

Openstreetmap

20 Minute Primer

Me?

Thomas Christlieb

Software engineer and DBA since ~ 15 years

Likes Hiking, Geocaching, Climbing

... all of which are in need for good and accurate maps!

What is Openstreetmap (OSM)?

- Founded 2004 by Steve Coast in the UK. Inspired by Wikipedia
- Like geocaching it needed the deactivation of GPS Selective Availability in 2000
 - It's primarily a ~1TB postgresql database
 - You can create maps from it
 - It's Opensource and Free!
- It's worldwide and sometimes more accurate than any other map!
- You can use it, you can build your own maps, you even can sell them.

Just add the Tag that it's from Openstreetmap!

How can you contribute?

- Donate :-)
- Use a Mapping app on your smartphone (you can install one right now!)
 - Use the online map and map some aerial pictures
 - Add Tags to buildings (new Cafe? Changed opening hours?)

It does NOT need to be accurate. Thats the big charm. Just add the stuff, it can be fixed later if needed!

-> Simply map everything on the ground!

What node types are there?

- Camera
- Postbox
- Company
- Bus Station
 - Bank
- Telephone

-
-
-



Whats in the database?

- Nodes

Alle Eigenschaften (11)

addr:city	Ansbach	🗑️	i
addr:housenumber	8	🗑️	i
addr:postcode	91522	🗑️	i
addr:street	Bahnhofplatz	🗑️	i
description	Softwarefirma. L ...	🗑️	i
level	2	🗑️	i
name	Tradebyte Softw ...	🗑️	i
office	company	🗑️	i
phone	+49981208220	🗑️	i
website	https://tradebyt ...	🗑️	i
wheelchair	yes	🗑️	i

- Ways

Alle Eigenschaften (7)

highway	residential	🗑️	i
lit	yes	🗑️	i
maxspeed	50	🗑️	i
name	Bahnhofstraße	🗑️	i
sidewalk	both	🗑️	i
source:maxspeed	DE:urban	🗑️	i
surface	asphalt	🗑️	i

- Relations

Alle Eigenschaften (2)

restriction	no_right_turn	🗑️	i
type	restriction	🗑️	i

+

Alle Mitglieder (3)

Ortsstraße	Bahnhofplatz	↗️
from	🗑️	
Punkt	↗️	
via	🗑️	
Ortsstraße	Bahnhofstraße	↗️
to	🗑️	

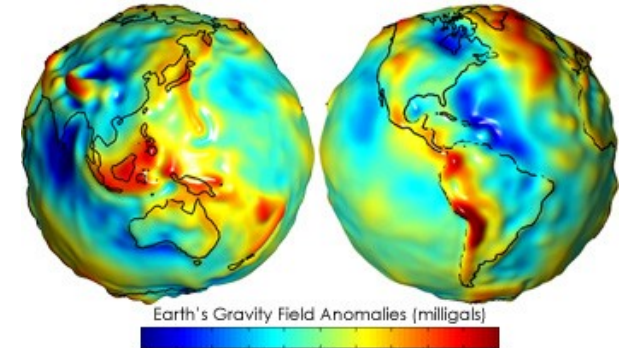
Let's look at our node in the database...

<http://download.geofabrik.de/europe/germany/bayern-latest.osm.pbf> → 540 MB for Bavaria
(import using osmosis)

```
SELECT *, ST_AsText(ST_Transform(geom, 4326)) FROM nodes WHERE tags -> 'name' = 'Tradebyte Software GmbH';
```

→ SRID 4326 is WGS84 Projection

```
-[ RECORD 1 ]-
id          | 4463317848
version     | 3
user_id     | 0
tstamp      | 2017-10-28 13:36:54
changeset_id | 0
tags        | "name"=>"Tradebyte Software GmbH", "level"=>"2", "phone"=>"+49981260220", "website"=>"https://tradebyte.com", "addr:city"=>"Ansbach", "wheelchair"=>"yes", "addr:street"=>"Bahnhofsplatz", "description"=>"Softwarefirma. Lösungen für eine nahtlose Vernetzung von Herstellern, Händlern und Retailern im digitalen Supply Chain.", "addr:postcode"=>"91522", "addr:housenumber"=>"8"
geom        | 0101000020E610000021D50B9986282540DAAB34CE4BA64840
st_astext   | POINT(10.5791519 49.2991884)
```



Some (usefull?) stuff...

Where is the next waterfall?

```
SELECT a.tags -> 'name' AS name,  
round(ST_DistanceSphere(a.geom, b.geom) / 1000) AS "Distance in km",  
ST_AsText(ST_Transform(a.geom, 4326)) AS Coordinates,  
degrees(ST_Azimuth(b.geom, a.geom)) AS Direction  
FROM  
nodes a,  
nodes b  
WHERE  
a.tags -> 'waterway' = 'waterfall'  
and a.tags -> 'name' != "  
and b.tags -> 'name' = 'Oettingen'  
and b.tags -> 'railway' = 'halt'  
order by 2;
```

name	Distance in km	coordinates	direction
Tiefenbachabsturz	70	POINT(11.3277964 49.3492889)	60.9422550314605
Röthenbach-Wasserfall	74	POINT(11.3346598 49.4042733)	57.9860563672948
Klingender Wasserfall	78	POINT(11.2983854 49.4782282)	52.640983913759
Kalkbach-Wasserfälle	91	POINT(11.2497829 49.6424409)	42.9862510409032
Kalkbach-Wasserfälle	91	POINT(11.2490756 49.6432751)	42.9210340435697
Sinterstufen	91	POINT(11.2799204 49.627587)	44.8981916459071
Ochsenklavier	91	POINT(11.0677786 49.7083163)	31.5251506708662
Moritzbach Wasserfall	94	POINT(11.1850145 49.703233)	37.6690795552998
Harnbach-Wasserfall	95	POINT(11.4636705 49.5895626)	53.2679379920274
Harnbach-Wasserfall	96	POINT(11.4661704 49.5911182)	53.2810975264065
Wedenbach-Wasserfall	107	POINT(11.217483 49.8144508)	35.399537157201
Große Kaskade	109	POINT(11.4860708 48.1576033)	131.582430526682
Ellernbach-Wasserfall	114	POINT(11.0809704 49.9161626)	26.3708914483251
Betonierter wasserfall	125	POINT(12.198349 48.5565354)	103.636621920367
Wasserfall	143	POINT(11.3774673 50.1260012)	33.3715986011868
Schleierfälle	144	POINT(10.9864347 47.6753641)	163.033585739769
Pfensag-Wasserfall	144	POINT(11.2613916 50.1699689)	28.3831474296451

Some (usefull?) stuff...

Where is the next Kentucky Fried Chicken?

```
SELECT
a.tags -> 'name' AS name,
a.tags -> 'addr:city' AS Stadt,
round(ST_DistanceSphere(c.geom, b.geom) / 1000) AS "Distance in km",
ST_AsText(ST_Transform(c.geom, 4326)) AS Coordinates,
concat(round(degrees(ST_Azimuth(b.geom, c.geom))), ' °') AS Direction
FROM
ways a, nodes b, nodes c
WHERE
c.id = a.nodes[1]
and lower(a.tags -> 'amenity') = 'fast_food'
and (lower(a.tags -> 'name') = 'kfc' or lower(a.tags -> 'name') = 'kentucky fried chicken')
and b.tags -> 'name' = 'Oettingen'
and b.tags -> 'railway' = 'halt'
```

```
UNION ALL
SELECT
a.tags -> 'name' AS name,
a.tags -> 'addr:city' AS Stadt,
round(ST_DistanceSphere(a.geom, b.geom) / 1000) AS "Distance in km",
ST_AsText(ST_Transform(a.geom, 4326)) AS Coordinates,
concat(round(degrees(ST_Azimuth(b.geom, a.geom))), ' °') AS Direction
FROM
nodes a, nodes b
WHERE
lower(a.tags -> 'amenity') = 'fast_food'
and (lower(a.tags -> 'name') = 'kfc' or lower(a.tags -> 'name') = 'kentucky fried chicken')
and b.tags -> 'name' = 'Oettingen'
and b.tags -> 'railway' = 'halt'
```

order by
3 asc;

name	stadt	Distance in km	coordinates	direction
KFC		55	POINT(11.3403355 49.0426112)	82 °
Kentucky Fried Chicken		64	POINT(11.0080418 49.4575247)	39 °
KFC	Nürnberg	66	POINT(11.0815578 49.4460633)	44 °
KFC	Nürnberg	66	POINT(11.081574 49.4460329)	44 °
Kentucky Fried Chicken		66	POINT(10.8749175 48.3813824)	154 °
KFC	Nürnberg	66	POINT(11.0682782 49.4488967)	43 °
Kentucky Fried Chicken	Augsburg	67	POINT(10.8676552 48.3713111)	155 °
KFC	Nürnberg	67	POINT(11.1256353 49.4343578)	47 °
Kentucky Fried Chicken	Augsburg	67	POINT(10.9348895 48.3887379)	149 °
Kentucky Fried Chicken	Ingolstadt	67	POINT(11.4632648 48.7487741)	103 °
KFC	Neu-Ulm	74	POINT(10.021679 48.3955108)	226 °

Name Tag can be on a way or on a node!

Some (usefull?) stuff...

Where is Hetzner if we only got the address? (Geocoding)

```
SELECT
nodes
FROM
ways
WHERE
tags -> 'addr:housenumber' = '25'
and tags -> 'addr:street' = 'Industriestraße'
and tags -> 'addr:postcode' = '91710'
and tags -> 'addr:city' = 'Gunzenhausen';
```

nodes

```
-----
{2436744740,2436744741,2436744743,2436744744,2436744745,2486322519,2436744747,2436744748,2436744749,2436744751,2436744752,2436744753,2436744740}
(1 row)
```

```
SELECT *,
ST_AsText(ST_Transform(geom, 4326))
FROM nodes
where id = 2436744740;
```

id	version	user_id	tstamp	changeset_id	tags	geom	st_astext
2436744740	2	0	2013-10-07 13:59:28	0		0101000020E61000006F6AA0F99C8F2540365B79C9FF8F4840	POINT(10.7804945 49.1249935)

Some (usefull?) stuff...

What is around our actual 2 km of our GPS Location? (Reverse Geocoding)

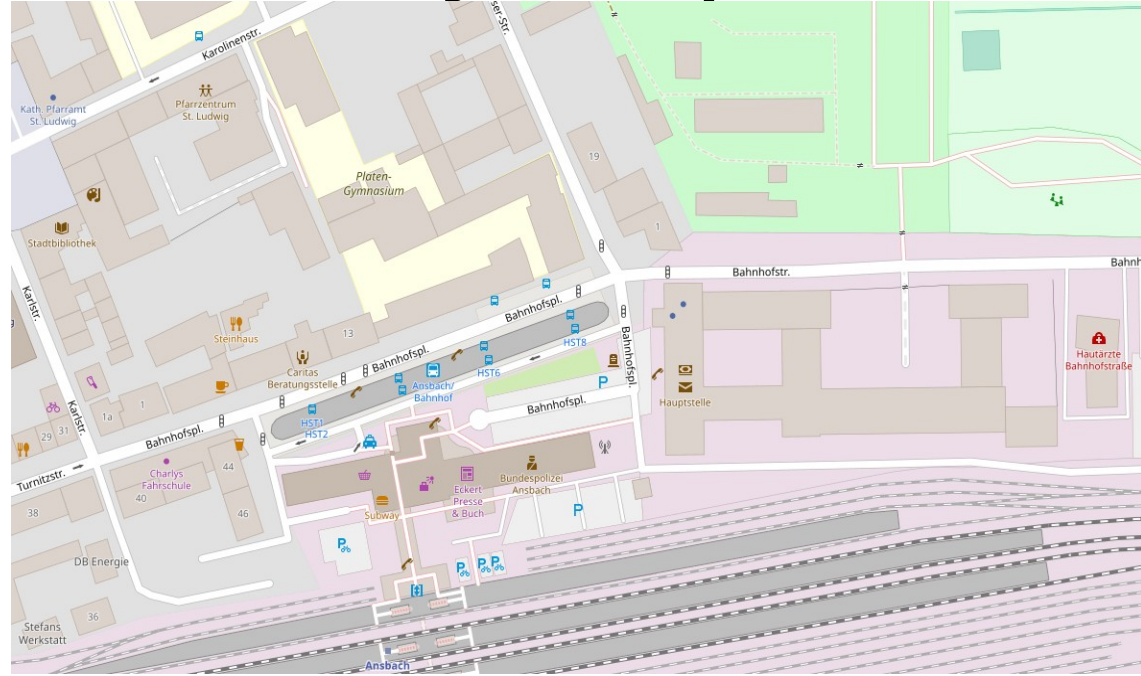
```
SELECT
tags -> 'name' AS name,
(ST_DistanceSphere(ST_GeomFromText('POINT(10.579172 49.298858)', 4326), ST_Transform(geom, 4326))/1000) AS "Distance in km",
degrees(ST_Azimuth(ST_GeomFromText('POINT(10.579172 49.298858)', 4326), geom)) AS Direction
FROM
nodes
WHERE
(ST_DistanceSphere(ST_GeomFromText('POINT(10.579172 49.298858)', 4326), ST_Transform(geom, 4326))/1000) < 2
and tags -> 'name' != ""
UNION ALL
SELECT
a.tags -> 'name' AS name,
(ST_DistanceSphere(ST_GeomFromText('POINT(10.579172 49.298858)', 4326), ST_Transform(b.geom, 4326))/1000) AS "Distance in km",
degrees(ST_Azimuth(ST_GeomFromText('POINT(10.579172 49.298858)', 4326), b.geom)) AS Direction
FROM
ways a, nodes b
WHERE
b.id = a.nodes[1]
and (ST_DistanceSphere(ST_GeomFromText('POINT(10.579172 49.298858)', 4326), ST_Transform(b.geom, 4326))/1000) < 2
and a.tags -> 'name' != ""
ORDER BY 2 ASC;
```

Name Tag can be on a way or on a node!

name	Distance in km	direction
Hauptstelle	0.00470371505	345.351163212131
Packstation 117	0.01174393515	302.981744768006
Bahnhofplatz	0.02204542948	276.983370705694
IHK-Geschäftsstelle Ansbach	0.03274358952	344.549333320552
Luitpold-Brunnen	0.03373878872	288.434948823069
Tradebyte Software GmbH	0.03676775318	356.518681624046
HST8	0.04926204787	287.981579218787
Ansbach - Bahnhof	0.05008066041	17.7342975077383
HST8	0.05289000007	291.269486388738
Bahnhofstraße	0.05503531369	318.259401551632
Ansbach	0.05571847052	184.25073149902
HST7	0.05756002688	294.130454197957
Gewerbegebiet Bahnhofstraße	0.06461409125	3.59722259643559
HST12	0.06684025068	294.208593141357
Bundespolizei Ansbach	0.06767025403	255.725206216893
HST6	0.07909885029	274.829383013845
HST11	0.080645623	285.155475611995
HST5	0.08471131238	279.172036790627
Bischof-Meiser-Straße	0.08895764229	319.496065423954

Whats with these fancy maps?

Maps are generated from selected nodes (and height information from other free sources like SRTM (Nasa Shuttle Radar Topography Mission)).
So every map is different!



Commands used...

(Ubuntu 18.04)

```
sudo apt-get install postgresql postgis osmosis
sudo -u postgres createuser -U postgres -d -e -E -l -P -r -s gisuser
psql -U gisuser -h 127.0.0.1 postgres
    create database gisdb;
    \connect gisdb
    create extension postgis;
    create extension hstore;
    \q
```

```
psql -U gisuser -h 127.0.0.1 gisdb <
/usr/share/doc/osmosis/examples/pgsnapshot_schema_0.6.sql
```

```
wget http://download.geofabrik.de/europe/germany/bayern-latest.osm.pbf
```

```
osmosis --read-pbf bayern-latest.osm.pbf --log-progress --write-pgsql database=gisdb
    host=127.0.0.1 user=gisuser password=gispass
    -> Patience! (~30 minutes)
```

```
psql -U gisuser -h 127.0.0.1 gisdb
```

You're done!

Thank you!