

FOSS4G-BE+NL

Algorithm for network planning and design in PostGIS

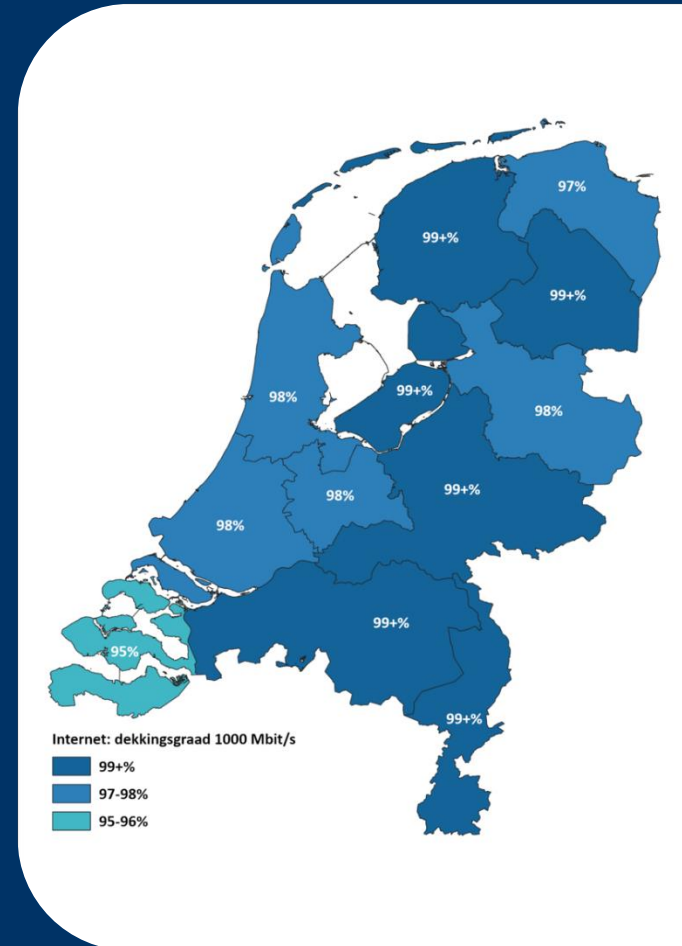
Wazir Sahebali

2024-09-26



Context

- High coverage of high-capacity fixed broadband (>1 Gbps) for households in the Netherlands
- Government aims for full (100%) coverage of >1 Gbps fixed internet by 2030
 - Remaining households often in regions that have been skipped by commercial parties for a reason:
 - Isolated
 - Too sparsely populated
 - No demand
 - ...



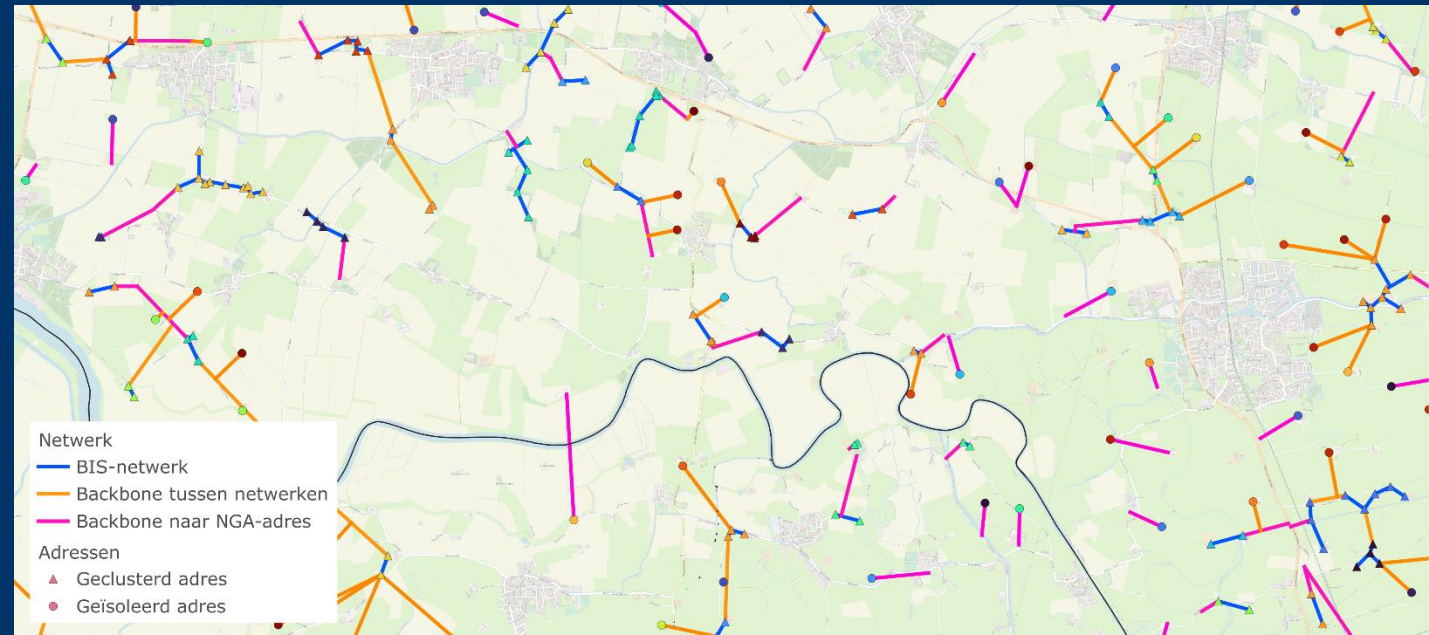
Project

- What would it cost to roll out fibre for these remaining household?
 - Major part of costs = distance
 - 30-50 €/m
 - So:

What is the distance required to connect the remaining households to a fibre network?

Solution I: incrementally expand

- Efficiently connect all remaining households to locations with high-speed connectivity.
- How?
 - WHILE households_remaining
 - Connect household closest to existing network
 - ST_SHORTESTLINE: draw line
 - LEFT JOIN LATERAL (ORDER BY a.geom<->b.geom): shortest pair



Solution I: incrementally expand

- Efficiently connect all remaining households to locations with high-speed connectivity.
- Issues:
 - Crossing obstacles is expensive (waterways, highways, railways, etc.)
 - As the crow flies $\neq$ real distance (and “ $\ast \sqrt{2}$ ” is a rough estimation)



Solution II

- How?
 - Connect short distance (<40 m) households using solution I
 - Connect other household using *more realistic connections*

Enter pgRouting + OSM

Follow the shortest route (A*) along the street.



<http://pgrouting.org/>



[https://github.com/
pgRouting/osm2pg
routing](https://github.com/pgRouting/osm2pgrouting)

[https://www.opens
treetmap.org](https://www.openstreetmap.org)

Solution II: Routing

- A. Find the household which is closest to the network
- B. Connect both points to the closest road-segment
- C. Find the shortest routes between the segments



Algorithm

1. Insert all households with high-speed internet into the network
2. Connect remaining households in a loop (while number of unconnected households > 0)
 - a. Short distance loop (while remaining households within 40m of the network)
 - i. Incrementally add straight line from the network to the remaining household closest to the network
 - b. Find unconnected household closest to network
 - c. Find part of network which is closest to that household
 - d. Connect both respectively to their closest road segment
 - e. Calculate the shortest route
 - f. Add these lines to the network
3. Calculate length of the network

Solution II: Routing



Fictional case - Dummy data

Code

- Postgis procedure/function

```
CALL network_builder_proc(  
    poi_schema => 'foss4g_20240926',  
    poi_table => 'no_fiber_set',  
    input_network_schema => 'foss4g_20240926',  
    input_network_table => 'fiber_set',  
    output_network_schema => 'foss4g_20240926',  
    output_network_table => 'network',  
    short_distance_limit => 20  
);
```

Applications

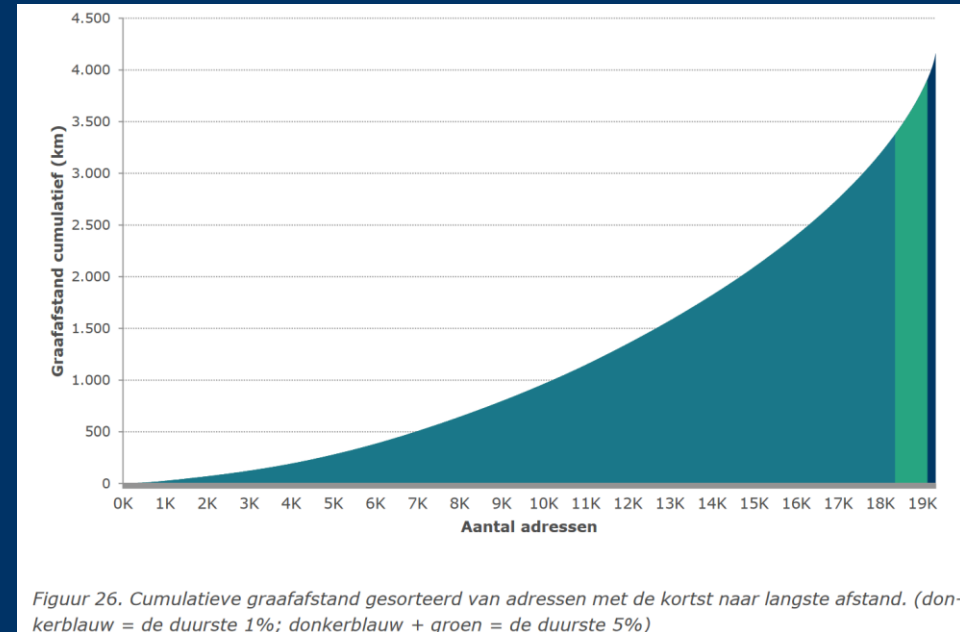
- Connecting remaining households
 - National case
 - Groningen case
- Business case smart city ring: Make or buy?
 - Different municipalities

dialogic

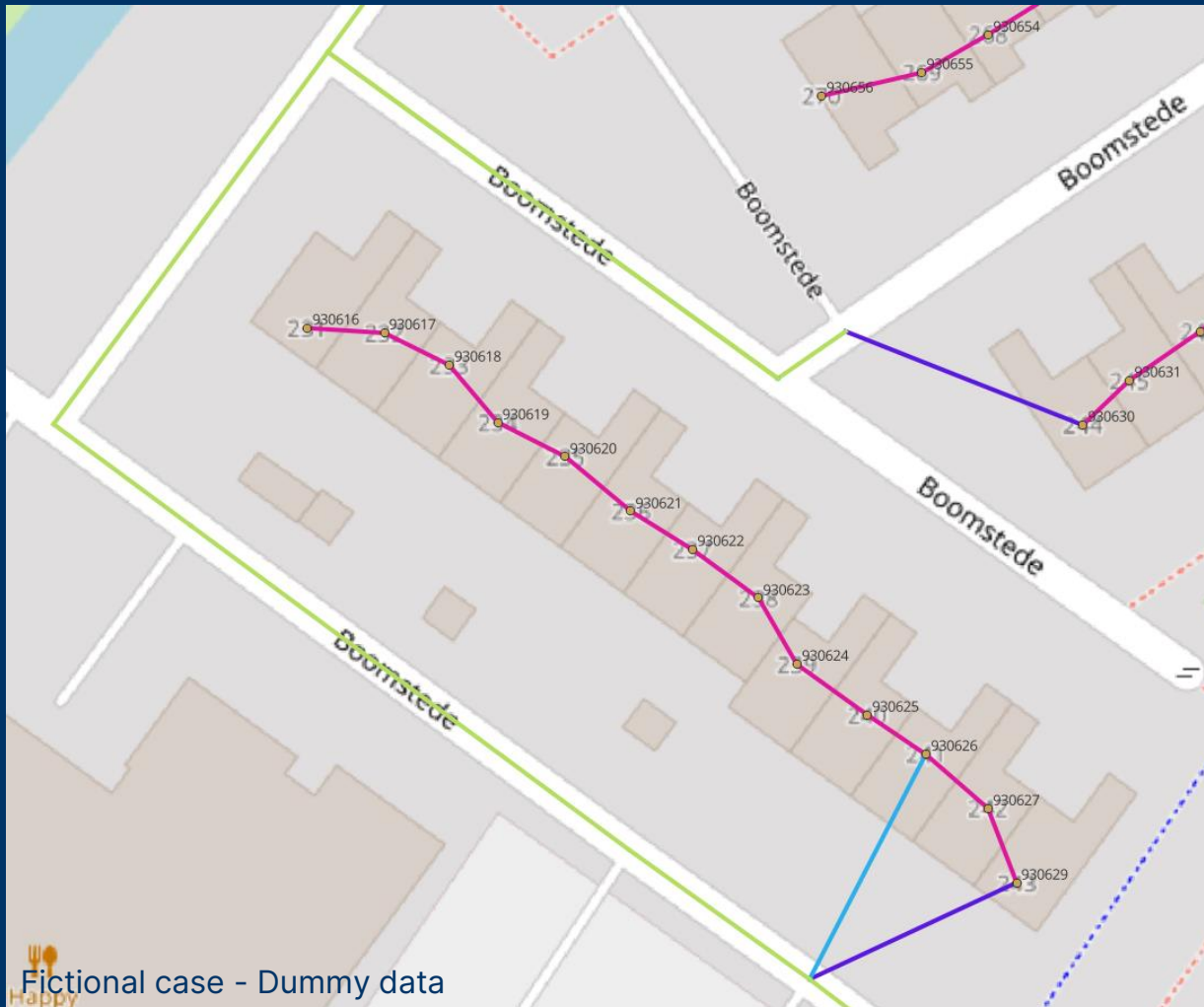


Results

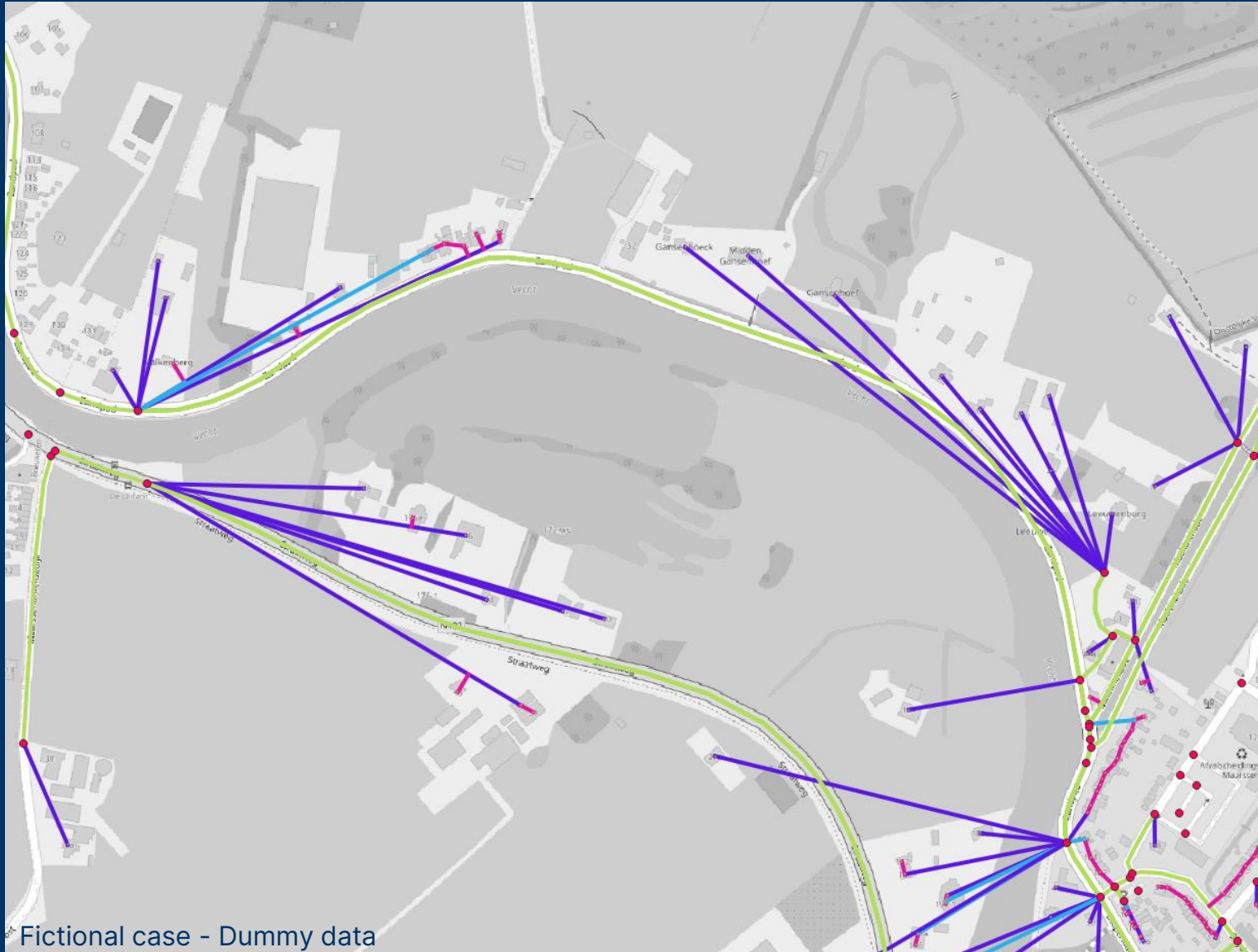
- **Visualization** of the network
- **Costs** of the total network
- **Order** of rolling out
- **Most expensive** routes/objects



Issues – Less ideal routes



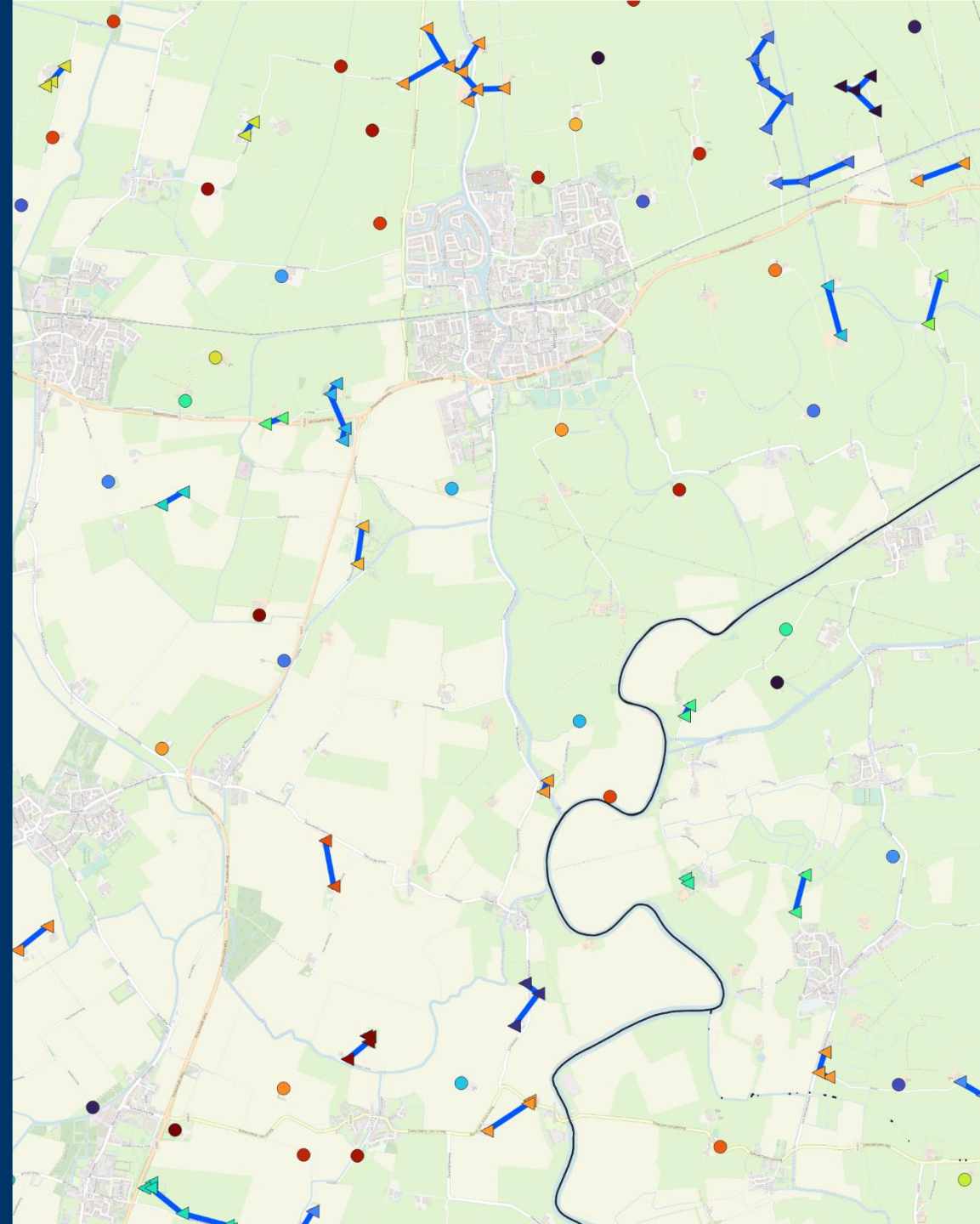
Issues – Too few ways-vertices



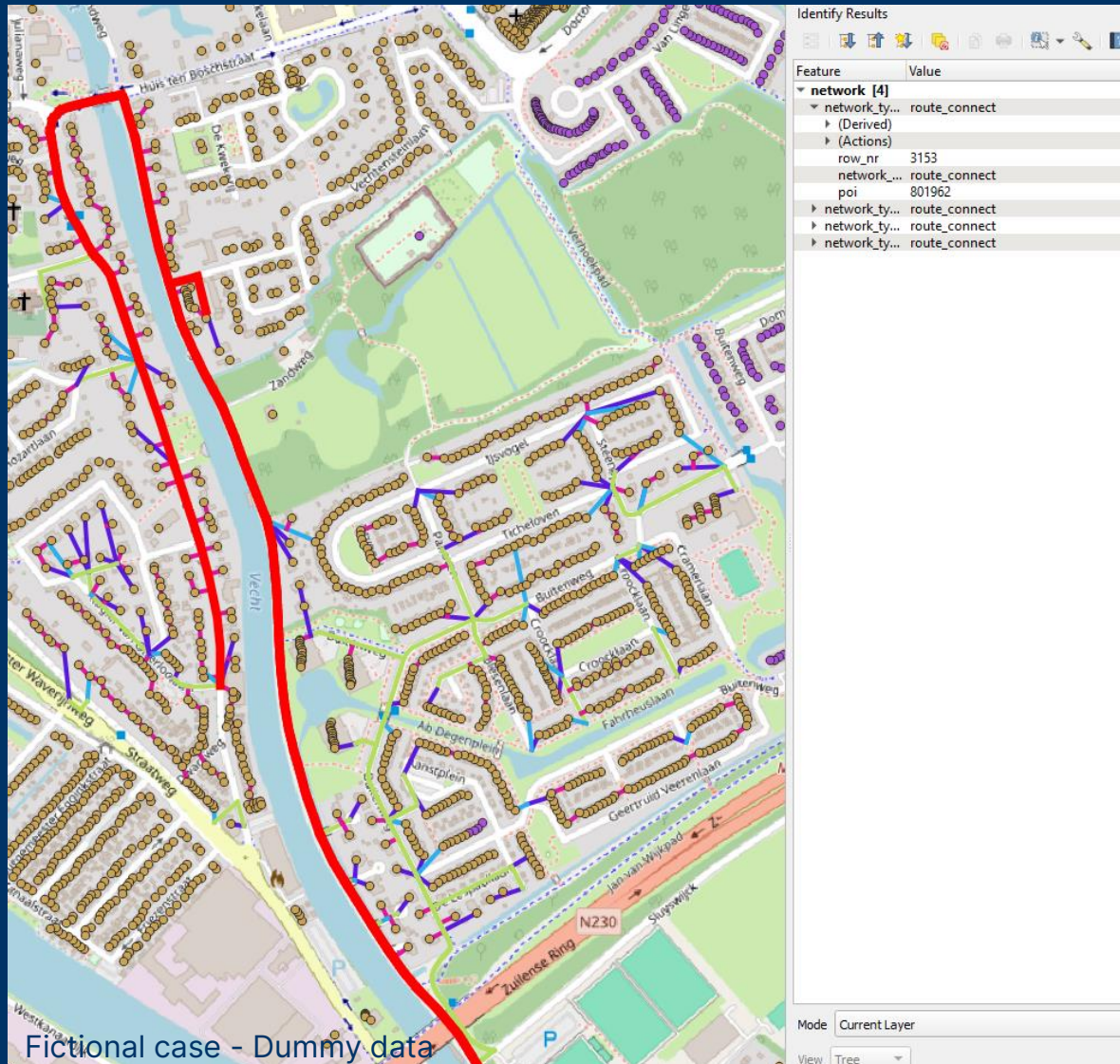
Fictional case - Dummy data

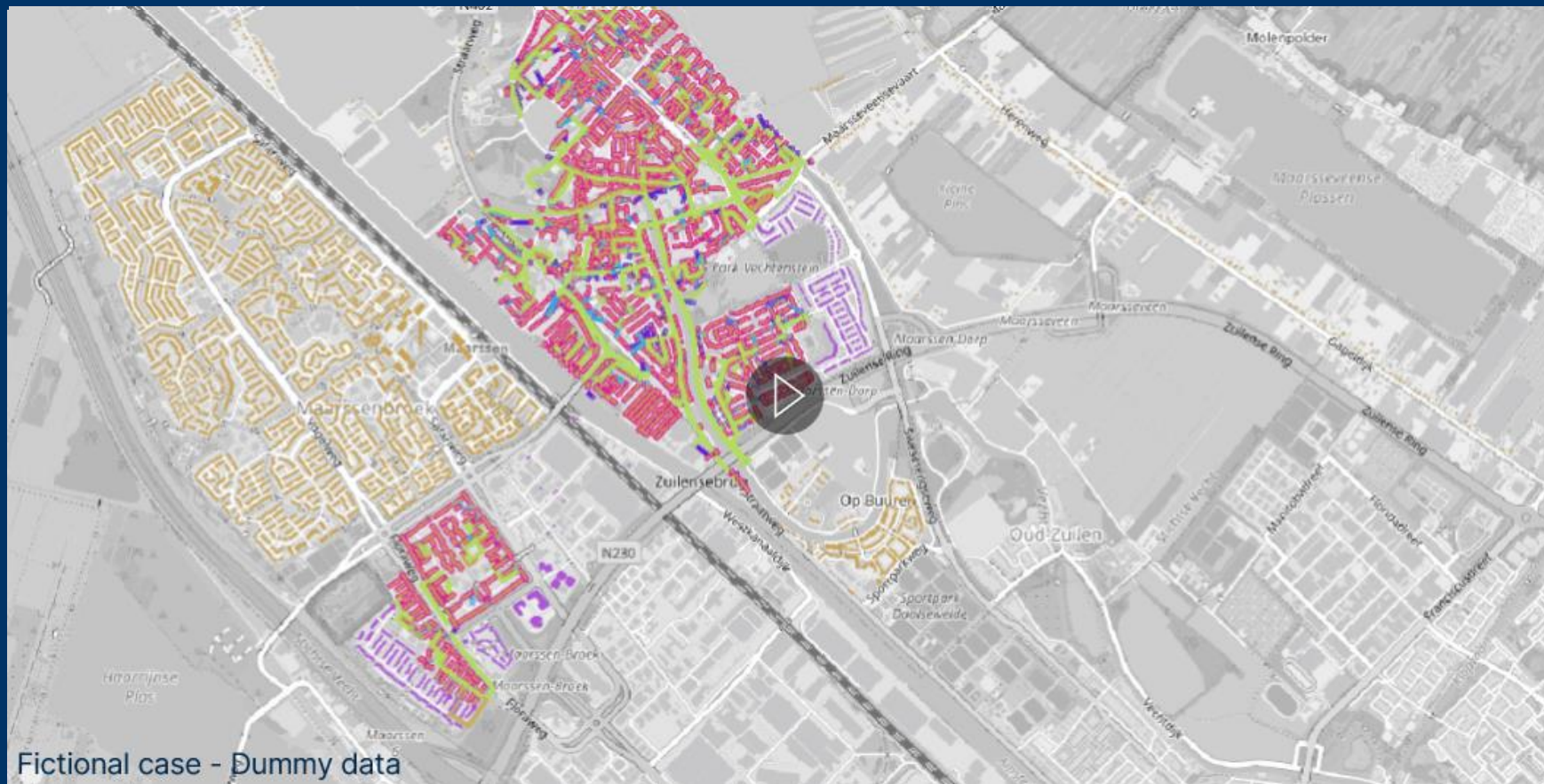
Issues – Slow performance

- Dense regions and incremental expansion slow down query
- Possible solutions:
 - Decrease dense input
 - We also developed a variant which:
 - Starts by clustering (ST_ClusterDBSCAN) dense regions
 - Efficiently connects the households within the cluster
 - Connects the cluster among each other by following the shortest street-route



Issues – Duplicate routes





Fictional case - Dummy data

dialogic

Wazir Sahebali (sahebali@dialogic.nl)

- Remaining households
- Households with fibre connectivity
- Short distance connections
- Routing routes