

## Nominatim on your laptop

Sarah Hoffmann  
lonvia@denofr.de

12 Nov 2023

SotM-EU  
Antwerp

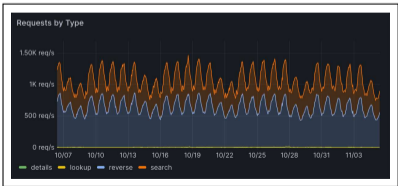
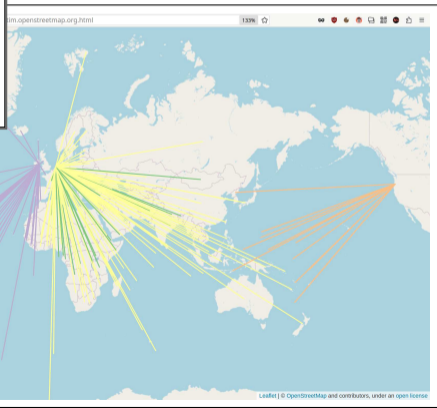
# The Nominatim Geocoder

The screenshot shows the OpenStreetMap website interface. The search bar at the top left contains the text "Antwerp". Below the search bar, the "Search Results" panel is open, displaying the following information:

- Search Results** (with a close button)
- Results from **OpenStreetMap Nominatim**
- City **Antwerp, Flanders, Belgium**
- State Boundary **Antwerp, Flanders, Belgium**
- County Boundary **Antwerp, Flanders, Belgium**
- Village Boundary **Antwerp, Flanders, Belgium**
- A button labeled "More results"

The main map area shows a satellite-style view of Antwerp, Belgium, with various roads, water bodies, and green spaces. The city name "Antwerpen" is prominently displayed in the center of the map. The browser's address bar shows the URL: `https://www.openstreetmap.org/search?query=Antwerp#map=11/51.2608/4.3575`. The browser's address bar also shows the page is zoomed in at 133%.

# nominatim.openstreetmap.org: the Big Installation



## Hwo About Tiny Nominatim?



(C) Raimond Spekking



(C) Mytho88

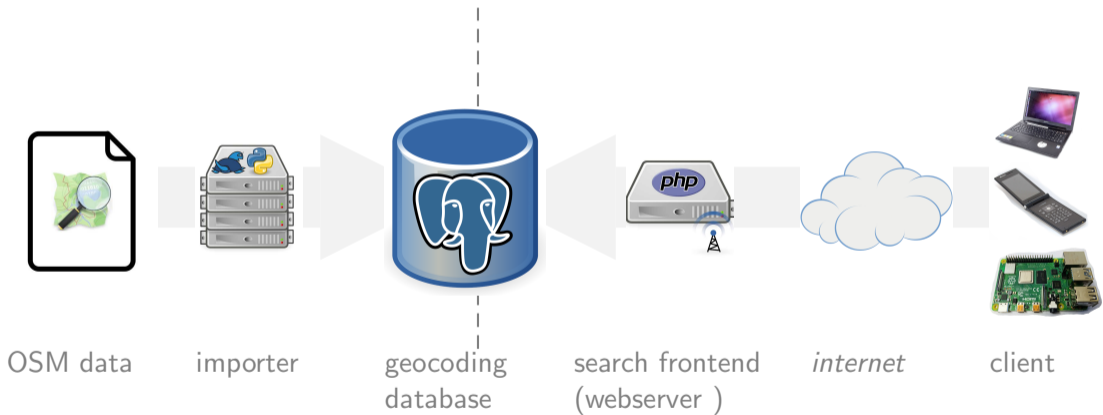


(C) Suyash.dwivedi

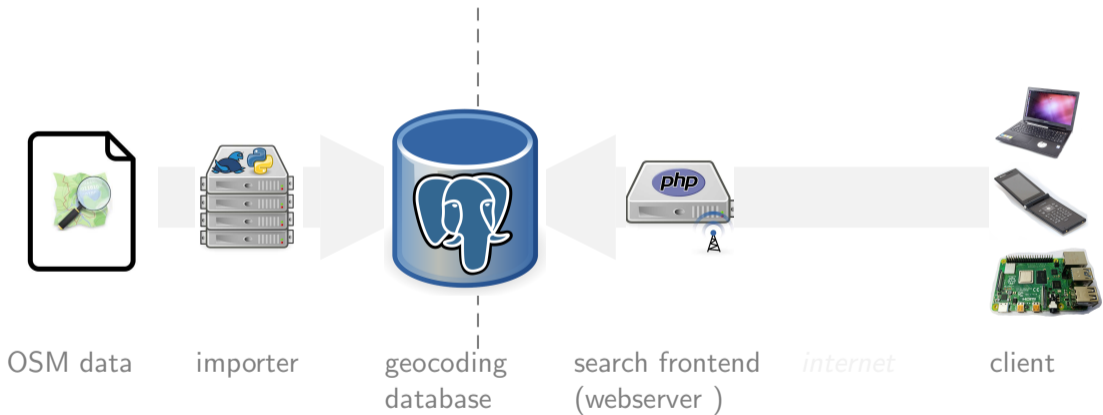
with small extracts

- faster than public API
- better privacy
- customizable

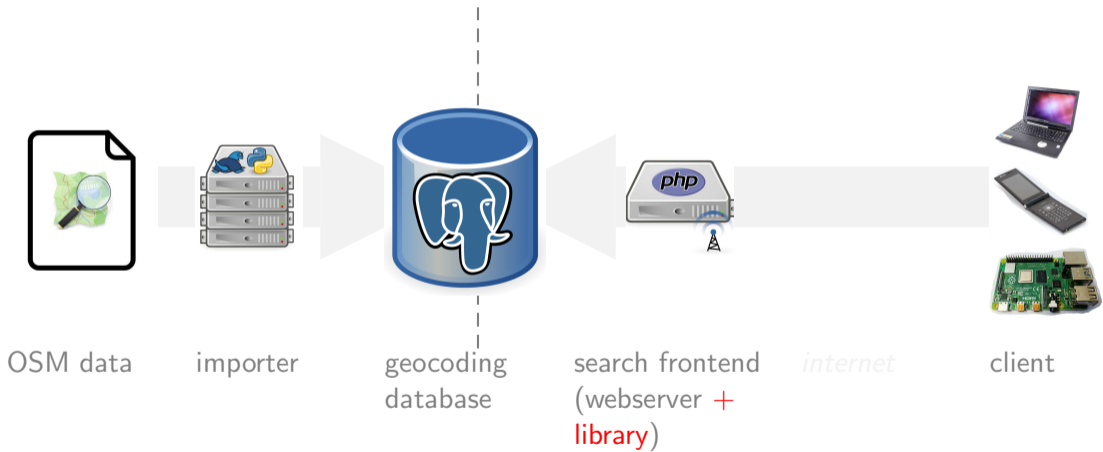
# The Geocoding Pipeline



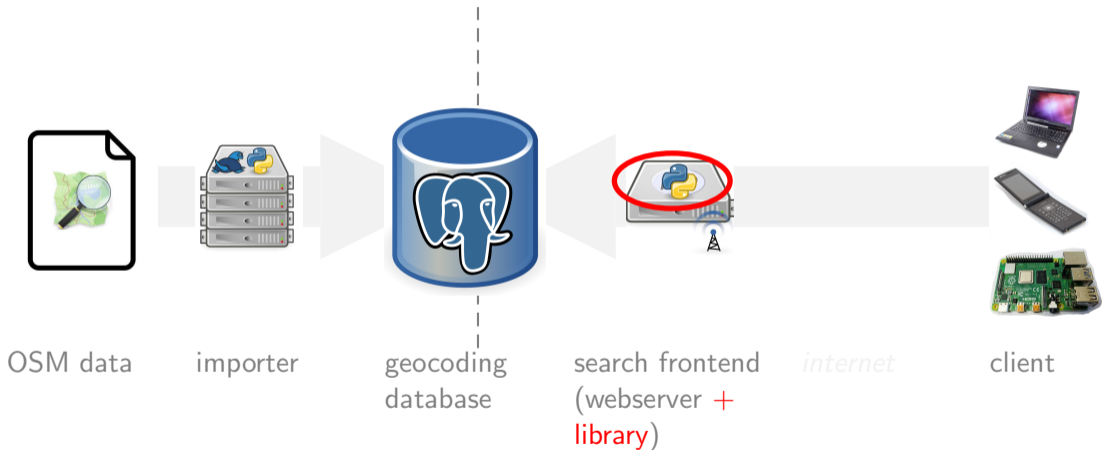
# The Geocoding Pipeline



# The Geocoding Pipeline

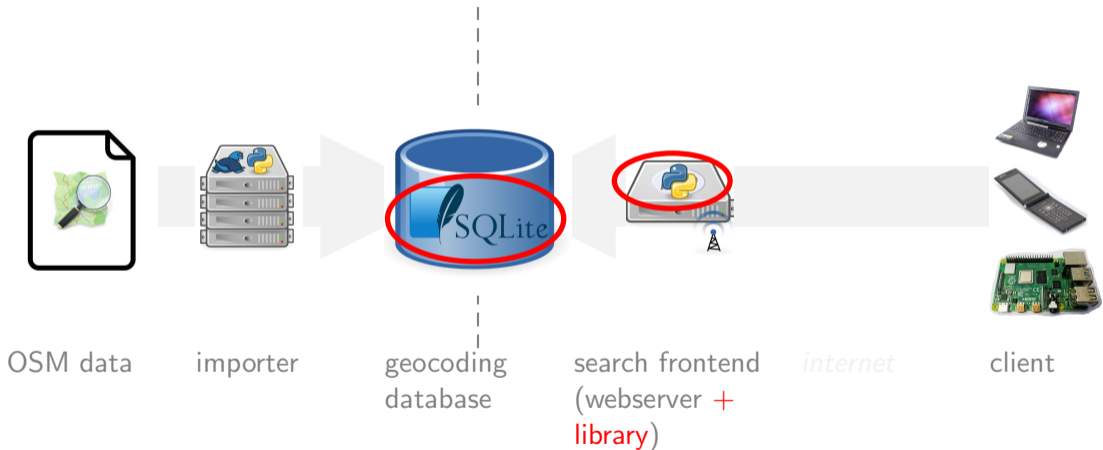


# The Geocoding Pipeline





# The Geocoding Pipeline



# From PHP to Python

## Why Python?

- ✓ favoured by users of Nominatim (together with Java)
- ✓ rich library support for language processing (and SQLAlchemy)
- ✓ easy to create user-extendible code
- ✗ language lock-in
- ✗ mobile support
- ✗ performance

# SQLAlchemy

```
<?php
$$SQL = 'SELECT place_id FROM';
$$SQL .= ' (SELECT place_id, rank_search,';
$$SQL .= '         ST_distance('.$sPointSQL;
$$SQL .= '         ,geometry) as dist';
$$SQL .= ' FROM placex';
$$SQL .= ' WHERE osm_type = \'N\'';
$$SQL .= ' AND country_code = \''.$sCountryCode.'\'';
// needed to select right index
$$SQL .= ' AND rank_address between 4 and 25';
$$SQL .= ' AND rank_search between 5 and ' ;
$$SQL .= '         min(25, $iMaxRank);
$$SQL .= ' AND type != \'postcode\'';
$$SQL .= ' AND name IS NOT NULL ' ;
$$SQL .= ' AND ST_Buffer(geometry, rpd(rank_search))';
$$SQL .= '         && '.$sPointSQL;
$$SQL .= ') as a ' ;
$$SQL .= 'WHERE dist <= rpd(rank_search)';
$$SQL .= ' ORDER BY rank_search DESC, dist ASC';
$$SQL .= ' LIMIT 1';
```

```
import sqlalchemy as sa

t = tables.placex
inner = sa.select(t.c.place_id, t.c.rank_search,
                  t.c.geometry.ST_Distance(pt)
                  .label('dist'))\
        .where(t.c.osm_type == 'N')\
        .where(t.c.country_code == cc)\
        .where(t.c.rank_address.between(4, 25))\
        .where(t.c.rank_search.between(5, min(maxRank, 25)))\
        .where(t.c.type != 'postcode')\
        .where(t.c.name != None)\
        .where(t.c.geometry.ST_Buffer(
            sa.func.rpd(t.c.rank_search)
            .intersects(pt)))

        .alias('a')

sql = sa.select(inner.c.place_id)\
        .where(inner.c.dist<=sa.func.rpd(inner.c.rank_search))\
        .order_by(inner.c.rank_search.desc())\
        .order_by(inner.c.dist)\
        .limit(1)
```

**Supports database adaption for PostgreSQL, SQLite and others.**

## Performance: PHP vs. Python

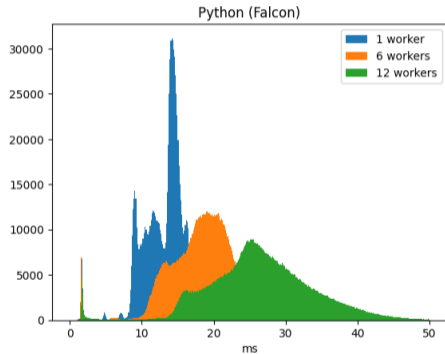
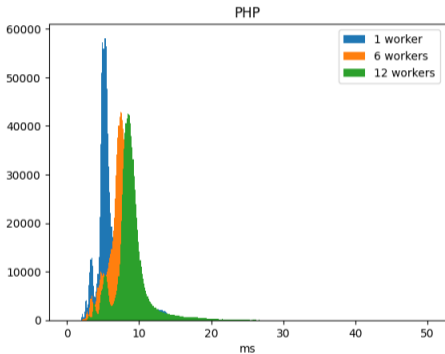
API call / request duration(*)	PHP	Python(1st try)	Python(master)
reverse geocoding	6.3 ms	13.8 ms	8.9 ms
forward search	28.1 ms	37.0 ms	32.7 ms

Unequal conditions:

- SQL string concatenation (PHP) vs. SQL library (Python)
- search algorithm not the same

(\*) *average over 10.000 real-world requests as seen on osm.org*

# Performance: Multitasking



# The Nominatim Python Library

since 4.3 release (experimental)

## A simple example: Tagging your photo with locations

```
from pathlib import Path
import exifread
from imgtag import ImgTag
import nominatim.api as napi

with open(FILENAME, "rb") as fh:
    y, x = exifread.utils.get_gps_coords(
        exifread.process_file(fh, details=False))

api = napi.NominatimAPI(Path('.'))
result = api.reverse((x, y), zoom=25, address_details=True)

address_parts = result.address_rows.localize(napi.Locales(['en']))

tags = ImgTag(filename=FILENAME)
tags.add_tags(address_parts)
tags.close()
```



## Different Kinds of Search

- free-text search

```
search('BluePoint, Antwerp'))
```

- structured search

```
search\_address(city='Berchem', street='Filip Williotstraat 9', postcode='2600')
```

- specific type of place near a named place

```
search\_category([('amenity', 'event\_venue']), near\_query='Antwerp')
```

- specific type of place near a location

```
search\_category([('amenity', 'event\_venue']), viewbox=(51.1, 4.4, 51.2, 4.5))
```

## And all other API functions

- point to address

```
reverse((x, y))
```

- OSM object to address

```
lookup([OsmID('N', 3452), OsmID('W', 555)])
```


- Dump full information of place

```
details(PlaceID(2345756))
```

... with many filters and constraints

# Rich Result Information

← → ↻  110% ☆

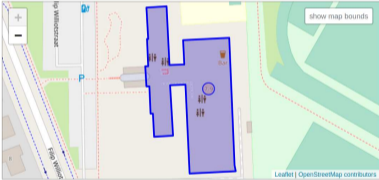
 **Nominatim** Search Reverse Search By ID About & Help

OSM type=id (W123, n123, M123, w123, R123, r123), Place id (1234) or URL (<https://openstreetmap.org/way/123>)

Data from API request (Debug output) Data last updated: 1 minute ago (Details)

## BluePoint Antwerpen [link to this page](#)

Name	<b>BluePoint Antwerpen</b> (name)
Type	amenity:events_venue
Last Updated	2023-10-04T17:20:06.226723+00:00
Search Rank	30
Address Rank	30 (house / building)
Importance	0.000009999999999995449
Coverage	Polygon
Centre Point (lat,lon)	51.186902,4.436224153643256
OSM	<a href="#">way 184336592</a>
Place Id	368228868 (on this server)
Computed Postcode	2600 ( <a href="#">how?</a> )
Address Tags	<b>Berchem</b> (city) <b>Filip Williotstraat</b> (street) <b>2600</b> (postcode) <b>7-9</b> (house:numero)
Extra Tags	<a href="https://www.bluepoint.be/nl/antwerpen/">https://www.bluepoint.be/nl/antwerpen/</a> (website) <b>house</b> (building) <b>Business Center, voorheen ALM</b> (description) <b>Gbg/3668369</b> (source:geometry:ref) <b>2012-03-22</b> (source:geometry:date)



Leaflet | OpenStreetMap contributors

## Address

Local name	Type	OSM	Address rank	Admin level	Distance	
BluePoint Antwerpen	amenity:events_venue	<a href="#">way 184336592</a>	30		0	<a href="#">details</a>
7-9	place:house_number		28		0	
Filip Williotstraat	highways:service	<a href="#">way 4341759</a>	27		0	<a href="#">details</a>
Pulhof	place:neighbourhood	<a href="#">node 193889869</a>	22		-0.006	<a href="#">details</a>

# Rich Result Information

The screenshot shows the Nominatim web interface for the location 'BluePoint Antwerpen'. The page displays various metadata such as Name, Type, Last Updated, Search Rank, Address Rank, Importance, Coverage, Centre Point, OSM ID, Place Id, Computed Postcode, Address Tags, and Extra Tags. Below this, the 'Address' section is visible, showing a table of address components.

Local name	Type
BluePoint Antwerpen	amenity:events_venue
7-9	place:house_number
Filip Willotstraat	highway:service
Pulhof	place:neighbourhood

```
>>> result = api.search('BluePoint, Antwerp', address_details=True)
>>> pprint.PrettyPrinter(depth=4).pprint(result[0].address_rows)
...
AddressLine(category=('boundary', 'administrative'),
             names={'ISO3166-2': 'BE-VAN',
                   'name': 'Antwerpen',
                   'name:ca': 'Anvers',
                   ...
                   'old_name:de': 'Antorf'},
             fromarea=True,
             isaddress=True,
             rank_address=8,
             distance=3.281105148737834e-06,
             place_id=84351529,
             osm_object=('R', 53114),
             extratags={'border_type': 'province',
                       'population': '1802719',
                       'population:date': '2014-01-01',
                       'ref:INS': '10000',
                       ...
                       'wikipedia': 'nl:Antwerpen (provincie)'},
             admin_level=6,
             local_name='Antwerpen'),
...
...
...
node 193889869 22 -0.006 details
```

# Nominatim inside SQLite

available on master

## PostgreSQL → SQLite

```
nominatim convert -o admin.sqlite
```

Extract / Database size (*)	PostgreSQL	SQLite
Germany	19 GB	18 GB
Admin	12 GB	12 GB

(\*) *non-updateable database, reverse-geocoding only*

## Using the SQLite database

```
me@home> echo 'NOMINATIM_DATABASE_DSN=sqlite:dbname=admin.sqlite' >> .env
me@home> python3 tag_photo_place.py
```

or

```
me@home> export NOMINATIM_DATABASE_DSN=sqlite:dbname=admin.sqlite
me@home> python3 tag_photo_place.py
```

or

```
...
api = nominatim.api.NominatimAPI(
    Path('.'),
    environ={'NOMINATIM_DATABASE_DSN': 'sqlite:dbname=admin.sqlite'})
...
```

## Performance

Extract / Requests per sec (*)	PostgreSQL	SQLite
Germany	454	23
Admin	149	21

*(\*) Random reverse coordinates near Germany/around the world.*



## What's next?

- SQLite forward search
- reducing database size
- improving indexes for faster queries
- GeoPy support?

# Thank you.

Nominatim Website: <https://nominatim.org>  
lonvia@denofr.de

Thanks to our supporters

