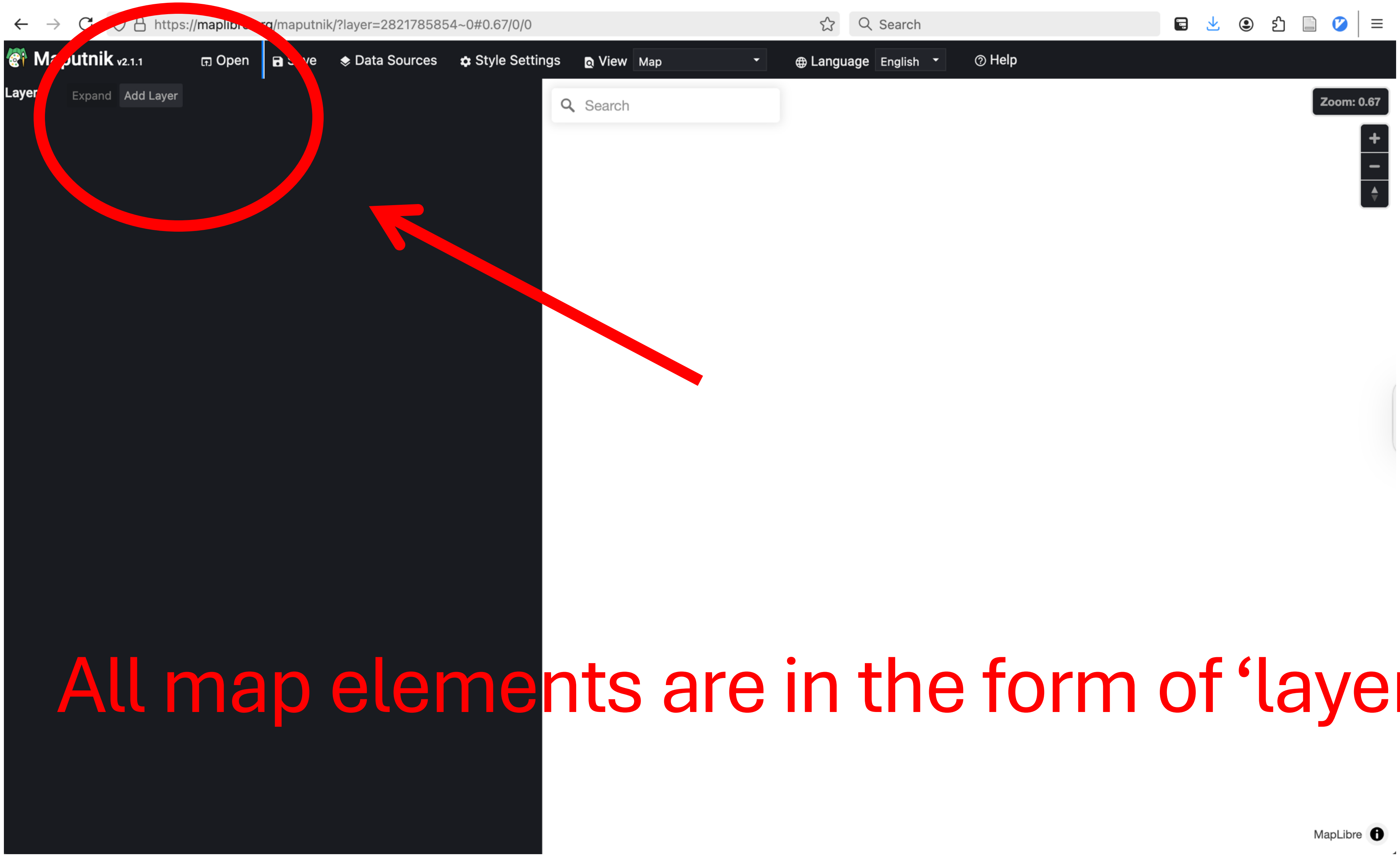
Load from a URL. Note that the URL must have CORS enabled.

Gallery Styles

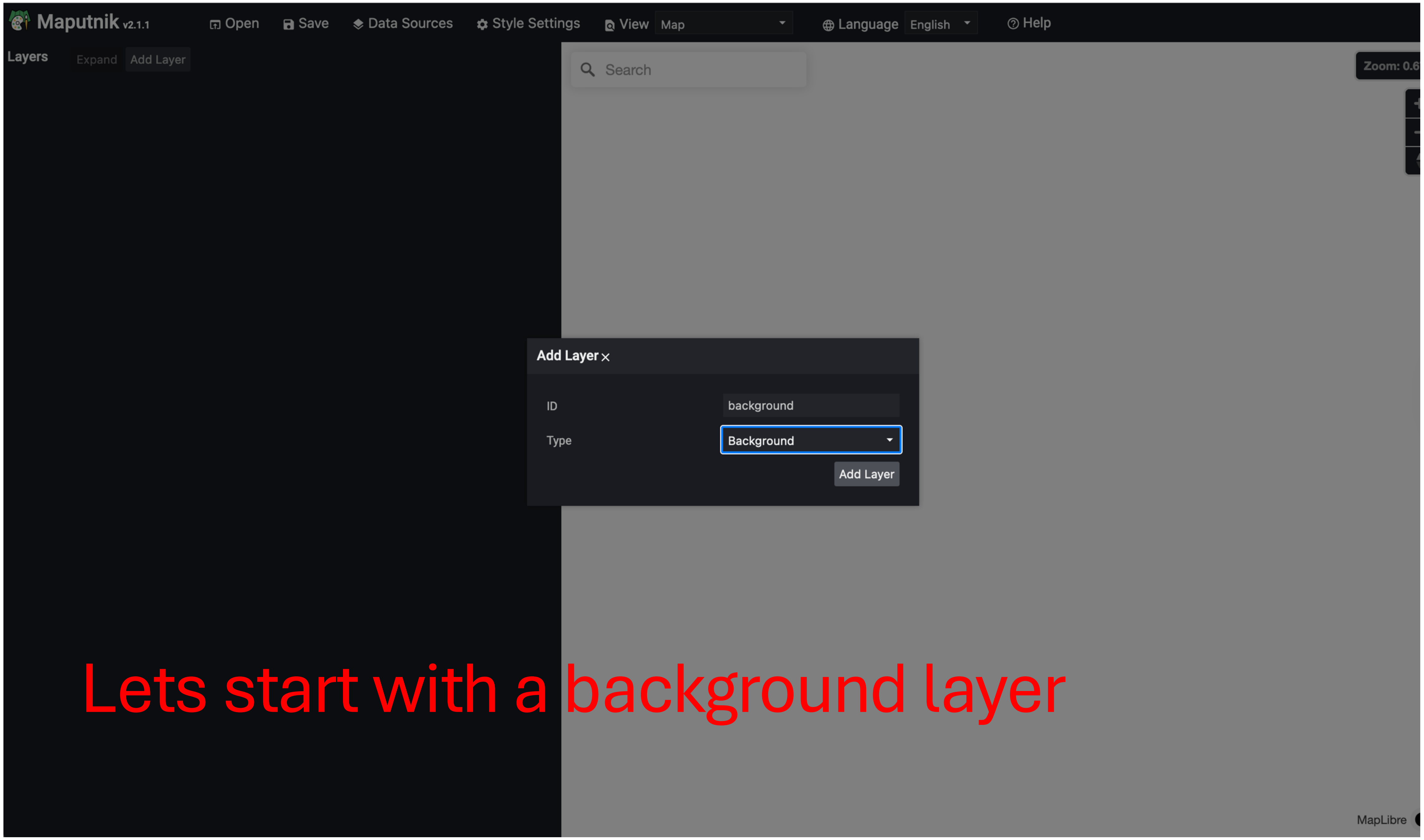
Open one of the publicly available styles to start from.

```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "h
  }
}
```


https://maplibre.org/maputnik/?layer=2821785854~0#0.67/0/0
Search
Maputnik v2.1.1
Open
Save
Data Sources
Style Settings
View
Map
Language
English
Help
Layers
Expand
Add Layer
Search
Zoom: 0.67
MapLibre



All map elements are in the form of 'layers'

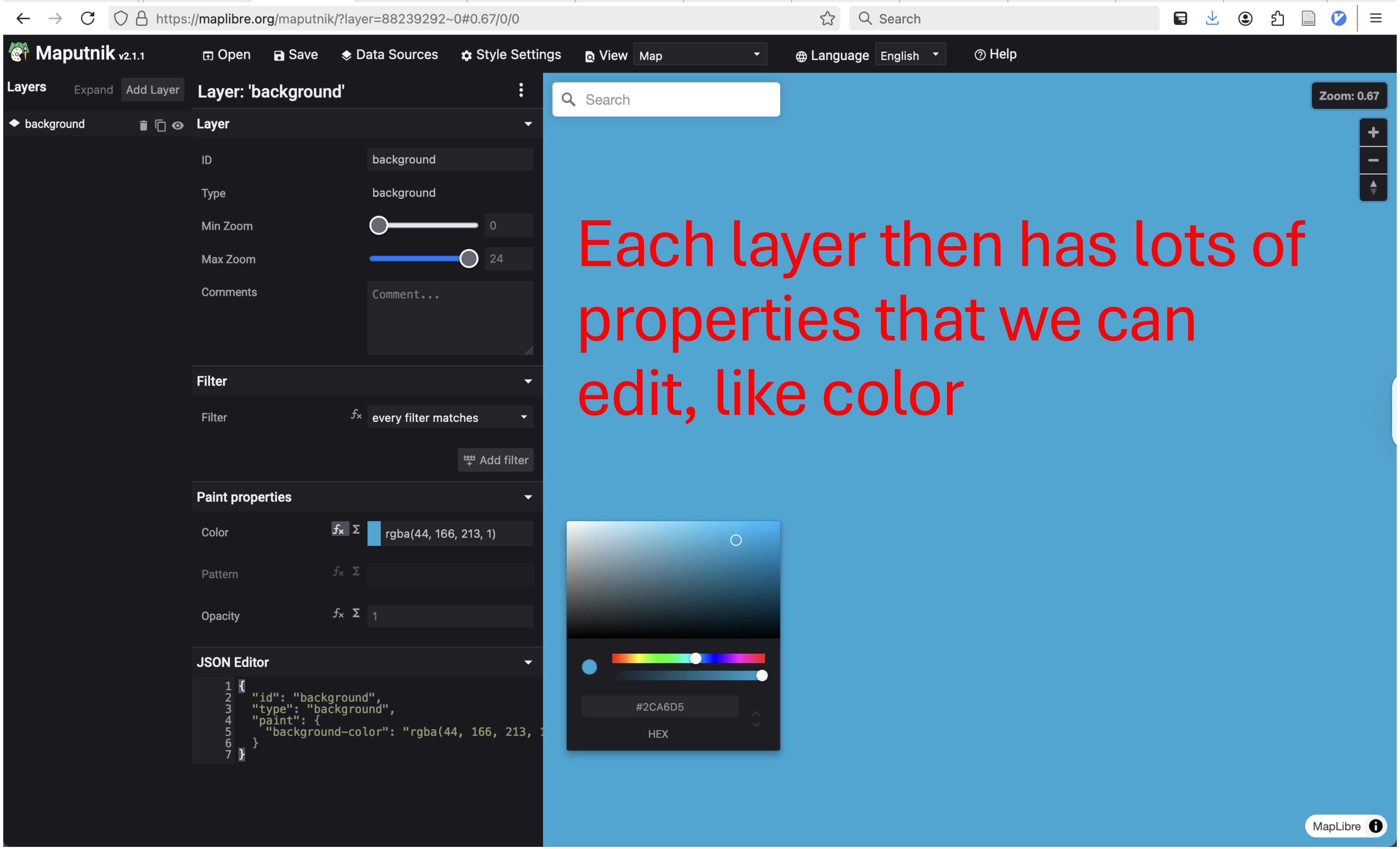


Add Layer

ID	background
Type	Background

Add Layer

Lets start with a background layer



Each layer then has lots of properties that we can edit, like color

Layer: 'background'

Layer

ID: background
Type: background
Min Zoom: 0
Max Zoom: 24
Comments: Comment...

Filter

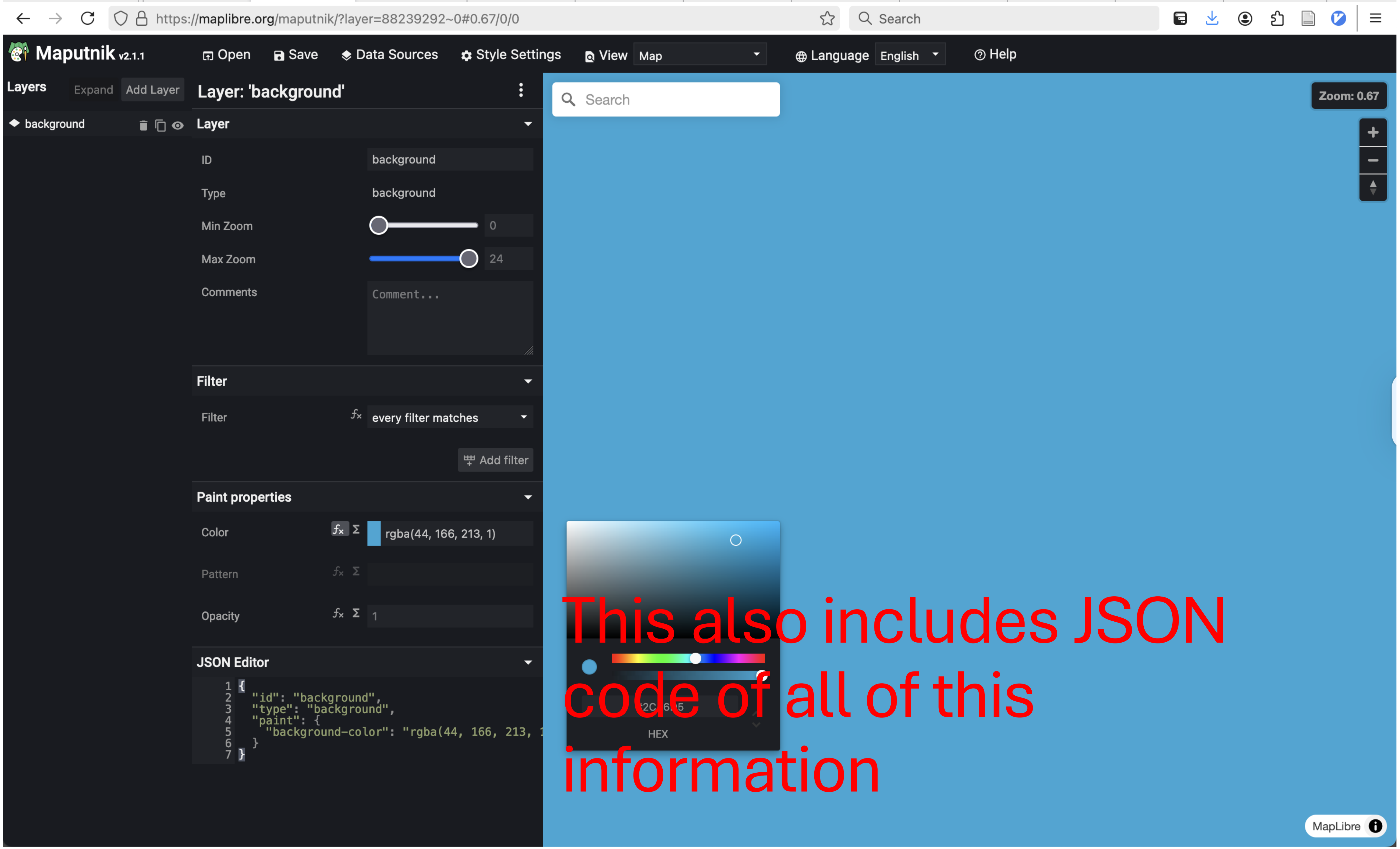
Filter: every filter matches
Add filter

Paint properties

Color: rgba(44, 166, 213, 1)
Pattern:
Opacity: 1

JSON Editor

```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "rgba(44, 166, 213, 1
  }
}
```



Layer: 'background'

Layer

ID: background
Type: background
Min Zoom: 0
Max Zoom: 24
Comments: Comment...

Filter

Filter: every filter matches
Add filter

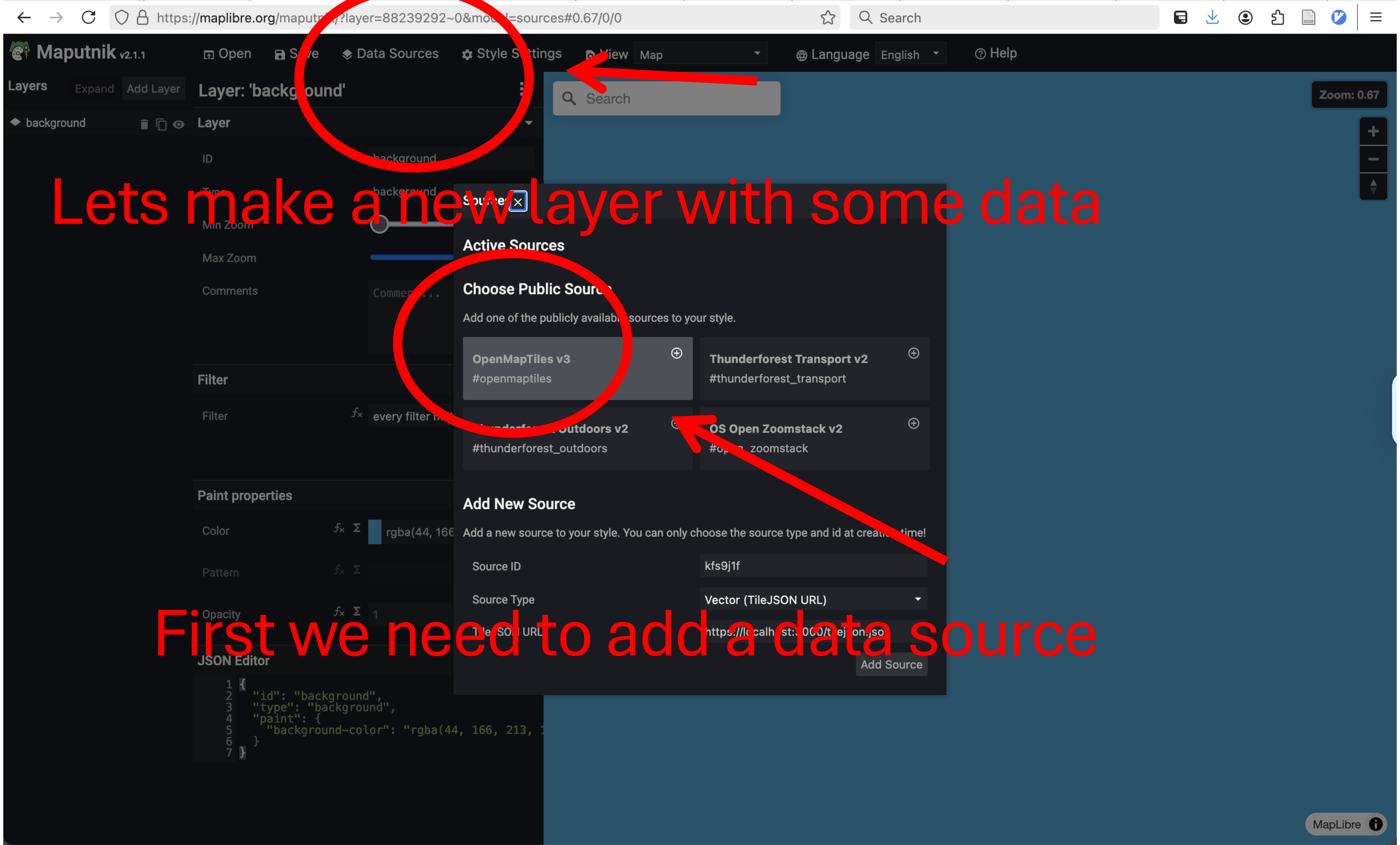
Paint properties

Color: rgba(44, 166, 213, 1)
Pattern:
Opacity: 1

JSON Editor

```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "rgba(44, 166, 213, 1
  }
}
```

This also includes JSON code of all of this information



Lets make a new layer with some data

Sources

Active Sources

Choose Public Source

Add one of the publicly available sources to your style.

OpenMapTiles v3 #openmaptiles

Thunderforest Transport v2 #thunderforest_transport

Thunderforest Outdoors v2 #thunderforest_outdoors

OS Open Zoomstack v2 #open_zoomstack

Add New Source

Add a new source to your style. You can only choose the source type and id at creation time!

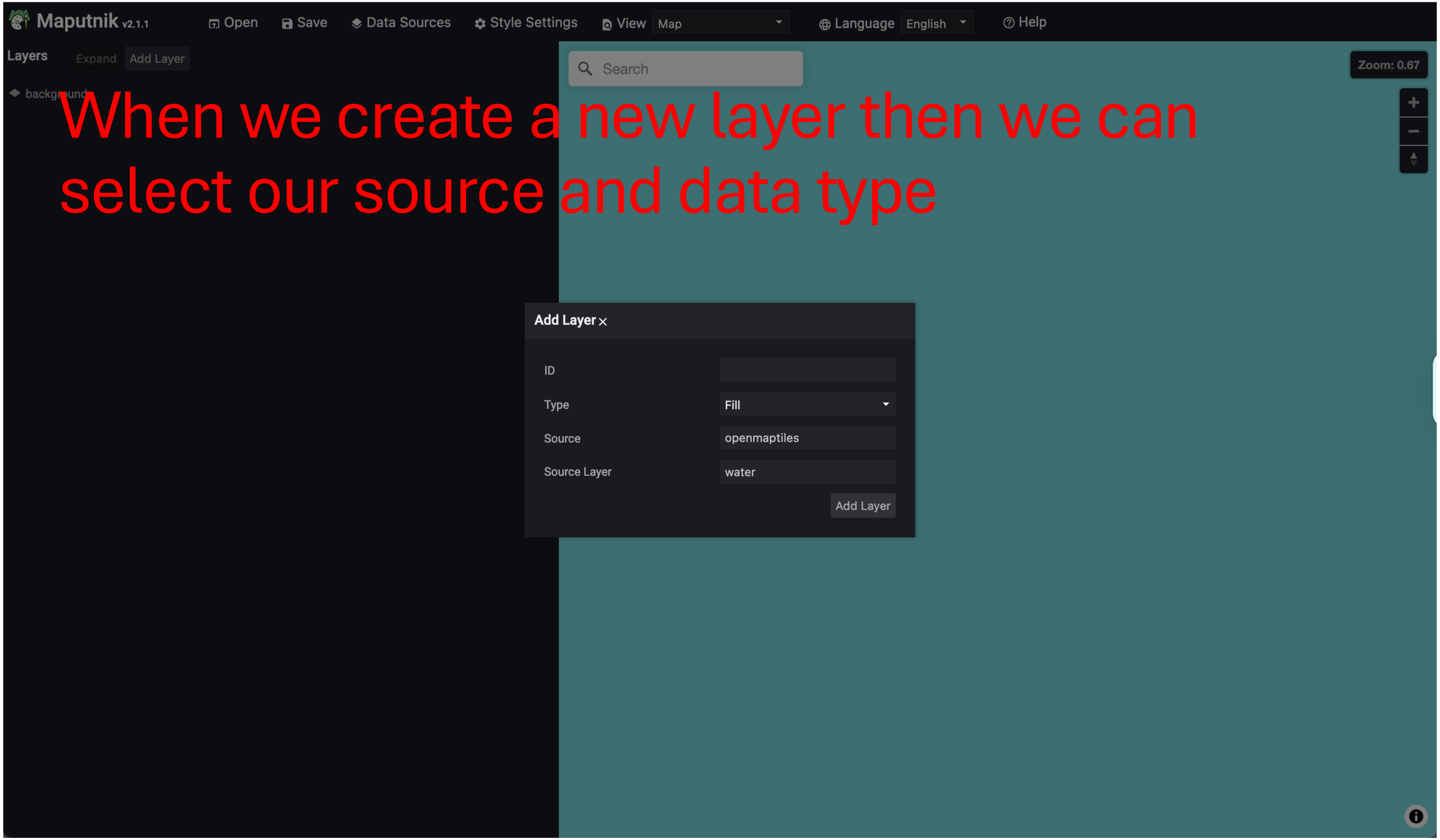
Source ID	kfs9j1f
Source Type	Vector (TileJSON URL)
TileJSON URL	https://localhost:3000/tilejson.json

Add Source

ID	background
Type	background
Min Zoom	
Max Zoom	
Comments	Comment...
Filter	every filter m
Color	rgba(44, 166
Pattern	
Opacity	1

```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "rgba(44, 166, 213, 1
  }
}
```

First we need to add a data source

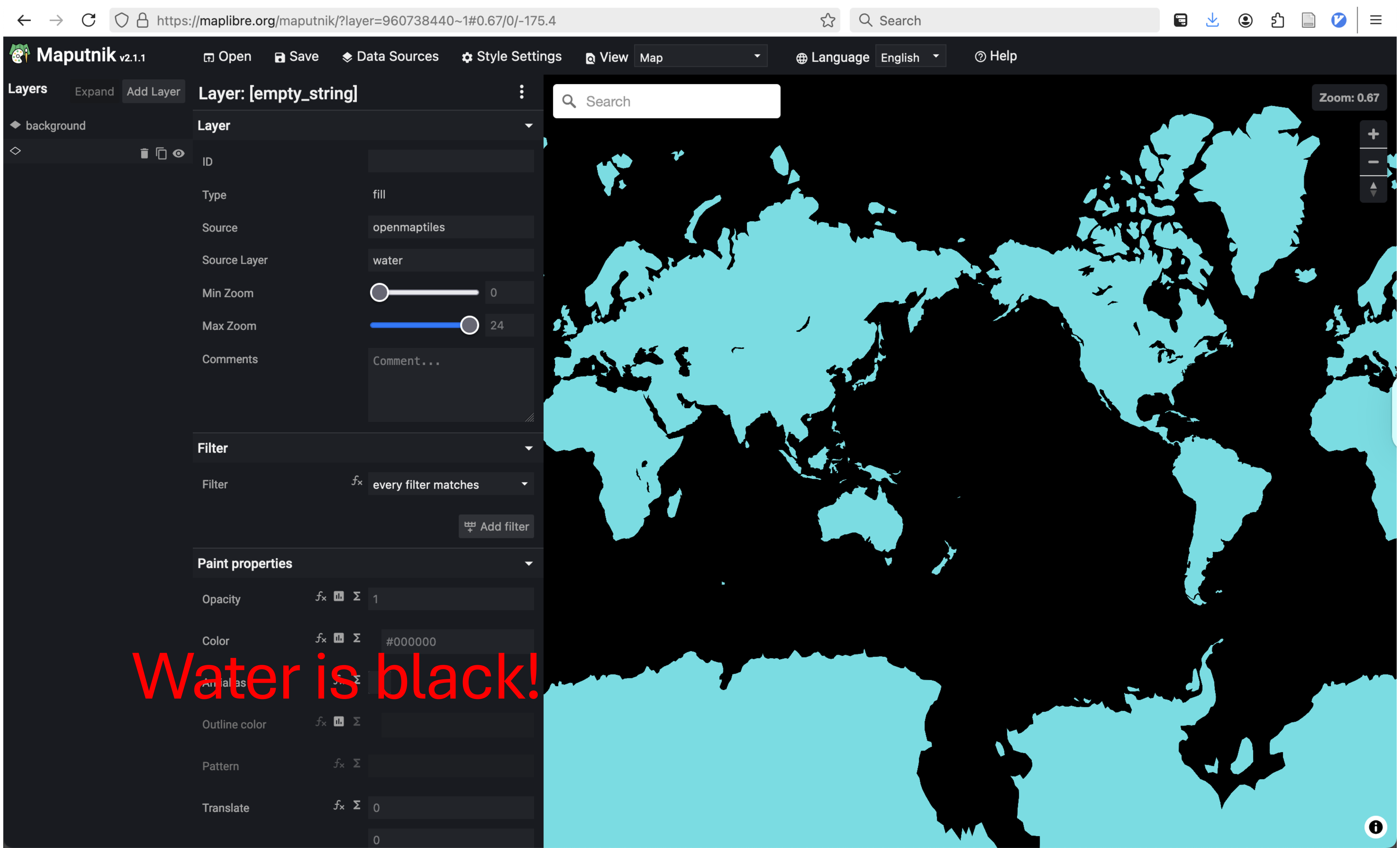


When we create a new layer then we can select our source and data type

Add Layer

ID	
Type	Fill
Source	openmaptiles
Source Layer	water

Add Layer


https://maplibre.org/maputnik/?layer=960738440~1#0.67/0/-175.4

Maputnik v2.1.1 | Open | Save | Data Sources | Style Settings | View: Map | Language: English | Help

Layers | Expand | Add Layer
- background
-

Layer: [empty_string]

Layer

ID	
Type	fill
Source	openmaptiles
Source Layer	water
Min Zoom	0
Max Zoom	24
Comments	Comment...

Filter

Filter	every filter matches

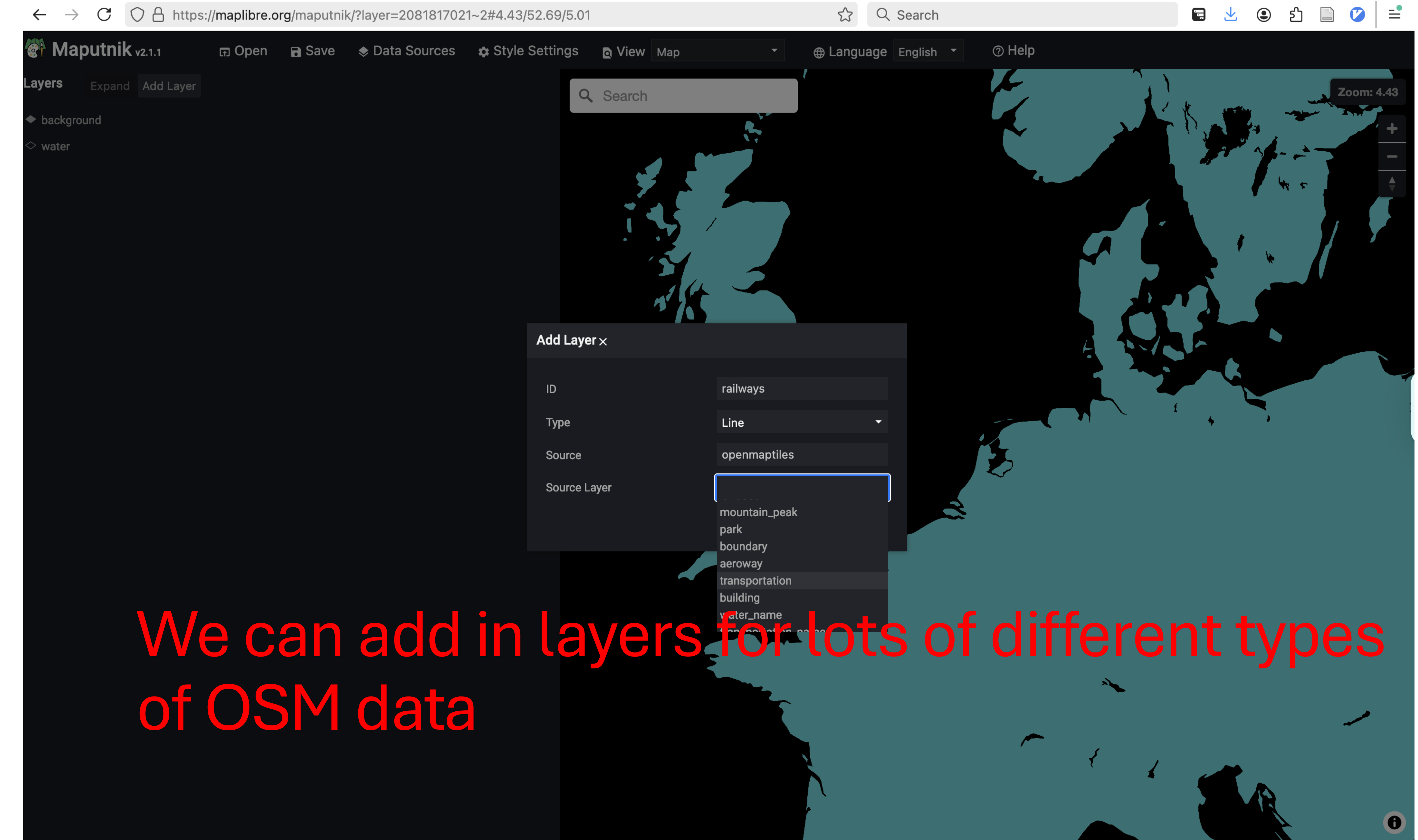
Add filter

Paint properties

Opacity	1
Color	#000000
Antialias	
Outline color	
Pattern	
Translate	0
	0

Zoom: 0.67

Water is black!

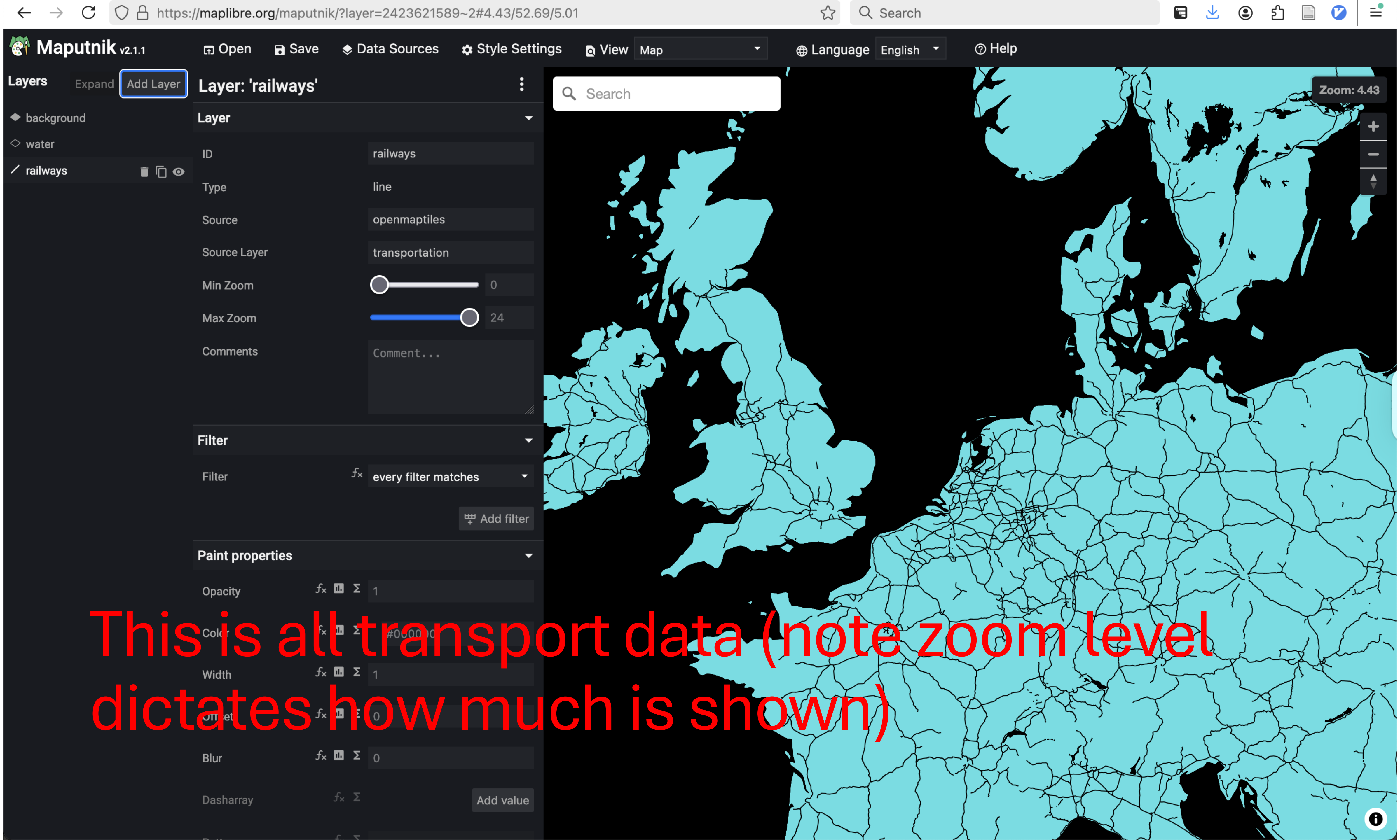


https://maplibre.org/maputnik/?layer=2081817021~2#4.43/52.69/5.01

Add Layer

ID	railways
Type	Line
Source	openmaptiles
Source Layer	

We can add in layers for lots of different types of OSM data



https://maplibre.org/maputnik/?layer=2423621589~2#4.43/52.69/5.01

Maputnik v2.1.1 | Open | Save | Data Sources | Style Settings | View: Map | Language: English | Help

Layers: background, water, railways

Layer: 'railways'

Layer

ID	railways
Type	line
Source	openmaptiles
Source Layer	transportation
Min Zoom	0
Max Zoom	24
Comments	Comment...

Filter

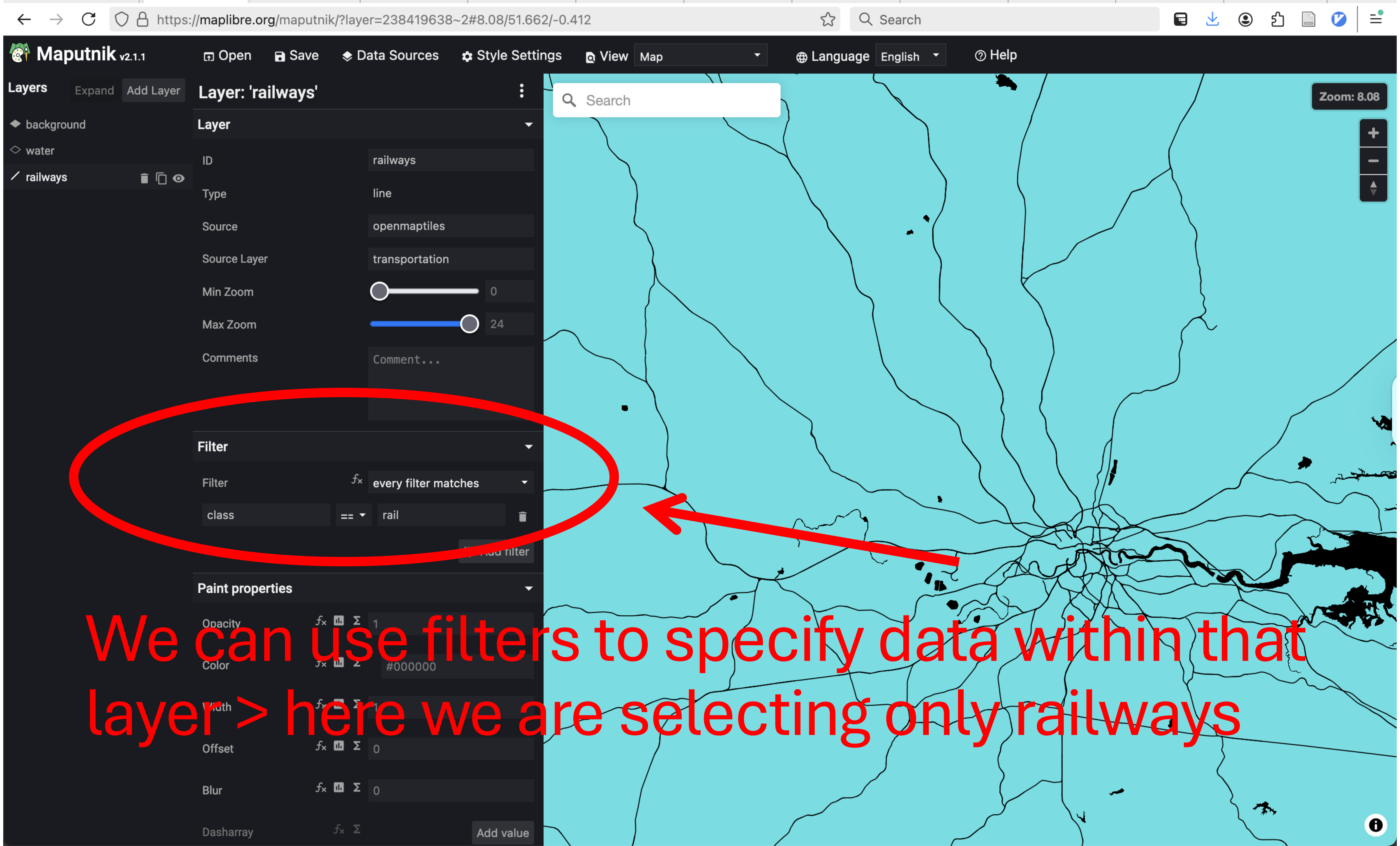
Filter	every filter matches

Paint properties

Opacity	1
Color	#000000
Width	1
Offset	0
Blur	0
Dasharray	Add value

Zoom: 4.43

This is all transport data (note zoom level dictates how much is shown)



https://maplibre.org/maputnik/?layer=238419638~2#8.08/51.662/-0.412

Maputnik v2.1.1 · Open · Save · Data Sources · Style Settings · View: Map · Language: English · Help

Layers: background, water, railways

Layer: 'railways'

ID	railways
Type	line
Source	openmaptiles
Source Layer	transportation
Min Zoom	0
Max Zoom	24
Comments	Comment...

Filter: every filter matches

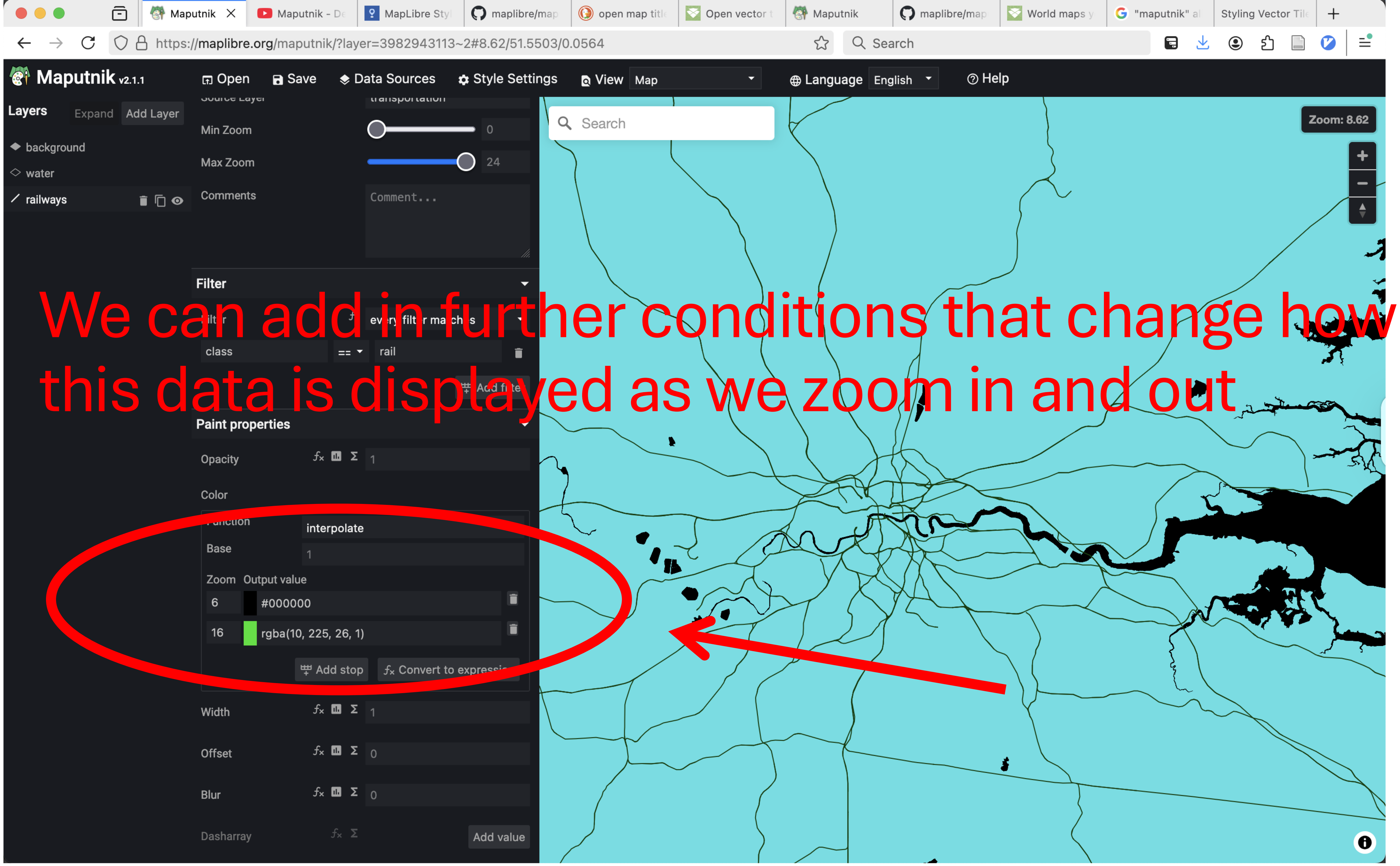
class == rail

Paint properties

Opacity	1
Color	#000000
Width	1
Offset	0
Blur	0
Dasharray	Add value

Zoom: 8.08

We can use filters to specify data within that layer > here we are selecting only railways



https://maplibre.org/maputnik/?layer=3982943113~2#8.62/51.5503/0.0564

Min Zoom 0
Max Zoom 24
Comments
Comment...

Filter
every filter matches
class == rail
Add filter

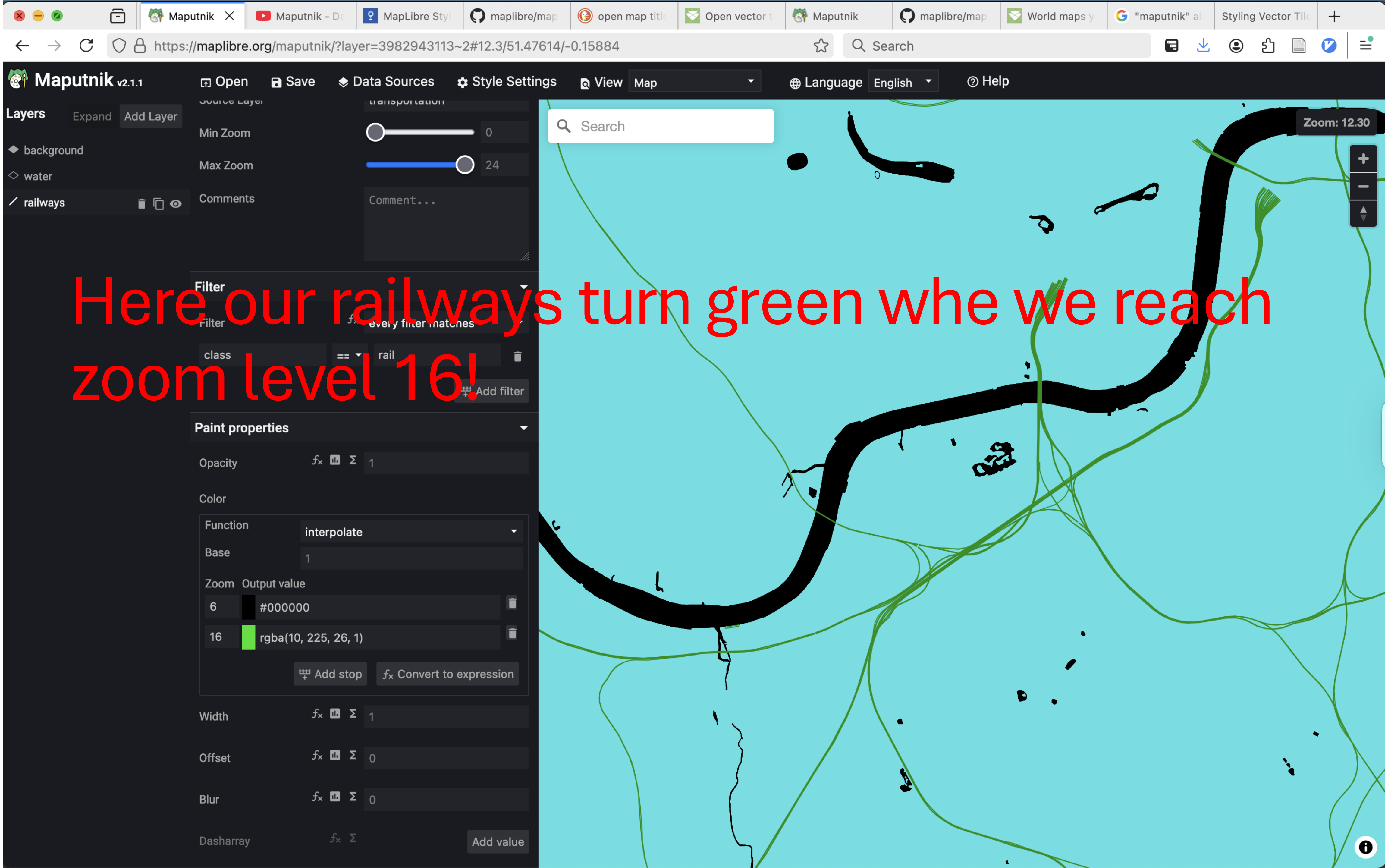
Paint properties
Opacity 1
Color
Function interpolate
Base 1

Zoom	Output value
6	#000000
16	rgba(10, 225, 26, 1)

Add stop
Convert to expression

Width 1
Offset 0
Blur 0
Dasharray
Add value

We can add in further conditions that change how this data is displayed as we zoom in and out



Here our railways turn green whe we reach zoom level 16!

https://maplibre.org/maputnik/?layer=3982943113~2#12.3/51.47614/-0.15884

Layers: background, water, railways

Min Zoom: 0
Max Zoom: 24
Comments: Comment...

Filter: every filter matches
class == rail

Paint properties

Opacity: 1

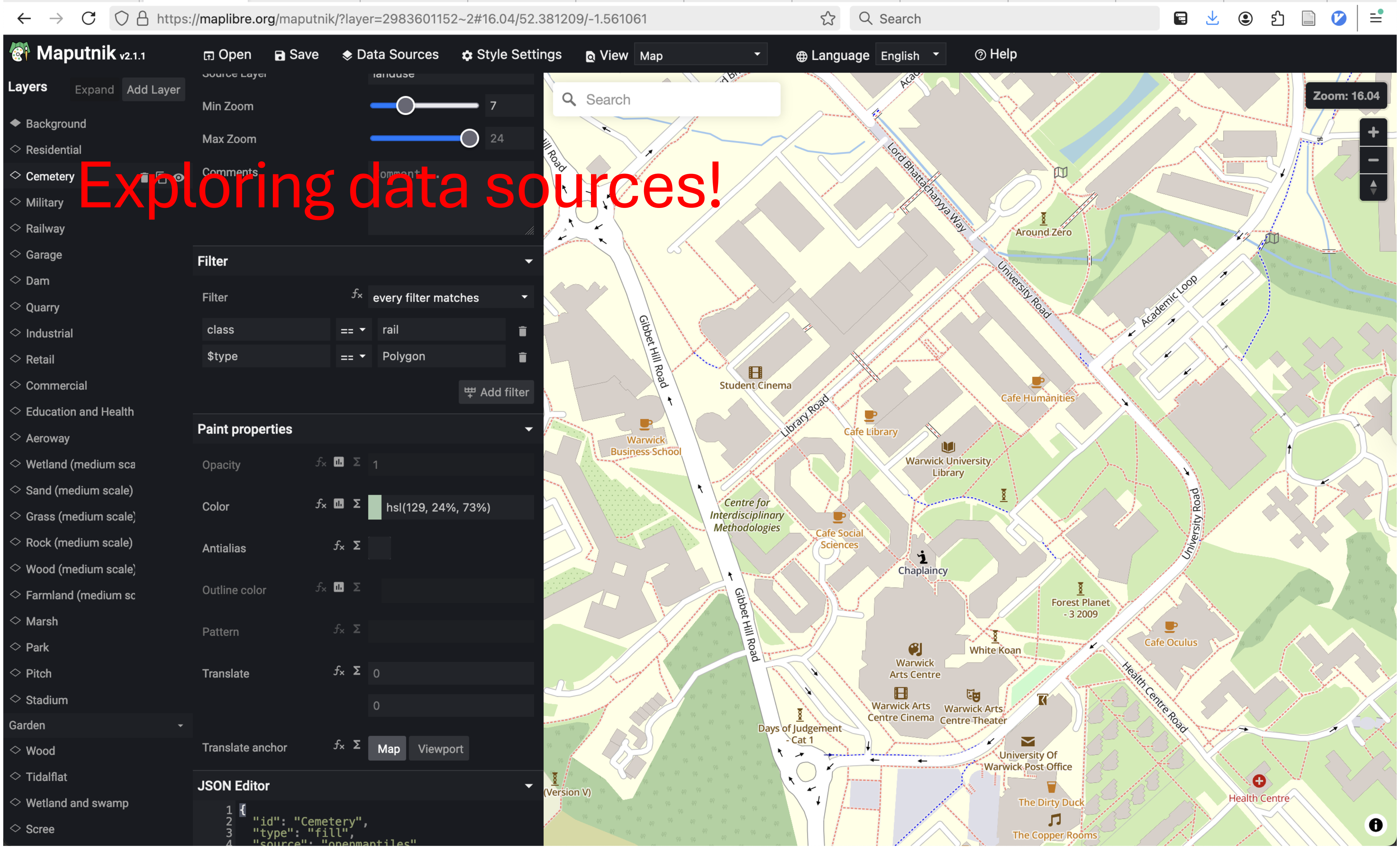
Color
Function: interpolate
Base: 1

Zoom	Output value
6	#000000
16	rgba(10, 225, 26, 1)

Width: 1
Offset: 0
Blur: 0
Dasharray

Zoom: 12.30

Exploring data sources!



https://maplibre.org/maputnik/?layer=2983601152~2#16.04/52.381209/-1.561061

Maputnik v2.1.1 — Open | Save | Data Sources | Style Settings | View: Map | Language: English | Help

Layers — Expand | Add Layer

Background, Residential, Cemetery, Military, Railway, Garage, Dam, Quarry, Industrial, Retail, Commercial, Education and Health, Aeroway, Wetland (medium sca, Sand (medium scale), Grass (medium scale), Rock (medium scale), Wood (medium scale), Farmland (medium sc, Marsh, Park, Pitch, Stadium, Garden: Wood, Tidalflat, Wetland and swamp, Scree

Source Layer: landuse
Min Zoom: 7
Max Zoom: 24
Comments: comment...

Filter

Filter: every filter matches

class	==	rail
$type	==	Polygon

Add filter

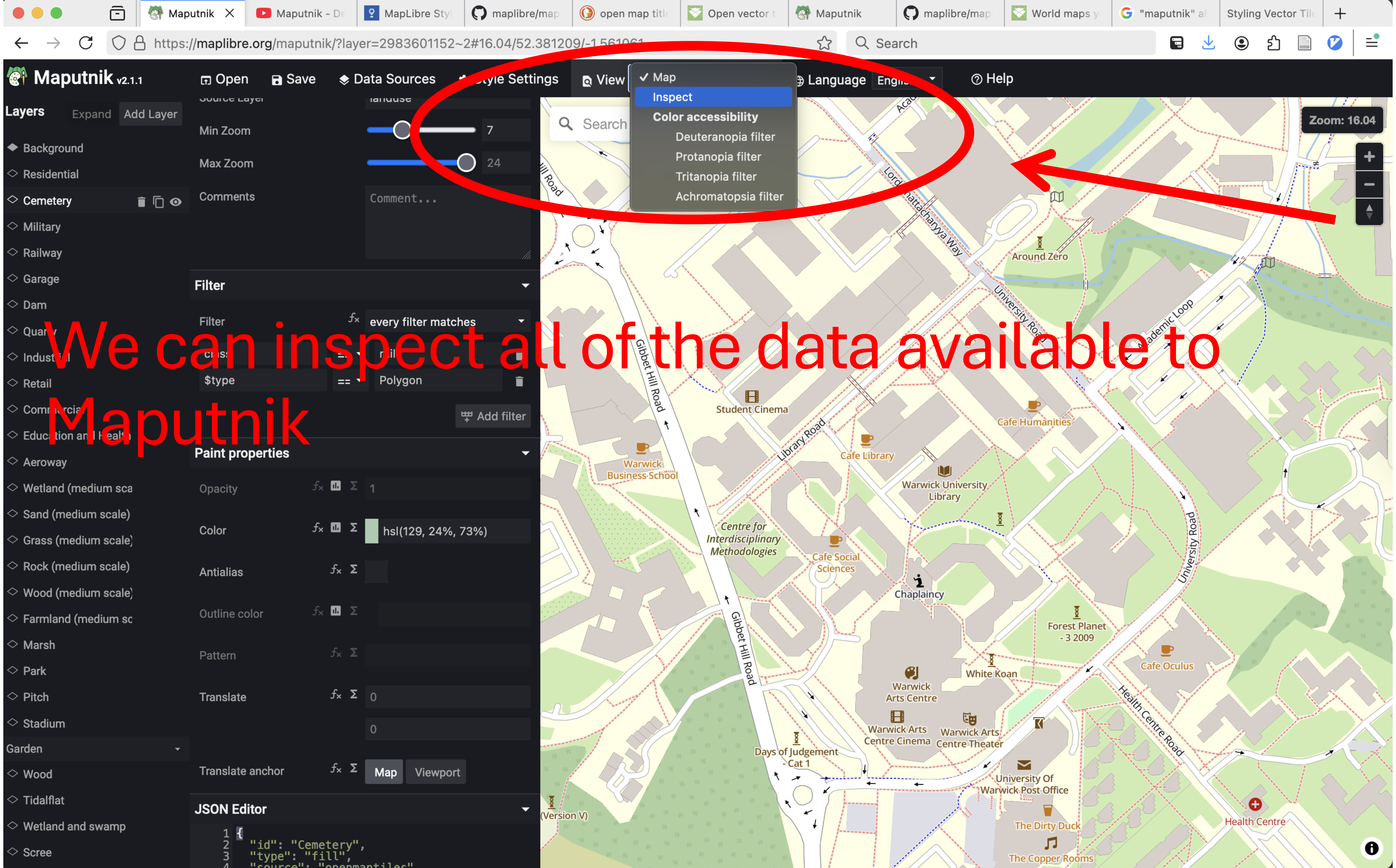
Paint properties

Opacity: 1
Color: hsl(129, 24%, 73%)
Antialias:
Outline color:
Pattern:
Translate: 0, 0
Translate anchor: [✓ Map] Viewport

JSON Editor

```
{
  "id": "Cemetery",
  "type": "fill",
  "source": "openmaptiles"
```

Zoom: 16.04



We can inspect all of the data available to Maputnik

Source Layer: landuse
Min Zoom: 7
Max Zoom: 24
Comments: Comment...

Filter

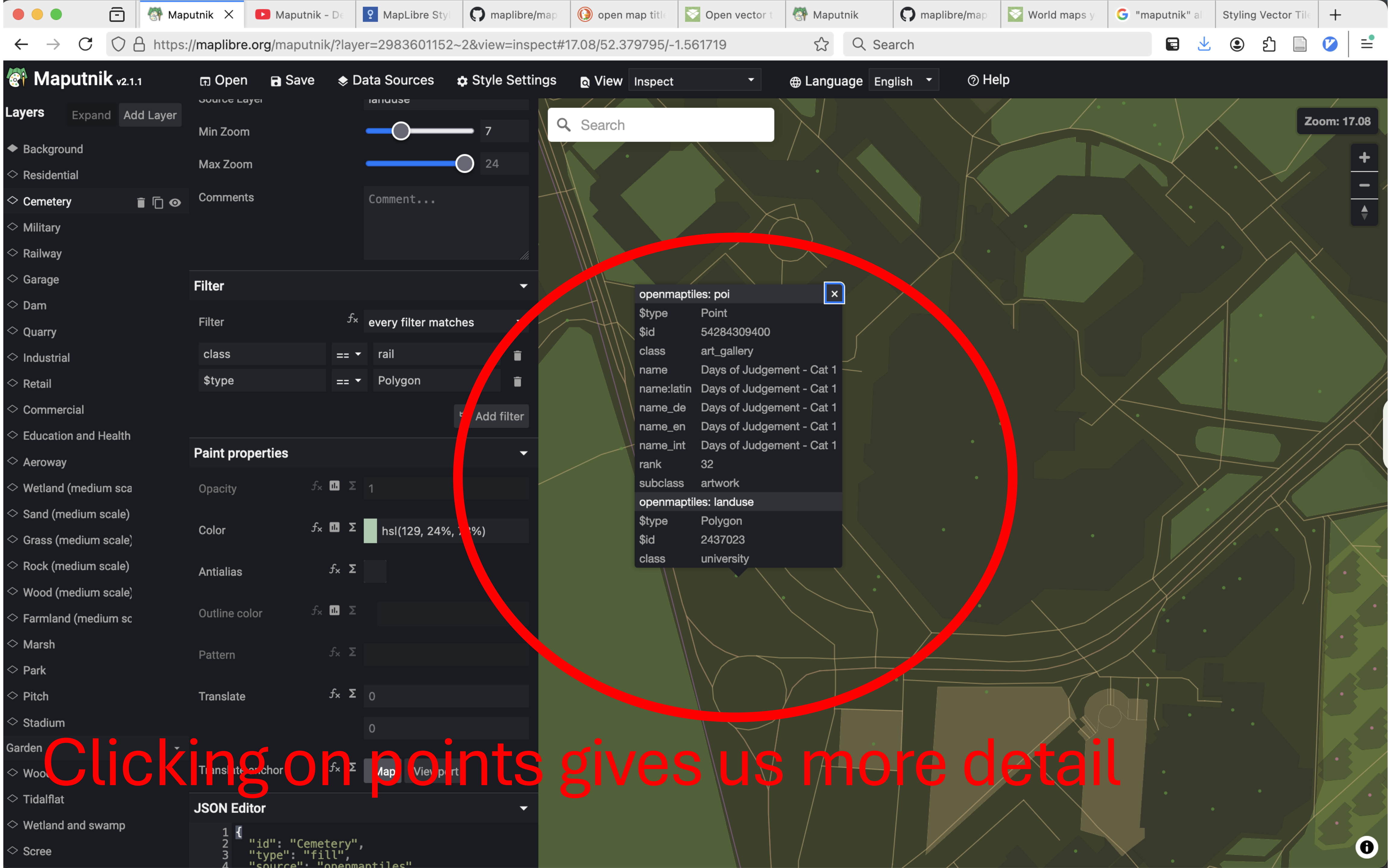
Filter: every filter matches
class == rail
$type == Polygon
Add filter

Paint properties

Opacity: 1
Color: hsl(129, 24%, 73%)
Antialias
Outline color
Pattern
Translate: 0, 0
Translate anchor: [✓ Map] Viewport

JSON Editor

```
{
  "id": "Cemetery",
  "type": "fill",
  "source": "openmaptiles",
```



https://maplibre.org/maputnik/?layer=2983601152~2&view=inspect#17.08/52.379795/-1.561719

Layers — Expand — Add Layer

- Background
- Residential
- Cemetery
- Military
- Railway
- Garage
- Dam
- Quarry
- Industrial
- Retail
- Commercial
- Education and Health
- Aeroway
- Wetland (medium sca
- Sand (medium scale)
- Grass (medium scale)
- Rock (medium scale)
- Wood (medium scale)
- Farmland (medium sc
- Marsh
- Park
- Pitch
- Stadium
- Garden
- Woo
- Tidalflat
- Wetland and swamp
- Scree

Source Layer	landuse
Min Zoom	7
Max Zoom	24
Comments	Comment...

Filter

Filter: every filter matches

class	==	rail
$type	==	Polygon

Add filter

Paint properties

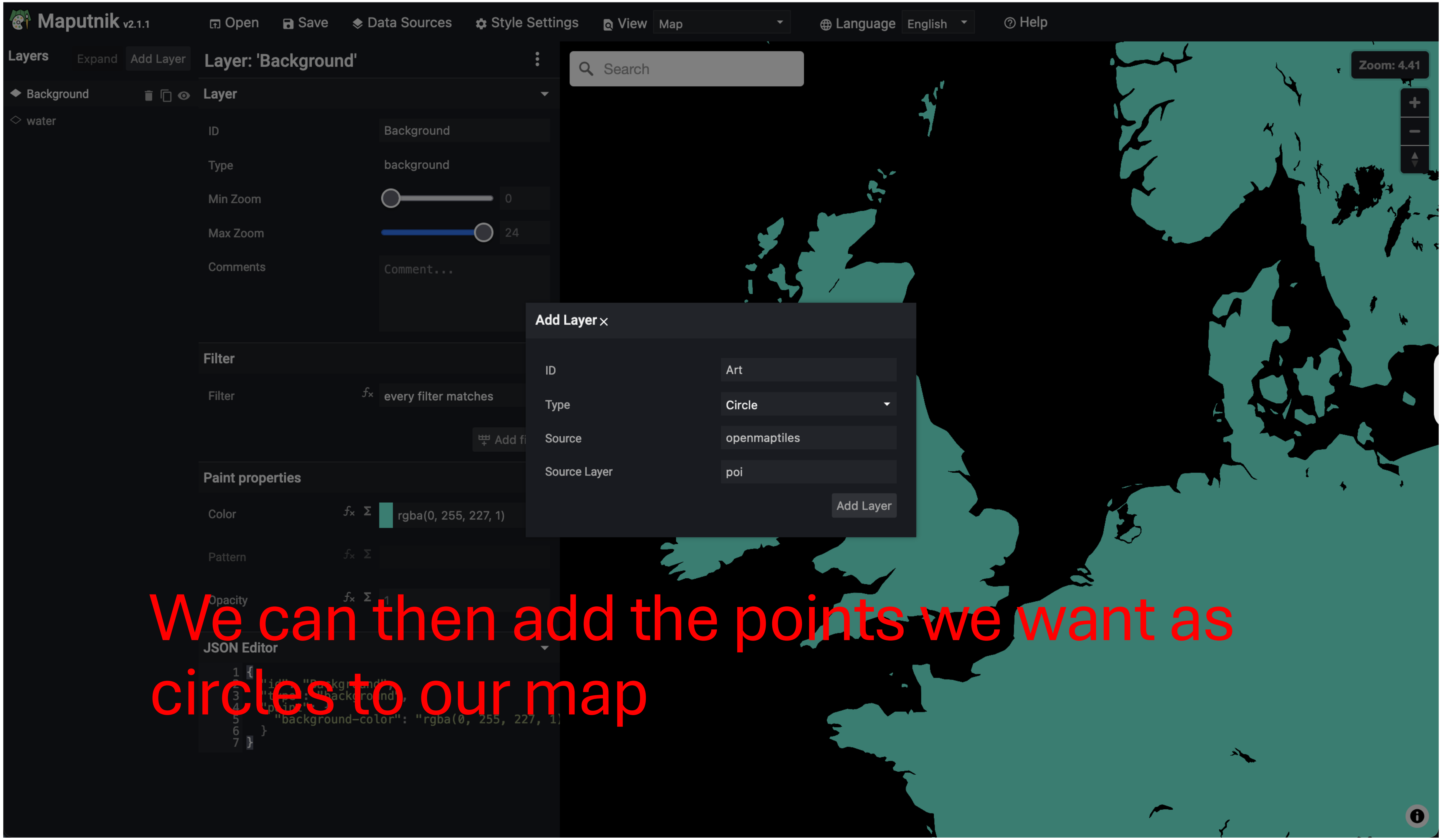
Opacity	1
Color	hsl(129, 24%, %)
Antialias	
Outline color	
Pattern	
Translate	0, 0
Translate anchor	Map / Viewport

JSON Editor

```
{
  "id": "Cemetery",
  "type": "fill",
  "source": "openmaptiles",
```

openmaptiles: poi	
$type	Point
$id	54284309400
class	art_gallery
name	Days of Judgement - Cat 1
name:latin	Days of Judgement - Cat 1
name_de	Days of Judgement - Cat 1
name_en	Days of Judgement - Cat 1
name_int	Days of Judgement - Cat 1
rank	32
subclass	artwork
openmaptiles: landuse	
$type	Polygon
$id	2437023
class	university

Clicking on points gives us more detail



Layer: 'Background'

Layer

ID	Background
Type	background
Min Zoom	0
Max Zoom	24
Comments	Comment...

Filter

Filter: every filter matches

Paint properties

Color	rgba(0, 255, 227, 1)
Pattern	
Opacity	1

JSON Editor

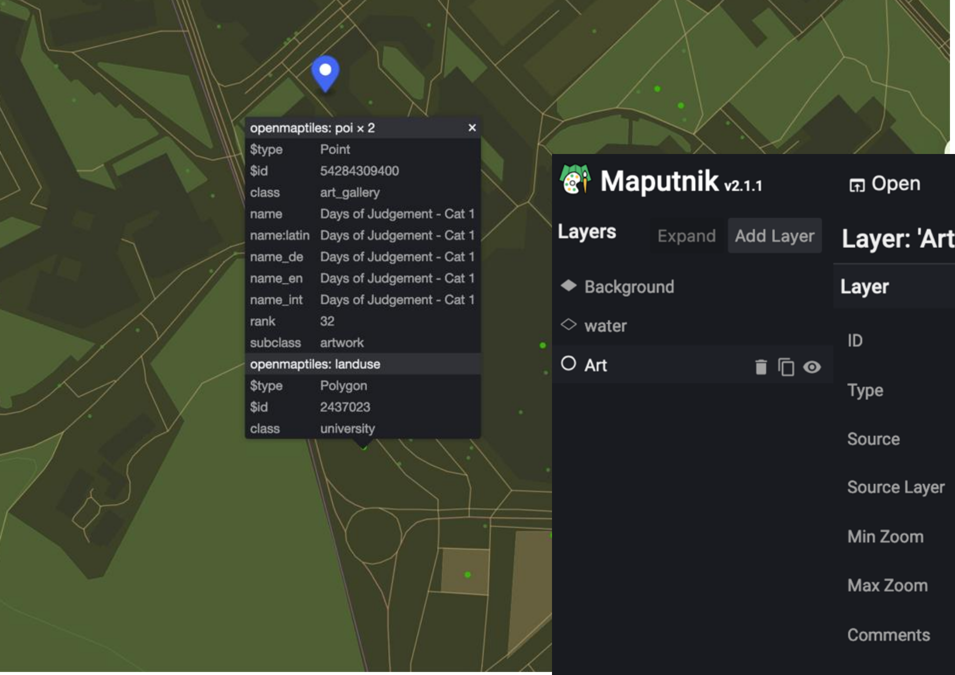
```
{
  "id": "background",
  "type": "background",
  "paint": {
    "background-color": "rgba(0, 255, 227, 1)
  }
}
```

Add Layer

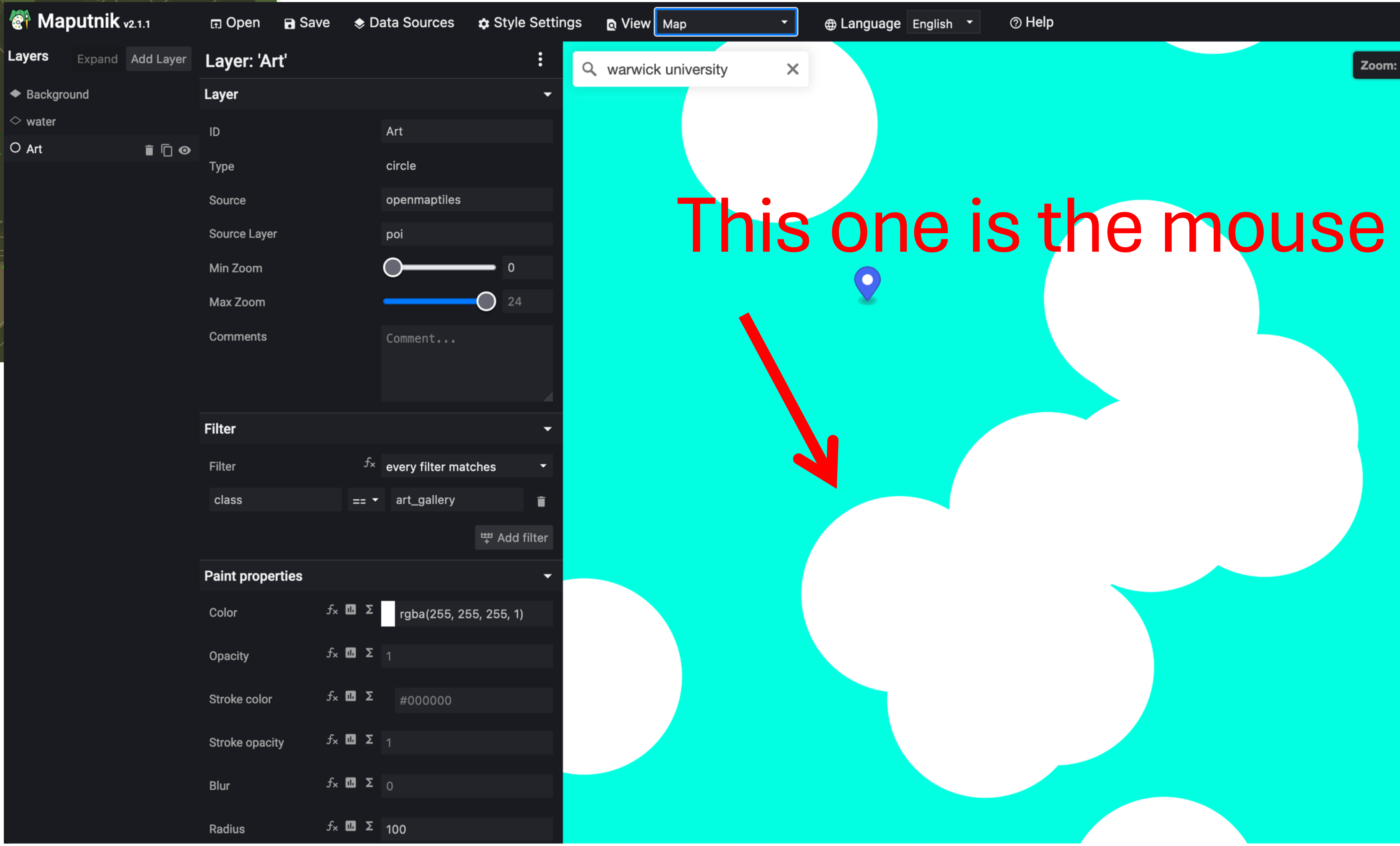
ID	Art
Type	Circle
Source	openmaptiles
Source Layer	poi

Add Layer

We can then add the points we want as circles to our map



openmaptiles: poi × 2	
$type	Point
$id	54284309400
class	art_gallery
name	Days of Judgement - Cat 1
name:latin	Days of Judgement - Cat 1
name_de	Days of Judgement - Cat 1
name_en	Days of Judgement - Cat 1
name_int	Days of Judgement - Cat 1
rank	32
subclass	artwork
openmaptiles: landuse	
$type	Polygon
$id	2437023
class	university



Layer: 'Art'

Layer

ID	Art
Type	circle
Source	openmaptiles
Source Layer	poi
Min Zoom	0
Max Zoom	24
Comments	Comment...

Filter

Filter	every filter matches		
class	==	art_gallery	

Add filter

Paint properties

Color	rgba(255, 255, 255, 1)
Opacity	1
Stroke color	#000000
Stroke opacity	1
Blur	0
Radius	100

This one is the mouse

Thanks, Tim, I can’t wait to make exactly the map I want….

Actually Maputnik sort of has a lot of limitations

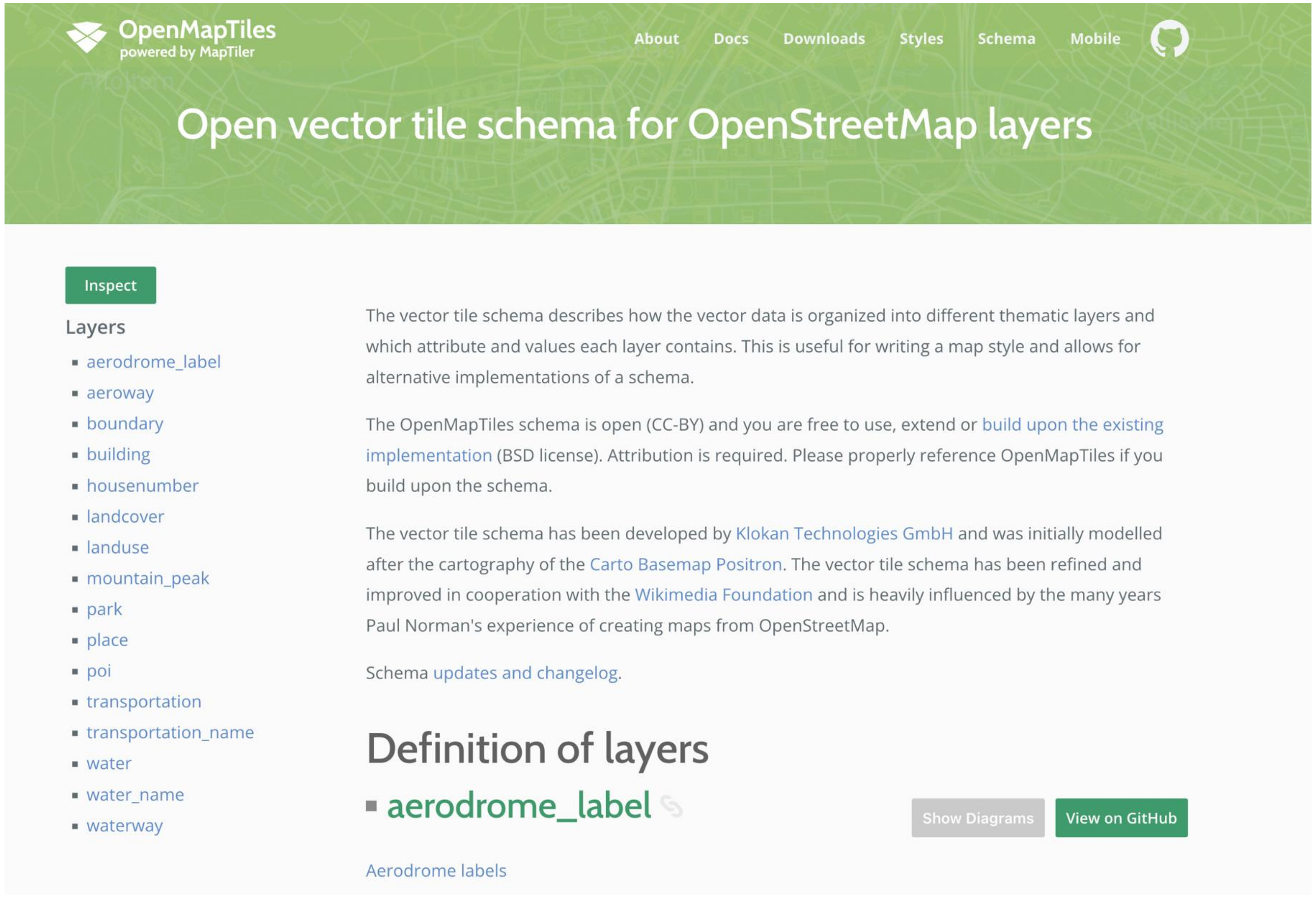
- Limited data from OSM
- Assumptions about crowding and zoom levels
- Fixed iconography

Actually Maputnik sort of has a lot of limitations

- Limited data from OSM
- Assumptions about crowding and zoom levels
- Fixed iconography

Which can all be worked around, if you really want to…

Limited data from OSM



Open vector tile schema for OpenStreetMap layers

Layers

- aerodrome_label
- aeroway
- boundary
- building
- housenumber
- landcover
- landuse
- mountain_peak
- park
- place
- poi
- transportation
- transportation_name
- water
- water_name
- waterway

The vector tile schema describes how the vector data is organized into different thematic layers and which attribute and values each layer contains. This is useful for writing a map style and allows for alternative implementations of a schema.

The OpenMapTiles schema is open (CC-BY) and you are free to use, extend or build upon the existing implementation (BSD license). Attribution is required. Please properly reference OpenMapTiles if you build upon the schema.

The vector tile schema has been developed by Klokan Technologies GmbH and was initially modelled after the cartography of the Carto Basemap Positron. The vector tile schema has been refined and improved in cooperation with the Wikimedia Foundation and is heavily influenced by the many years Paul Norman's experience of creating maps from OpenStreetMap.

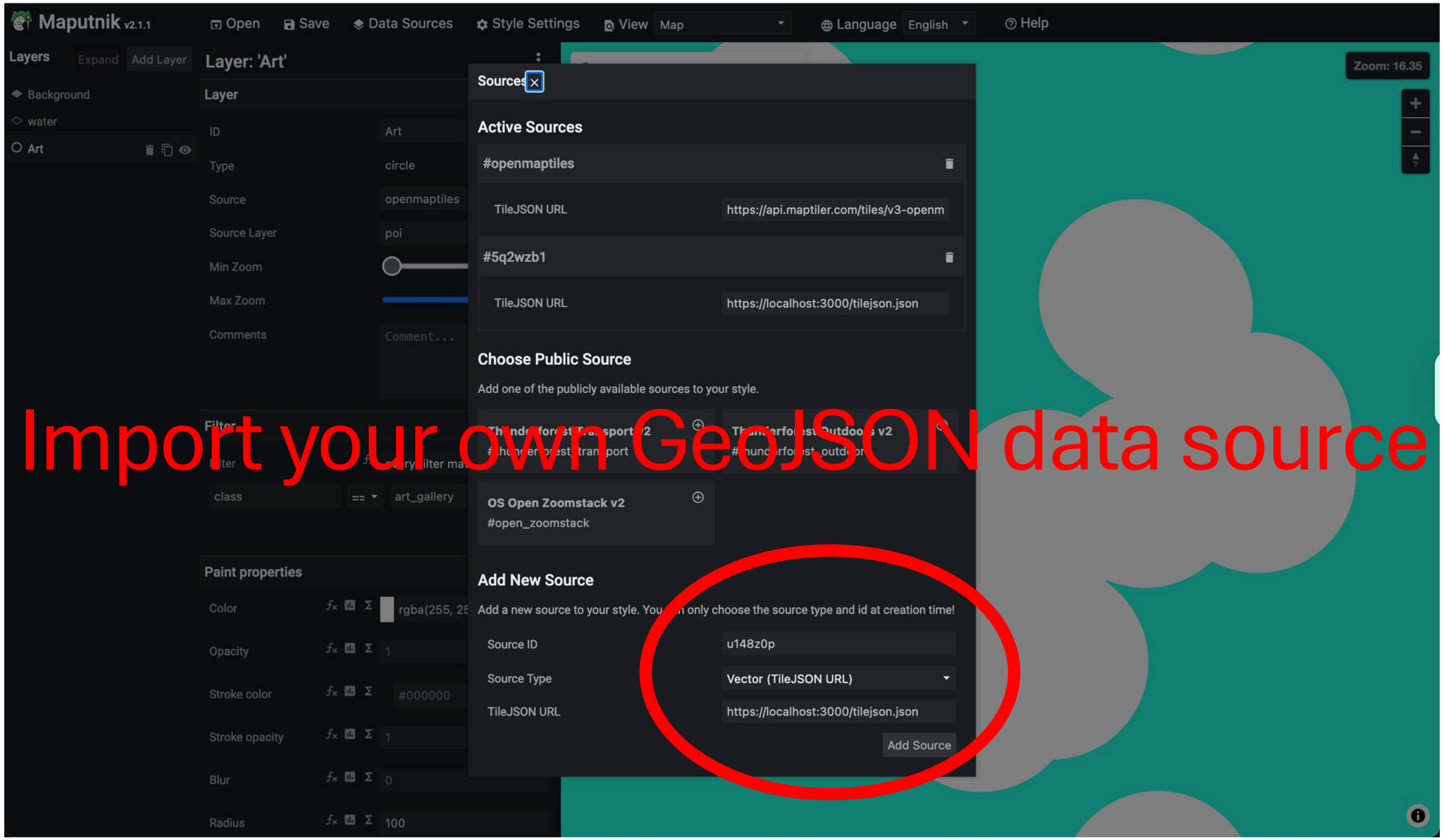
Schema updates and changelog.

Definition of layers

- aerodrome_label

Aerodrome labels

Work around



Import your own GeoJSON data source

Sources

Active Sources

#openmaptiles

TileJSON URL	https://api.maptiler.com/tiles/v3-openm

#5q2wzb1

TileJSON URL	https://localhost:3000/tilejson.json

Choose Public Source

Add one of the publicly available sources to your style.

OS Open Zoomstack v2
#open_zoomstack

Add New Source

Add a new source to your style. You can only choose the source type and id at creation time!

Source ID	u148z0p
Source Type	Vector (TileJSON URL)
TileJSON URL	https://localhost:3000/tilejson.json

Add Source

Layer: 'Art'

ID	Art
Type	circle
Source	openmaptiles
Source Layer	poi
Min Zoom	
Max Zoom	
Comments	Comment...

Filter: class == art_gallery

Paint properties

Color	rgba(255, 25
Opacity	1
Stroke color	#000000
Stroke opacity	1
Blur	0
Radius	100

Work around

- Find the feature you are looking for (overpass / taginfo)
- Query and download the data from overpass
- Host the GeoJSON somewhere yourself (could be github)
- Import it as a layer in Maputnik

Zoom levels

- Convent the geographical data you want to GeoJSON
- Use TippeCanoe (a tool for converting geodata to vector files) command line to adjust the underlying zoom level assumptions

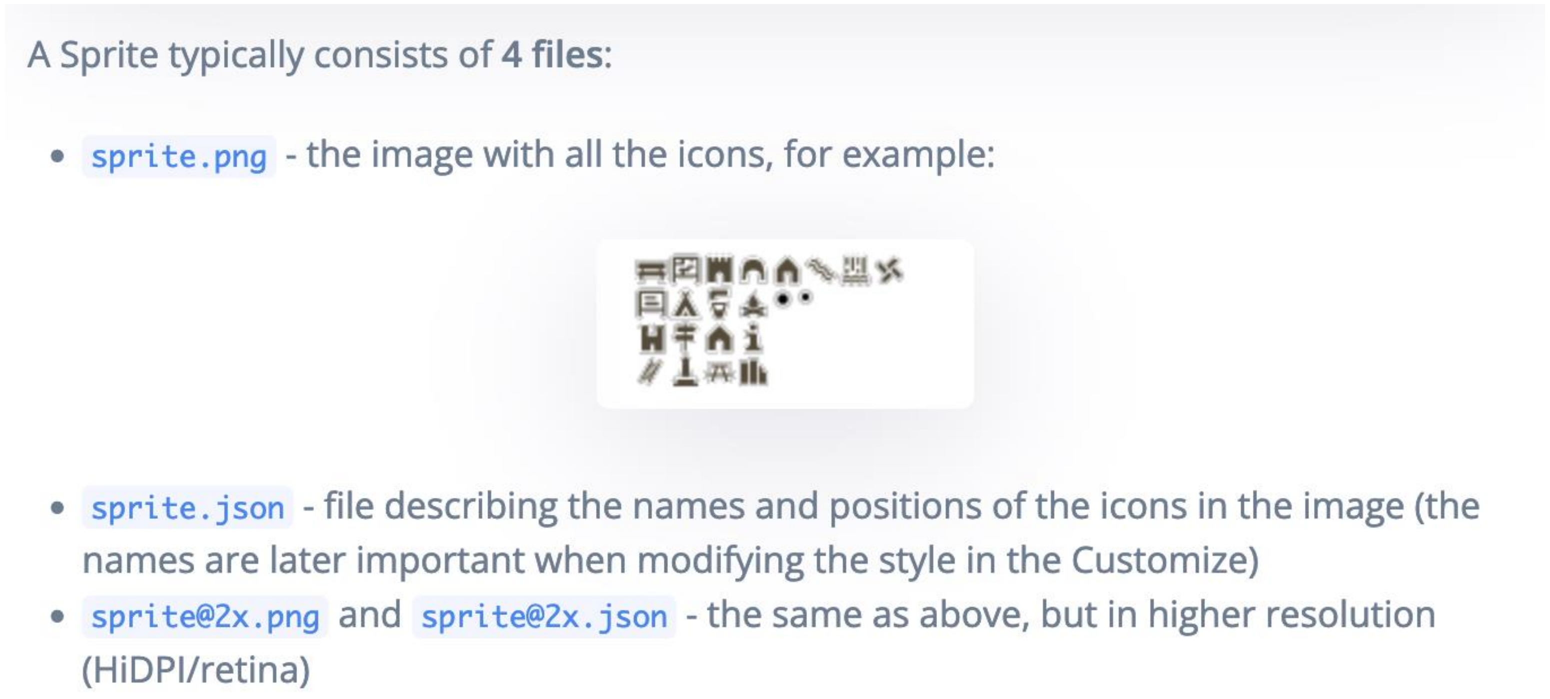
```
    tippecanoe -o output-file -Z zoom_level_in -z
zoom_level_out input_file
```

- Host the .mbtiles output on maptiler or self-host as a data source

(see: https://www.theinformationlab.co.uk/community/blog/zoom-levels-mapbox/)

Iconography

- Icons are not just an image
- Icons are a mix of one image and a separate JSON file – which Maputnik then looks up



A Sprite typically consists of **4 files**:

- `sprite.png` - the image with all the icons, for example:

- `sprite.json` - file describing the names and positions of the icons in the image (the names are later important when modifying the style in the Customize)
- `sprite@2x.png` and `sprite@2x.json` - the same as above, but in higher resolution (HiDPI/retina)

Iconography

- You can create your own sprite packs using command line tools like Spreet

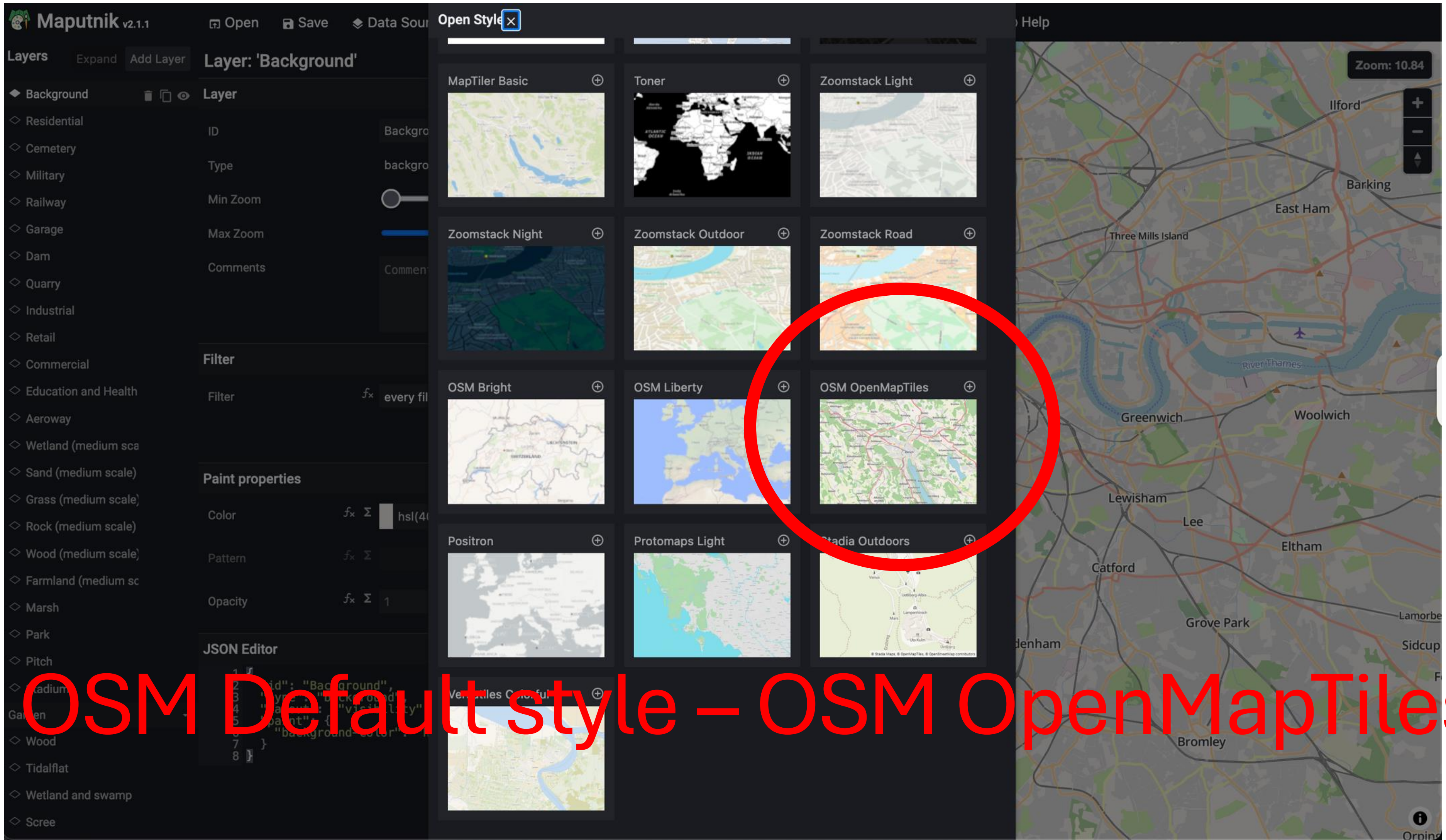
- You need to do this for **ALL** the symbols on your map at once

- Then you need to host these files (as an API!) for Maputnik to query

OpenSource software as a service

- While MapTiler is the driving force behind Maputnik and lots of these software – and this is to be applauded – they also sell services that use this software
- Notably, lots of these ‘problems’ are more easily fixable if you pay for MapTiler services or consult with them to make a map for you …

What we are going to do today



OSM Default style – OSM OpenMapTiles

What we are going to do today

- Try editing the default map to make it closer to what we want
  - Delete unwanted layers
  - Try adding in new ones
  - Changing colors and styles of existing layers
- Making a (post-it) note every time we encounter
  - Can’t change something we want to
  - Find something we want to add (but isn’t there on Maputnik)
  - Different symbols we would want to use (sketch them out!)
- Save your resulting JSON file!