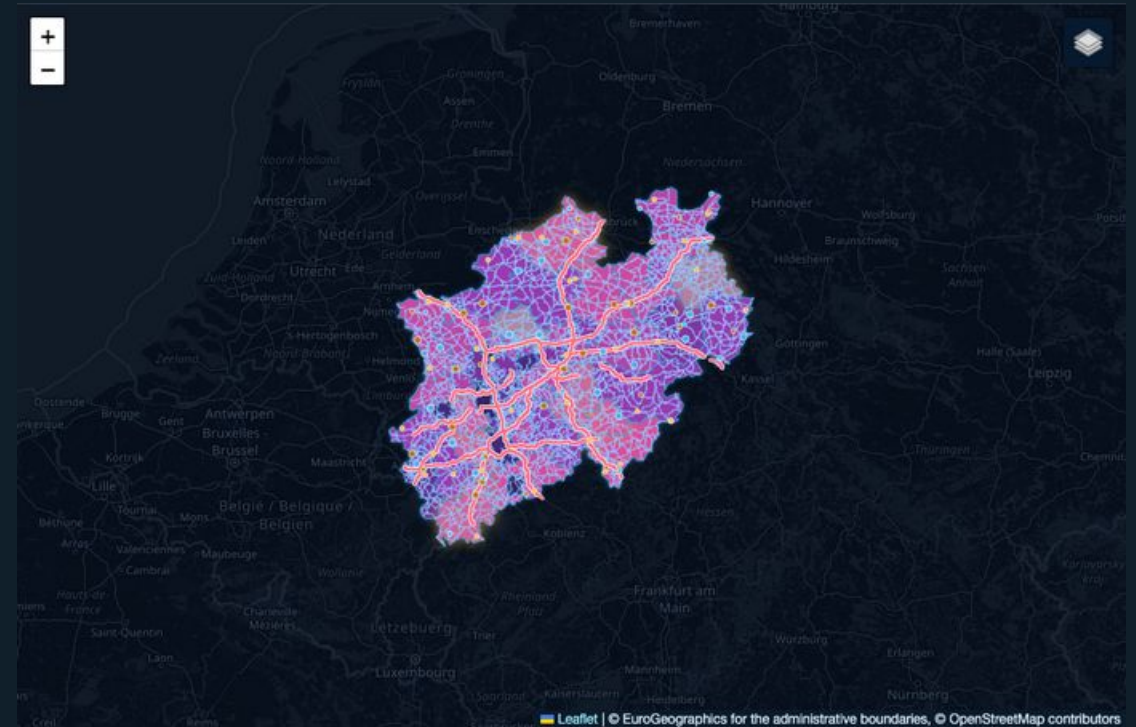


NRW Charging Atlas

A closer look at charging coverage
for electric vehicles (EV)

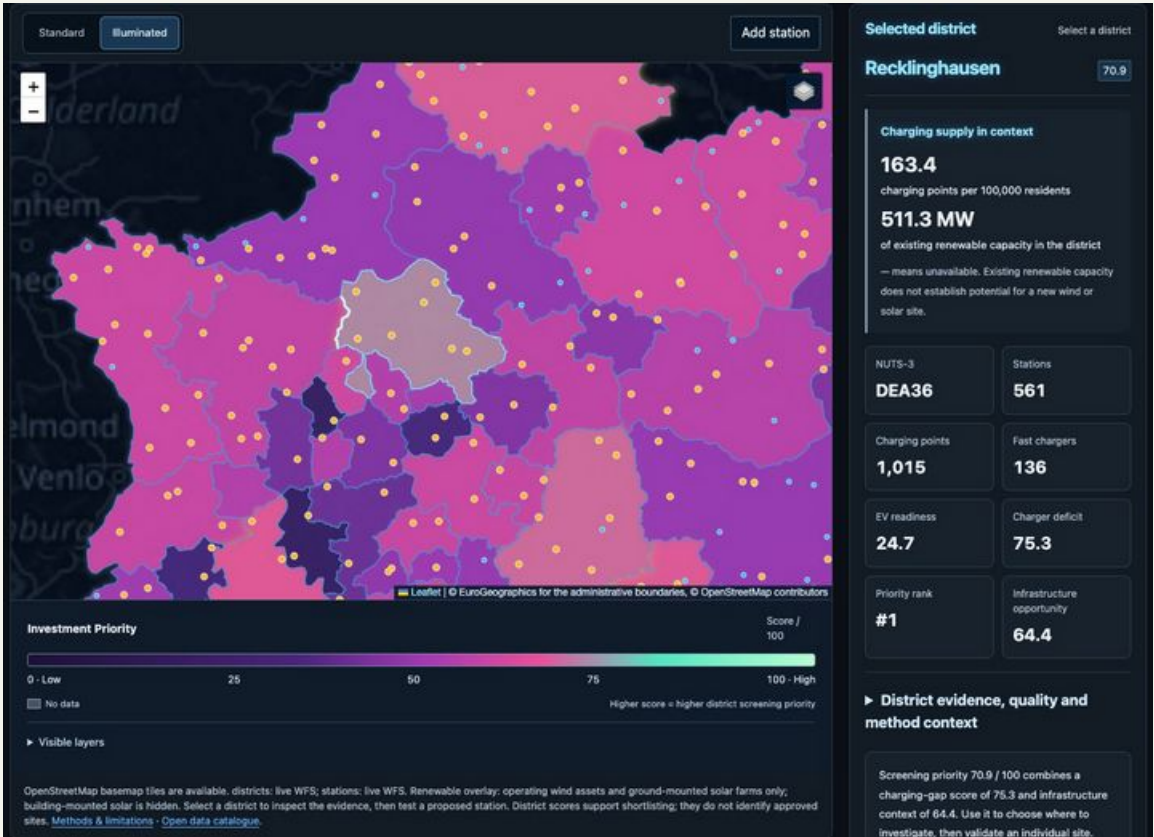
North Rhine-Westphalia (NRW), Germany



Map: © EuroGeographics, © OpenStreetMap contributors

Charging coverage across 53 districts

You can compare charging gaps and investment priority, then open a district to see why it ranks there.



What put this district here?

The evidence panel shows the measurements and weights behind its score.

Choosing a site still needs local checks, including grid capacity.

What goes into EV readiness?

40%

Charging points
per km²

30%

Distance to the
nearest station

30%

Charging points
per 100,000 people

I normalize at p5 and p95, then scale each component to 0-100.
Those baseline bounds stay fixed when you add a proposed station.

Distance uses the district centroid and a straight line. Shorter distances score higher.

What if you added a station here?

Add proposed charging station ×

Selected location: **51.440313, 7.668457**

Name

Example charging hub

Latitude

51,440313

Longitude

7,668457

Charging points

4

Total station power (kW)

300

Maximum point power (kW)

150

Cancel

Add and recalculate

You can propose a new station

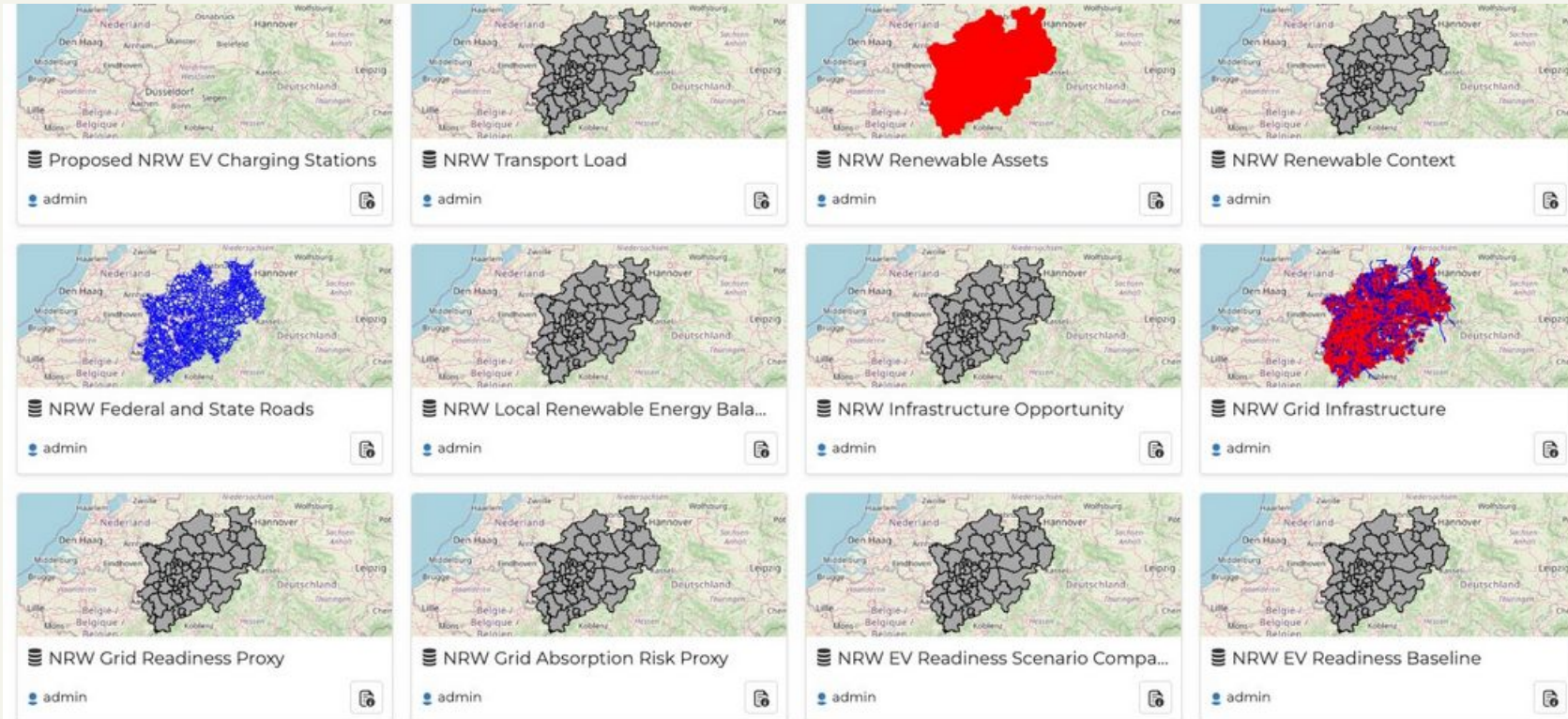
Pick a location, enter its charging-point count and power, then choose Add and recalculate.

Compare Current network, With proposals and Difference to see what changed.

Your proposals stay saved across restarts. The official charging records stay read-only.

The datasets have a home, too

You can browse the published layers and check their sources in GeoNode.



16 published datasets

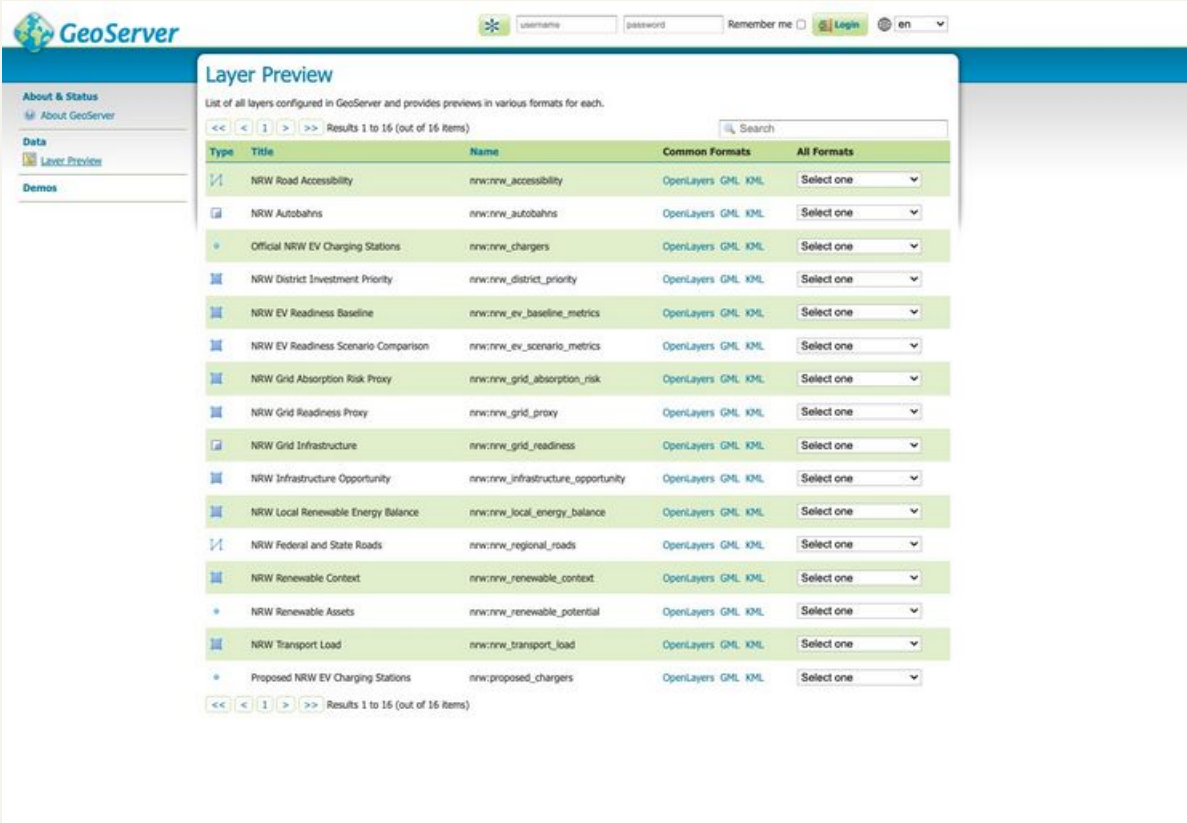
Open a layer to zoom and check its source.

The proposals layer starts empty.

localhost:8000/datasets#/

Dataset previews: © EuroGeographics, © OpenStreetMap contributors

You can inspect the published layers



WMS

Rendered map layers

WFS

Features and their attributes

The dashboard reads these services. WFS-T handles station proposals.

The maths lives in the database

PostGIS joins the public data and calculates the district scores.
The map reads the results, so it uses the same numbers as the database.

```
SELECT district_name,  
       ev_readiness_score,  
       investment_priority_score  
FROM publish.nrw_ev_baseline_metrics  
ORDER BY priority_rank;
```

You can query the published view directly to inspect the results.

Your own copy of the atlas

Git, Python 3.11+, uv and Docker Compose 2.24.4+

Give Docker at least 4 CPUs and 4 GiB RAM, with room left for your host.

```
git clone https://github.com/zhittsova/nrw-charging-atlas.git
```

```
cd nrw-charging-atlas
```

```
uv sync --frozen
```

```
uv run python -m scripts.project_stack bootstrap
```

Then open localhost:8081 and pick a district.

The first build and data import can take tens of minutes. Full setup is in the README.

github.com/zhittsova/nrw-charging-atlas