Creating Printable Maps

How to convert OSM data into printable files

Hartmut Holzgraefe

hartmut@php.net

August 21, 2016

Overview

1. The Simple Plan
2. A better way ...
3. An even better way ...
4. Running your own ...
5. Other tools
6. The End ...

The Obligatory "Who am I?"

- Hartmut Holzgraefe
- Living in Bielefeld, Germany
- Studied electric engineering and computer science
- Worked on some PHP core features and
  PHP documentation / translation a decade ago
- Working as a Support Engineer for MariaDB now
  (and formerly for MySQL, Sun, Oracle, too)
- Started to do OpenStreetMap stuff in 2007

Motivation

- Had been using OpenStreetMap for quite a while
- Always liked to have wall maps in my flats
- Got a DIN A1 (24 inch) printer for a good price
- Lately: creating neighbourhood maps for social facilities

The Simple Plan

1. The Simple Plan
2. A better way ...
3. An even better way ...
4. Running your own ...
5. Other tools
6. The End ...

Lets just print a screenshot!

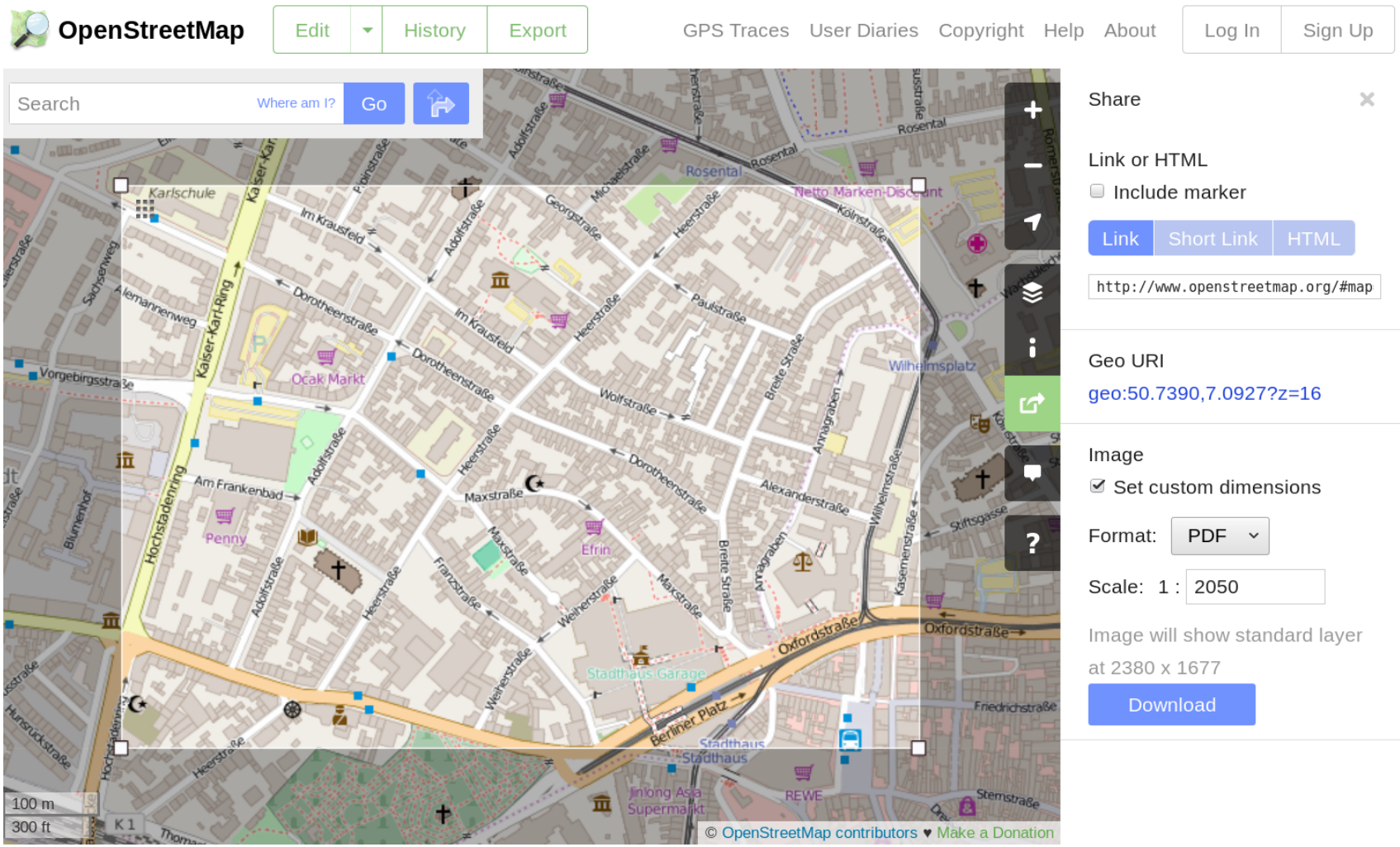
- Lets just print a screenshot!
- only good up to a certain paper size

- ... unless you have a **very** large screen

A better way ...

1. The Simple Plan
2. A better way ...
3. An even better way ...
4. Running your own ...
5. Other tools
6. The End ...

openstreetmap.org Download



Download features and woes

- Advantages:
  - Supports PNG, SVG and PDF
  - Select coordinates via a draggable box
- Disadvantages:
  - Only provides the default OSM rendering style
  - Hard to make the box fit the papers aspect ratio
  - Even harder to know which scale factor is good for which paper size
- Some of this could be worked around by post processing, but still ...

An even better way ...

1. The Simple Plan
2. A better way ...
3. An even better way ...
4. Running your own ...
5. Other tools
6. The End ...

MapOSMatic

http://maposmatic.org/

MapOSMatic Features

- Based on Mapnik renderer
- Select city to render, or rectangular map area
- Several predefined paper formats
- Select from different installed Mapnik map styles
- Optionally have a street index on the side or bottom
- Creates a single page document, or a multi page booklet
- Possible output formats are: PDF, SVG, PNG

MapOSMatic Problems

- Not really actively maintained anymore
- OSM style used on website is several years out of date
- Website has been offline for long periods several times
- Installing your own instance requires pretty old software

My own MapOSMatic instance

I've set up my own instance (covering Germany only so far) by now to cope with the frequent outages of the original site

`http://maposmatic.osm-baustelle.de/`

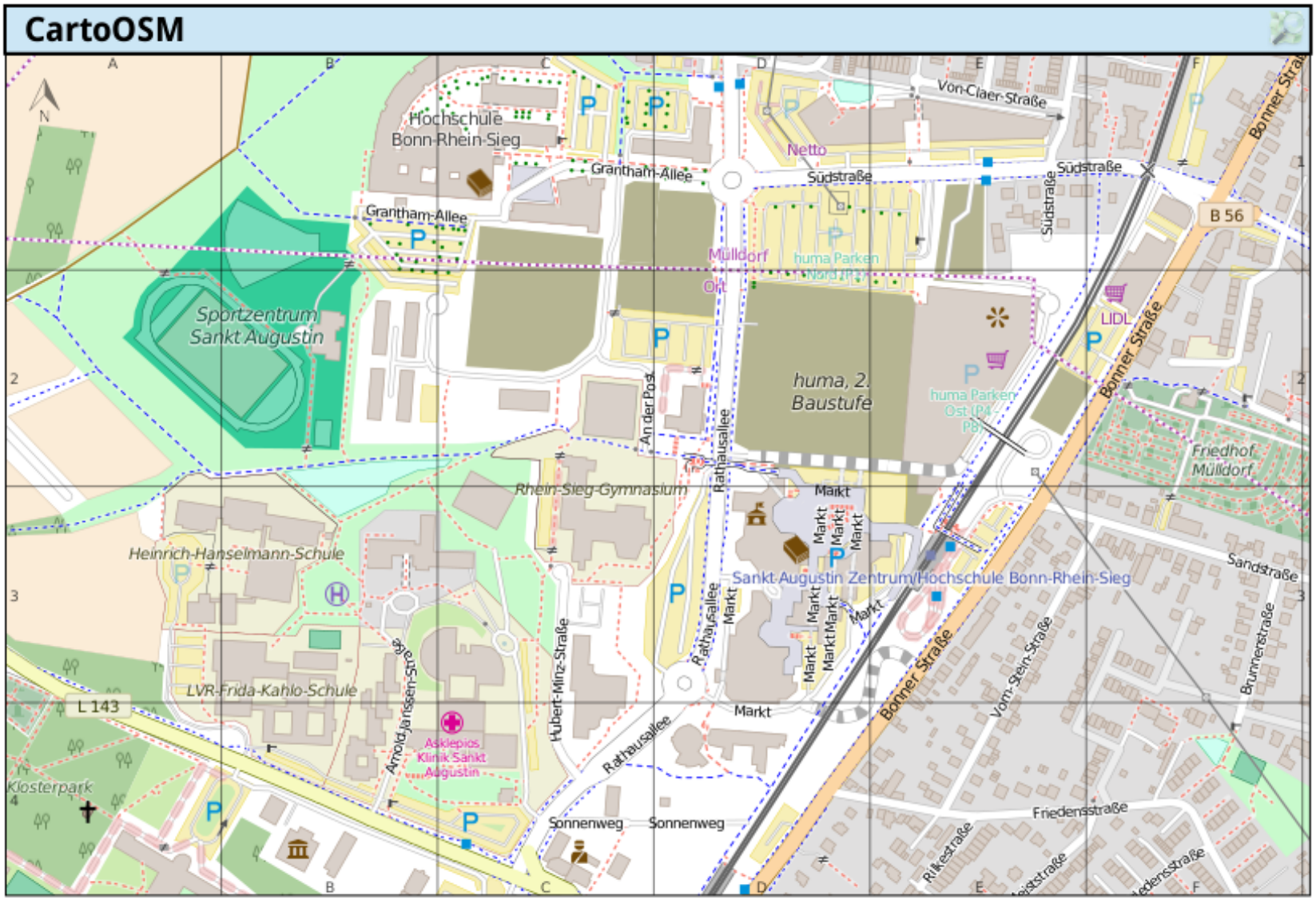
and tried to solve some of the other problems, too:

- Using up to date OSM style, plus some extra ones
- Runs with current versions of required tools and libs
- Allows to select a second style sheet as an overlay
- Fixed some other small bugs and feature requests

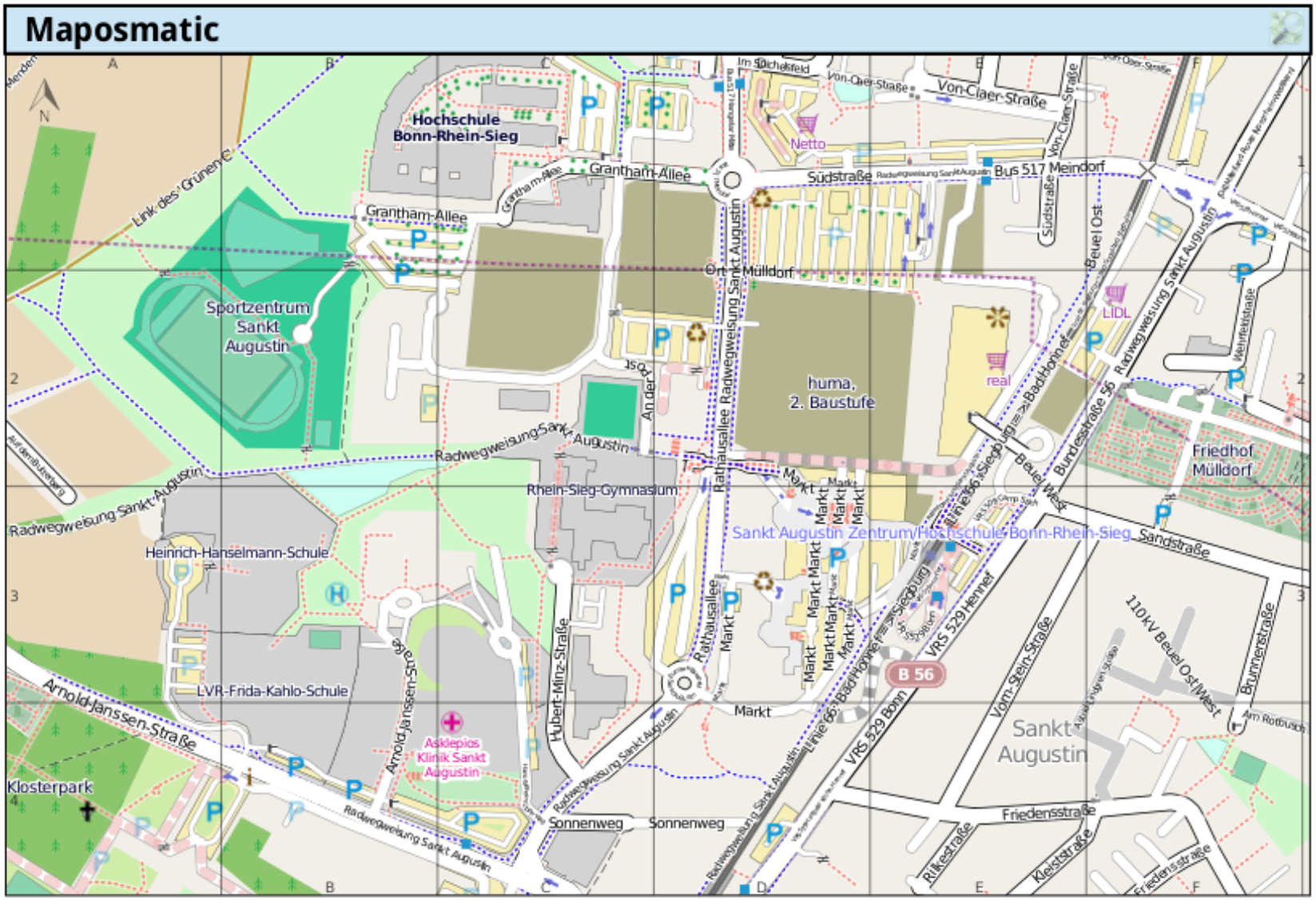
Available Stylesheets

- Current default OSM style
- MapOSMAtic printable stylesheet
- HikeBikeMap style
- HOT Humanitarian style
- MapQuest Europe stylesheet
- German OSM style
- Stripped down CartoCSS OSM style
- OpenTopoMap (but without contour lines)

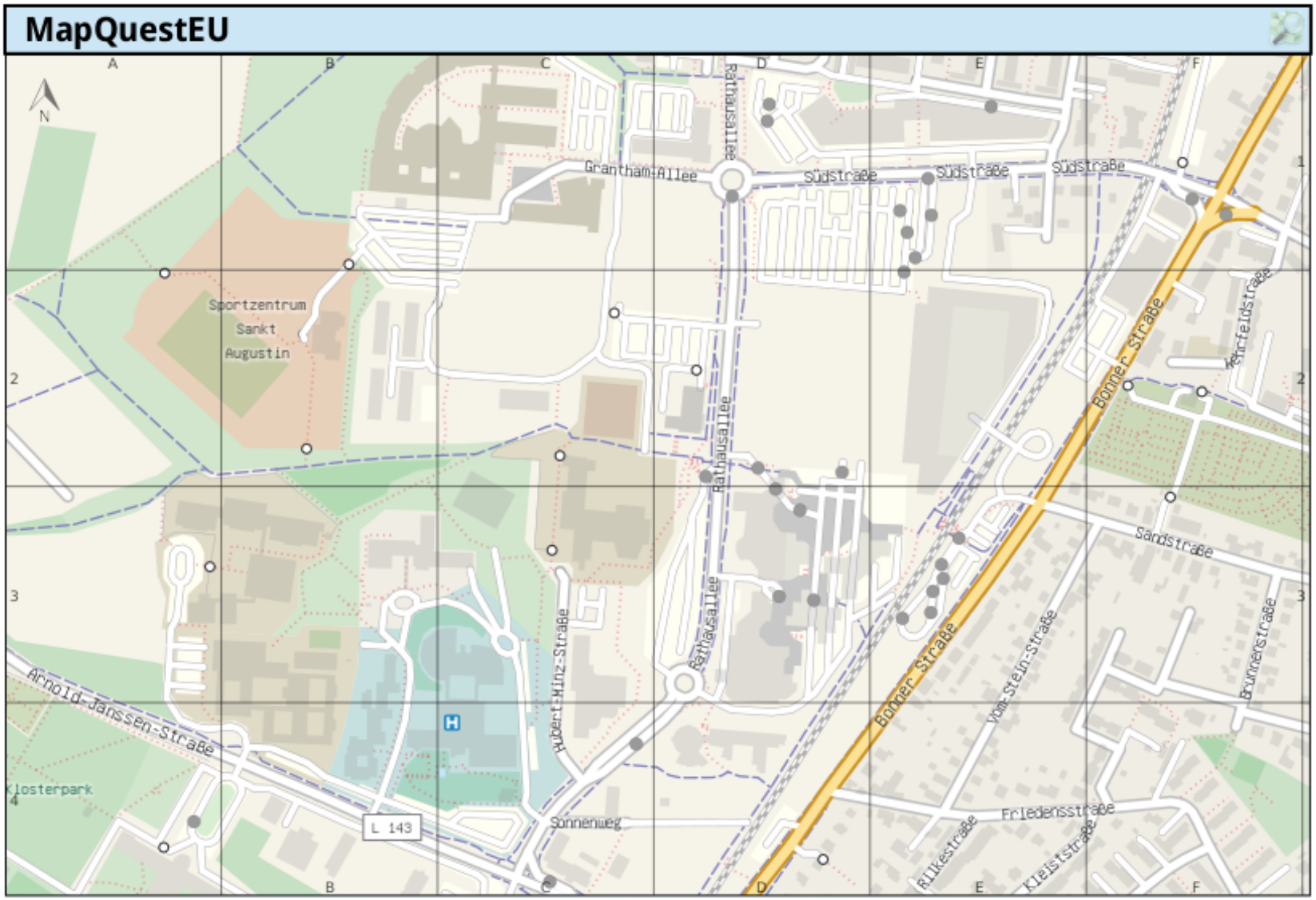
OSM default style



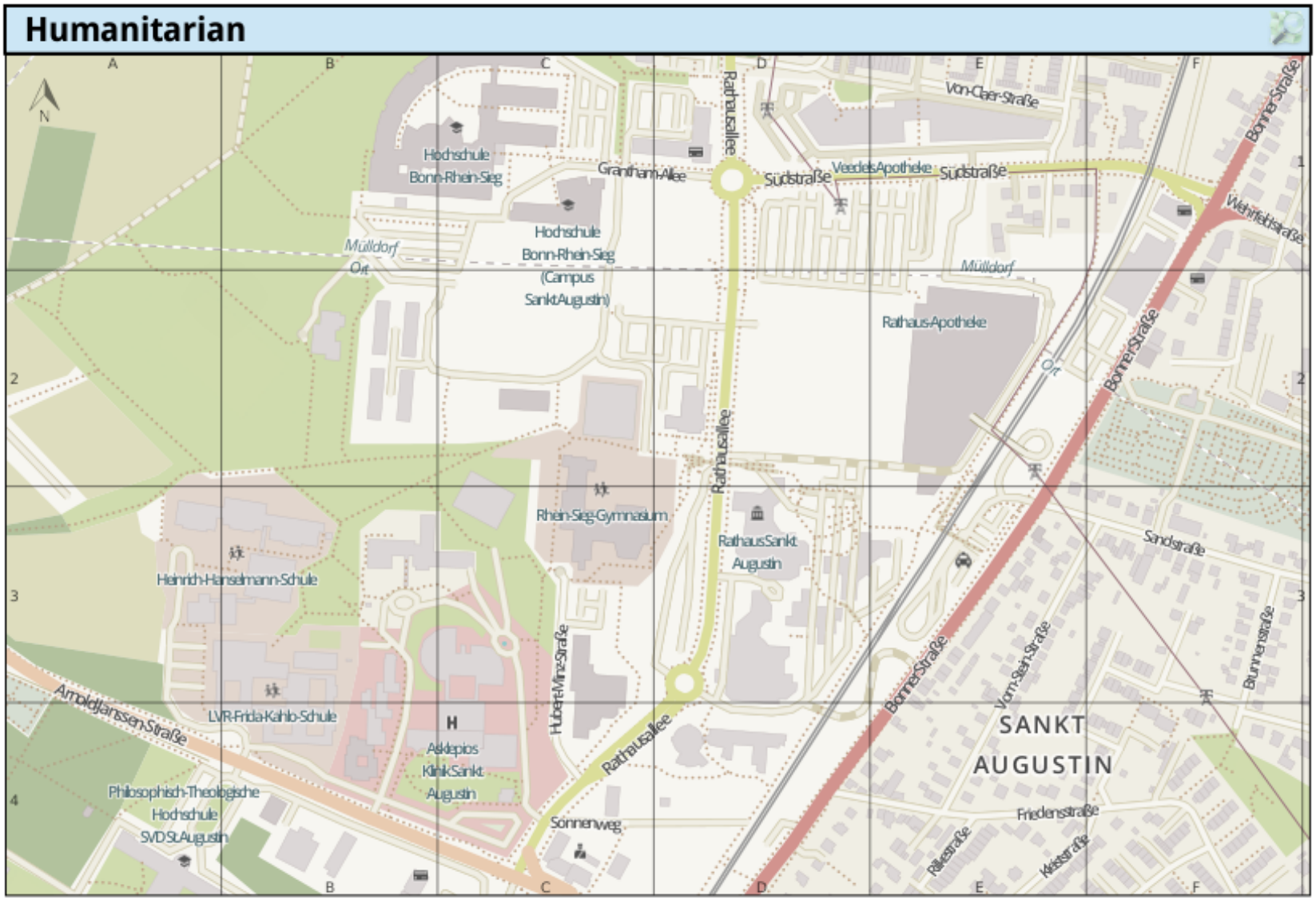
MapOSMatic Printable style



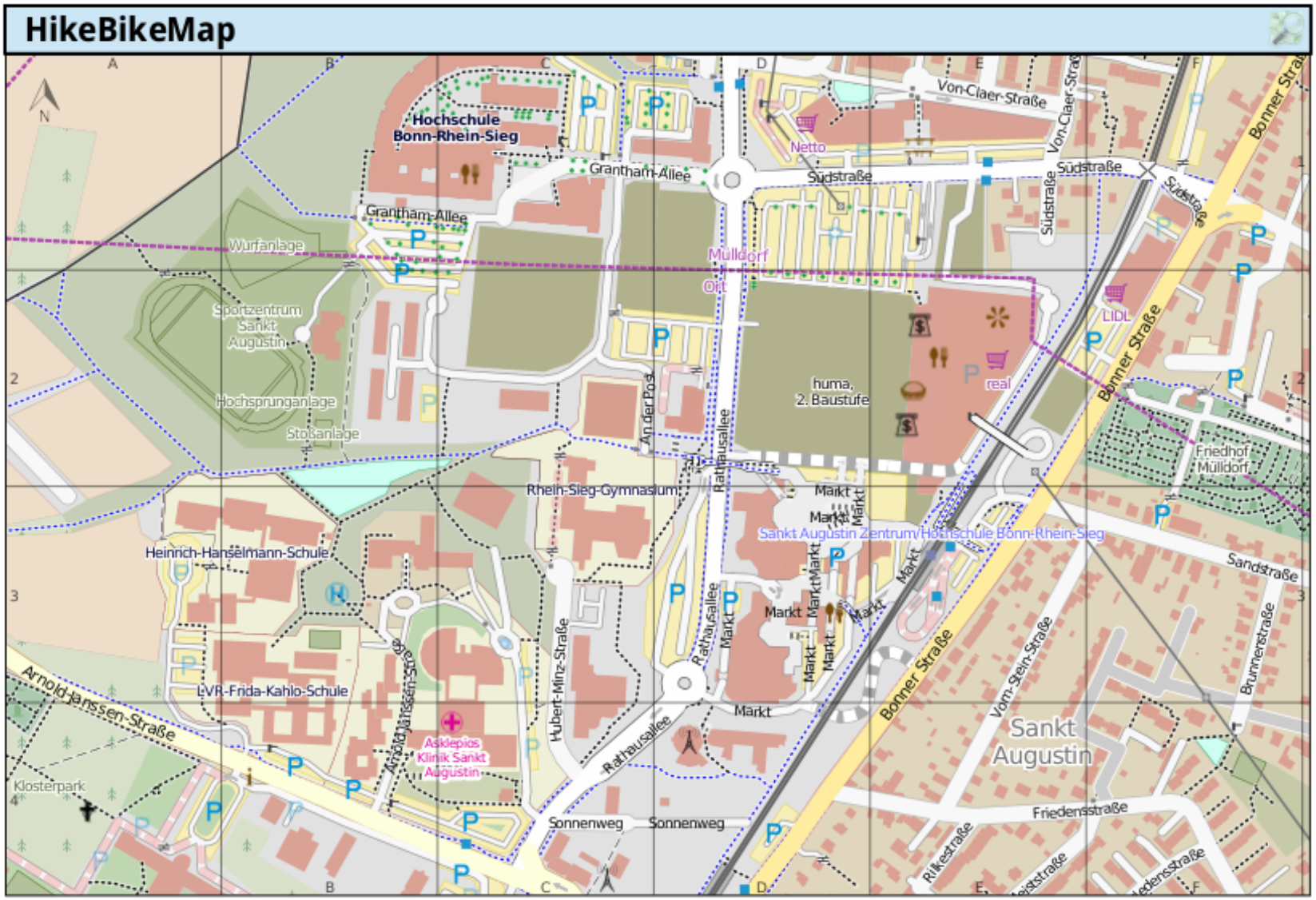
MapQuest EU style



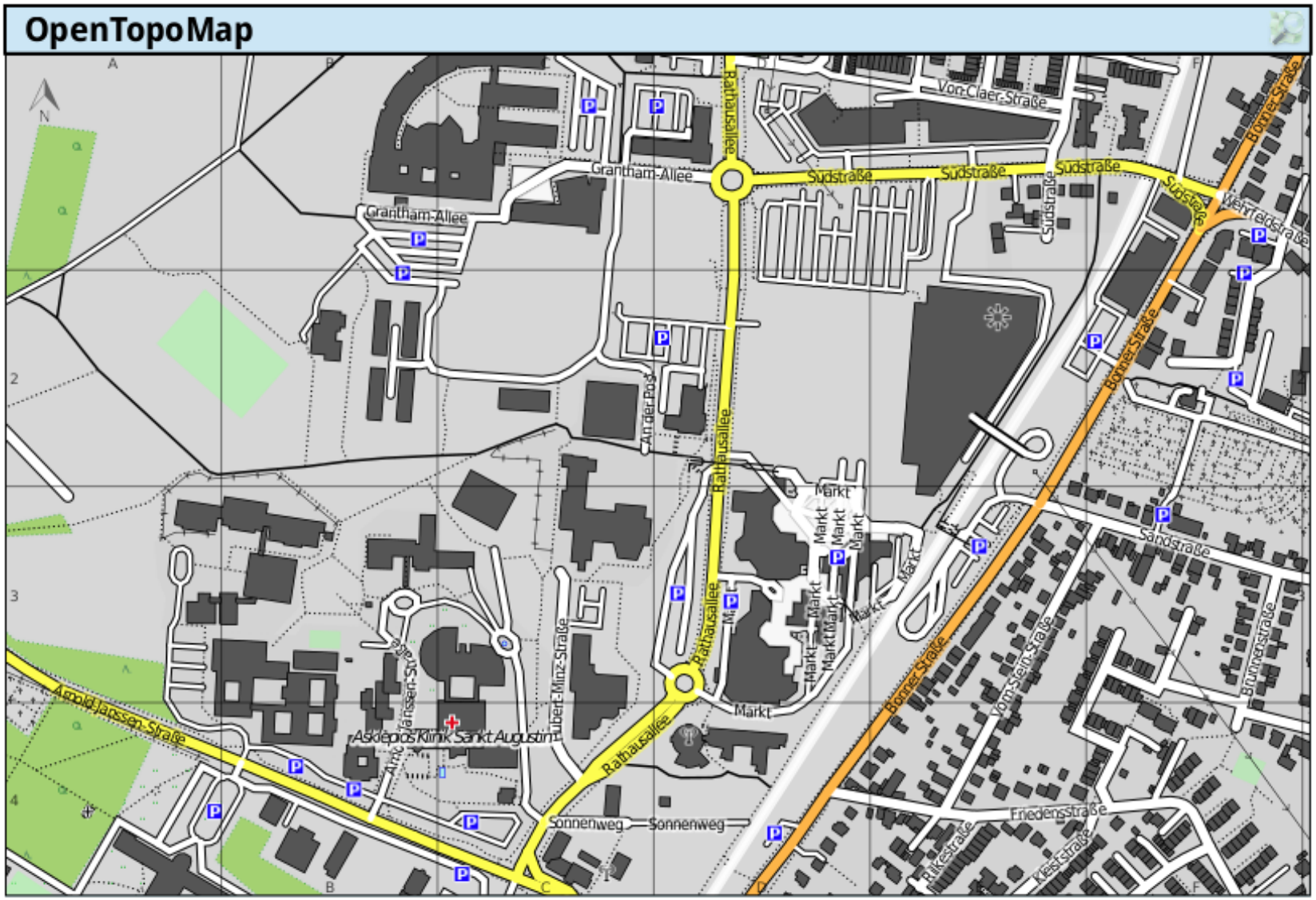
MapQuest EU style



Hike and Bike map style



OpenTopoMap style



Copyright © 2016 MapOSMatic/OCitySMap developers. Map data © 2016 OpenStreetMap.org and contributors, http://osm.org/
Map rendered on: 21 August 2016. OSM data updated on: 17 August 2016 18:21. The map may be incomplete or inaccurate.

Overlay Stylesheets

- Golf Course detail overlay
- Fire Hydrant overlay

Fire Hydrant overlay source

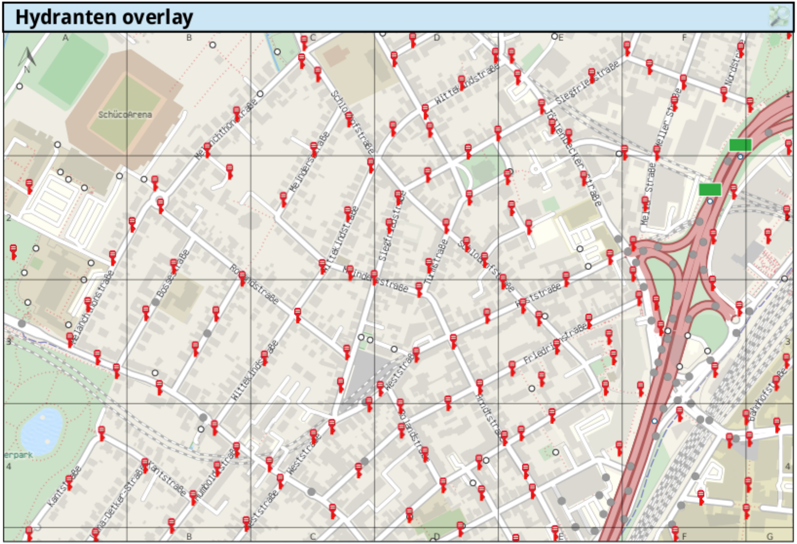
```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE Map [
]>

<Map srs="+init=epsg:3857" bgcolor="transparent">

<Style name="fire_hydrant">
  <Rule>
  <MaxScaleDenominator>20000</MaxScaleDenominator>
      <PointSymbolizer file="symbols/fire_hydrant.png"/>
  </Rule>
</Style>

<Layer name="fire_hydrant" status="on"  srs="+init=epsg:3857">
  <StyleName>fire_hydrant</StyleName>
  <Datasource>
    <Parameter name="table">
     (SELECT way
        FROM planet_osm_point
       WHERE tags->'emergency' = 'fire_hydrant'
     ) AS way
    </Parameter>
    <Parameter name="geometry_field">way</Parameter>
      <Parameter name="dbname">gis</Parameter>
      <Parameter name="estimate_extent">false</Parameter>
      <Parameter name="extent">-20037508,-19929239,20037508,19929239</Parameter>
      <Parameter name="user">maposmatic</Parameter>
      <Parameter name="type">postgis</Parameter>
      <Parameter name="port">5432</Parameter>
  </Datasource>
</Layer>
```

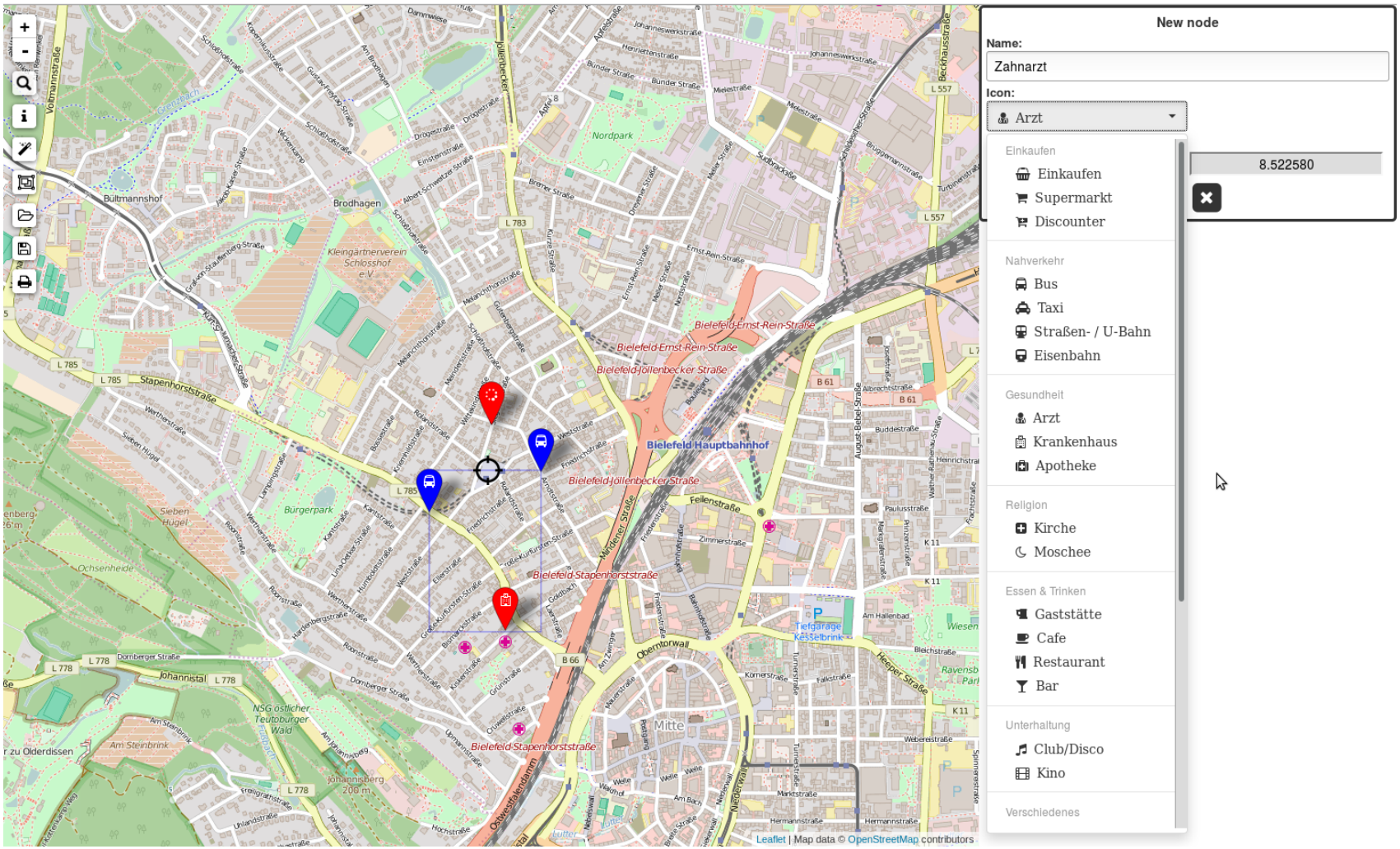
Fire Hydrant overlay



Copyright © 2016 MapOSMatic/OCitySMap developers. Map data © 2016 OpenStreetMap.org and contributors. http://osm.org/
Map rendered on: 21 August 2016. OSM data updated on: 17 August 2016 18:21. The map may be incomplete or inaccurate.

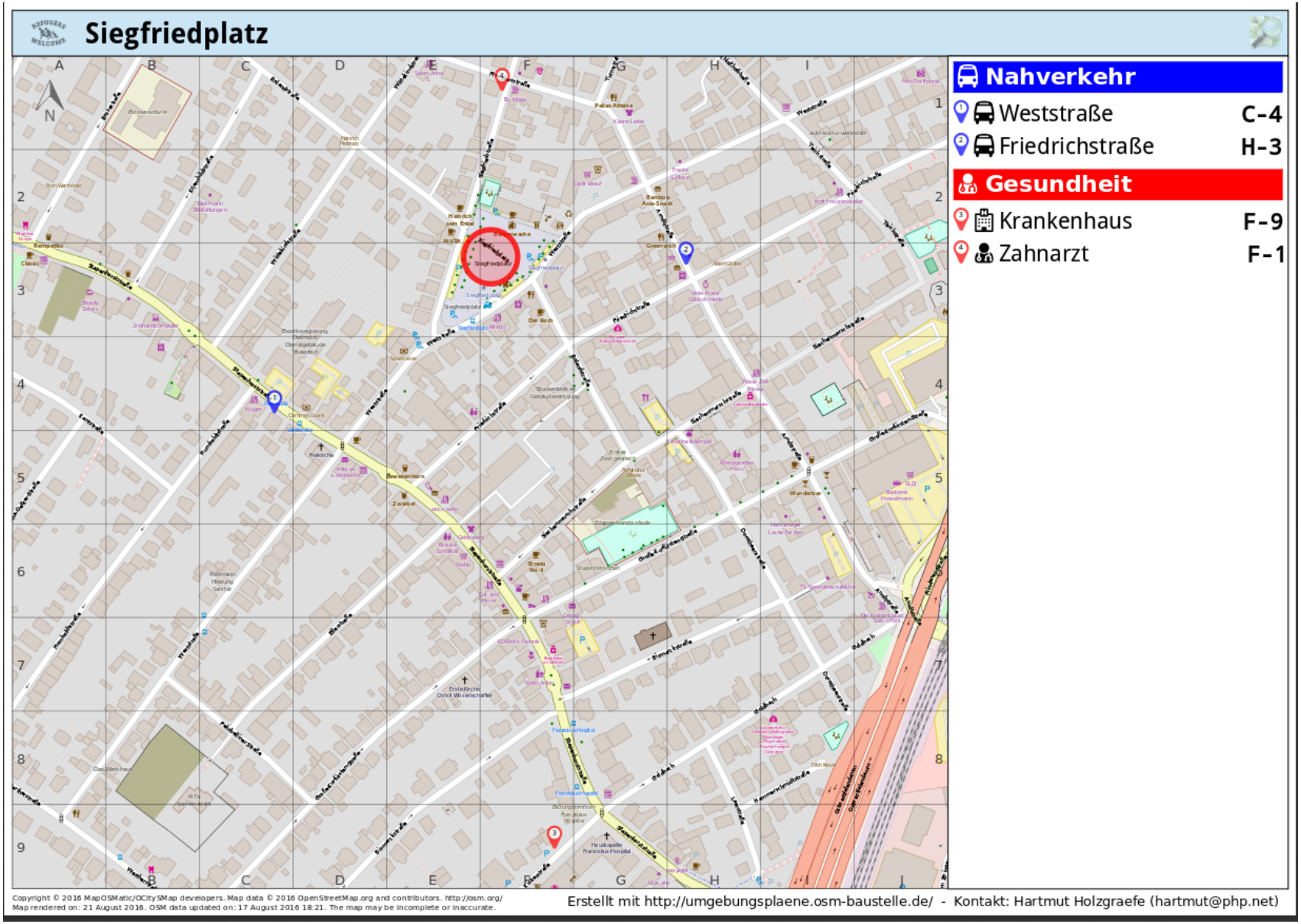
Alternative front end

I also created an alternative web frontend for creating neighbourhood plans with markers on them:
`http://umgebungsplaene.osm-baustelle.de/`



Neighbourhood plan

PDF result then looks like this:



Running your own ...

1. The Simple Plan
2. A better way ...
3. An even better way ...
4. Running your own ...
5. Other tools
6. The End ...

Why should I ...?

Maybe you want to:

- tweak stylesheets ... especially overlays
- be able to work offline
- keep your rendered maps local
- help us fix bugs or add features (just dreaming ;)
- because you can

Things to install

If you want to run Mapnik based solutions you need:

- obviously Mapnik itself
- Mapnik Style Files
- Maybe some additional files:
  - Coast lines
  - Height information (e.g. NASA SRTM)
  - ...
- OSM data imported into PostGIS in a given format
  - OSM .xml or .pbf file as input
  - typically `osm2pgsql` as import tool
  - optionally a none-default `osm2pgsql` style file

Wouldn't it be nice ...?

Wouldn't it be nice if we had something that automates this?

A vagrant box for the full stack

For this I created a Vagrant project that creates a VM with ...

- all needed components installed
- PostGIS database pre-configured
- provided OSM data file imported into PostGIS (`data.osm.pbf`)
- MapOSMatic render daemon and web interface started

Starting up the VM

```
git clone https://github.com/hholzgra/maposmatic-vagrant

cd maposmatic-vagrant

wget -o data.osm.pbf http://download.geofabrik.de/...

vagrant up
```

Once it is up and running you can connect to the MapOSMatic web interface at:

```
http://localhost:8000/
```

Other Mapnik based tools

`nik2img` is a command line tool that is part of the Mapnik distribution which can be used to directly generate maps on the command line.

`https://github.com/mapnik/mapnik/wiki/Nik2Img`

`nik4` is a slightly more advanced version that can be installed in parallel to Mapnik.

`https://github.com/Zverik/Nik4`

These tools require that you import OSM data into a PostGIS database and install the stylesheet(s) you want to use beforehand.

Both tools are installed within the VM image!

QGIS

"Create, edit, visualize, analyze and publish geospatial information"

- full stack GIS processing tool
- print is just one aspect of its capabilities
- can produce great results
- ... but with a nontrivial learning curve

Maperitive

Maperitive

- .Net application, also works with Mono
- provides a GUI, but still needs textual commands
- free to use, but not open source
- last updated in 2013 (?)

- input: OSM XML, bitmap tiles, NASA height information
- output: PNG (large or tiled), SVG, but not PDF
- has its own stylesheet language and scripting interface

Post processing with Inkscape

If you want to manually tweak the generated map before printing it you can download the generated SVG and load it into a vector drawing program.

Of the OpenSource alternatives only Inkscape is viable choice for SVGs of this size, and even Inkscape gets very slow when faced with too many vector objects.

You should use a style with lower detail level than the default OSM style when planning to process maps rendered for formats larger than DIN A3 (40x30cm).

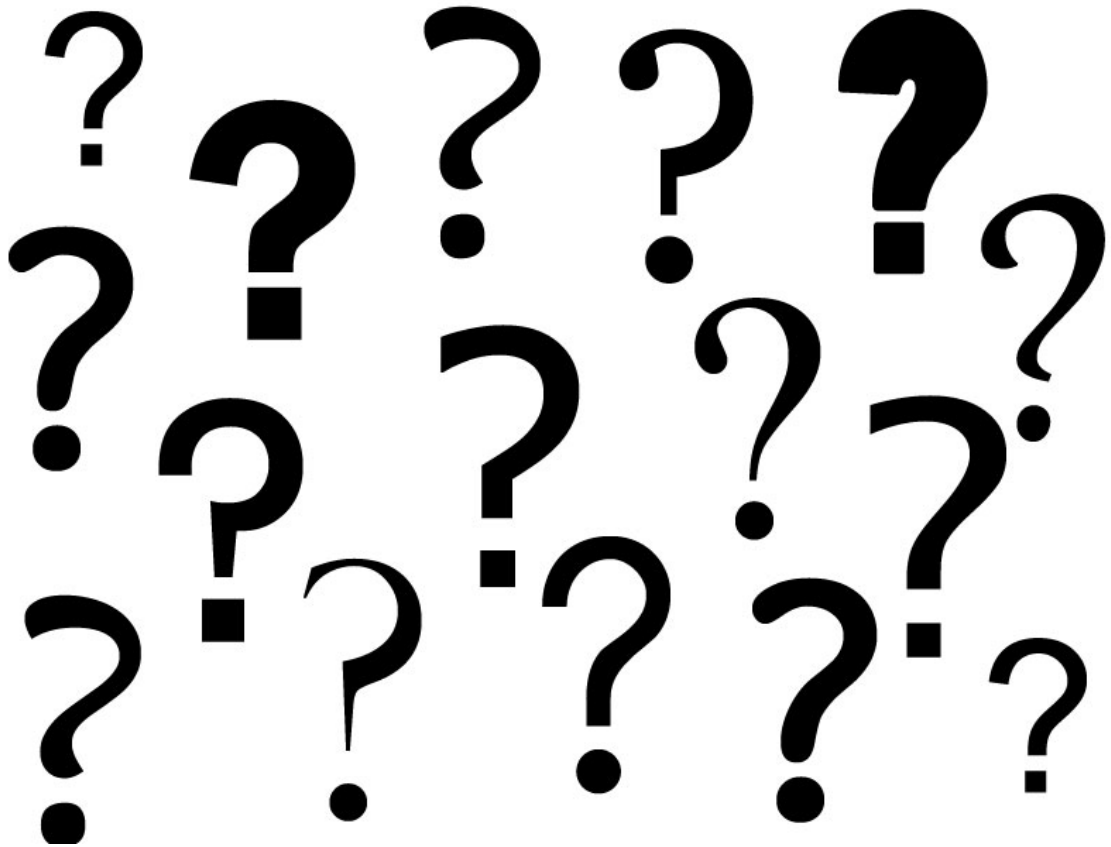
The End ...

1. The Simple Plan
2. A better way ...
3. An even better way ...
4. Running your own ...
5. Other tools
6. The End ...

Summary and Conclusions

- Printing OSM data is a bit harder than showing it on the web
- Proper tools are available, but require some preparation
- Exposing print services via a web interface looks like a good thing to do
- Parts of the process were frustrating, but the results were worth it

Questions!



References

Contact hartmut@php.net

My MapOSMatic Instance `http://maposmatic.osm-baustelle.de/`

My Neighbourhood Editor `http://umgebungsplaene.osm-baustelle.de/`

Github Projects maposmatic web interface
`https://github.com/hholzgra/maposmatic`
maposmatic render script
`https://github.com/hholzgra/ocitysmap`
maposmatic vagrant VM `https://github.com/hholzgra/maposmatic-vagrant`

The End?

Or just the beginning?