

# Large scale road network generalization for vario-scale map

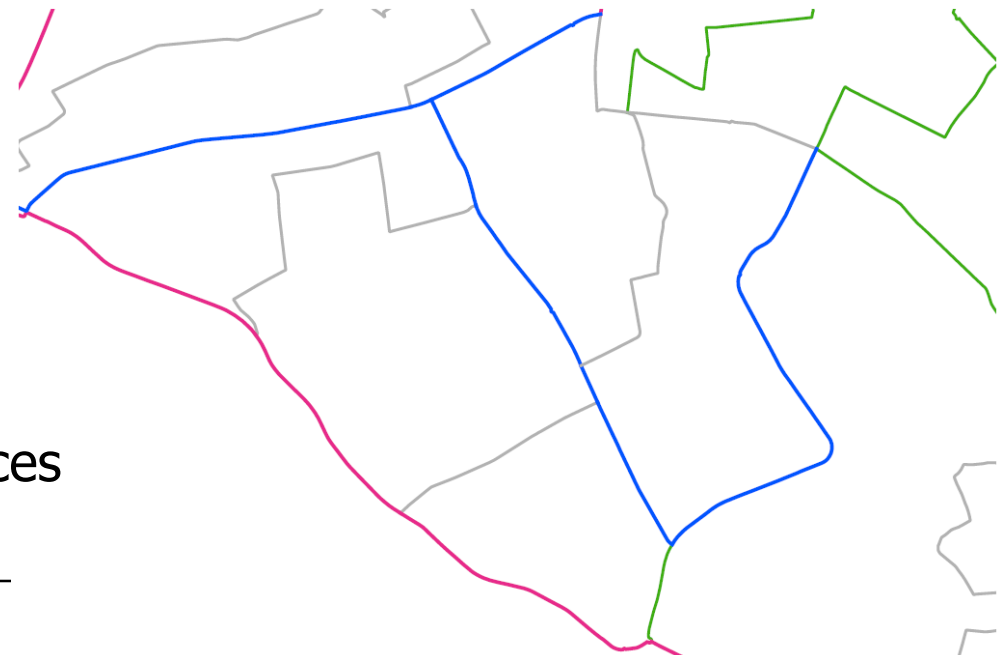
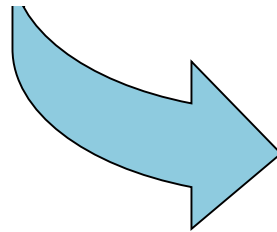
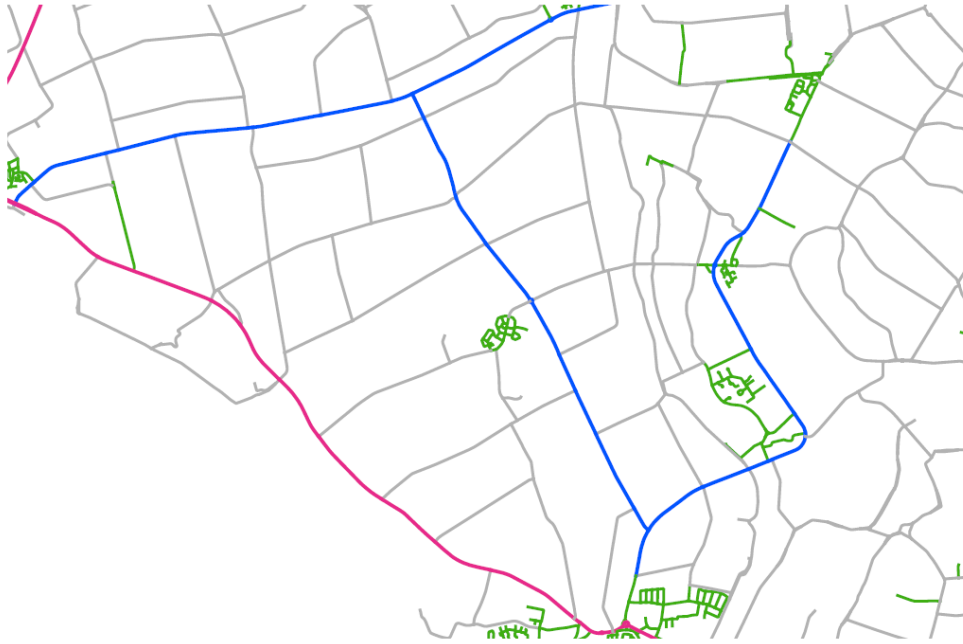
7-9-2015

Radan Šuba, Martijn Meijers, and Peter van Oosterom

18<sup>th</sup> ICA Workshop on Generalisation and Multiple Representation,  
21<sup>st</sup> August 2015, State University of Rio de Janeiro, Brazil



# Smaller scale road network generalization in tGAP (ICA 2014)

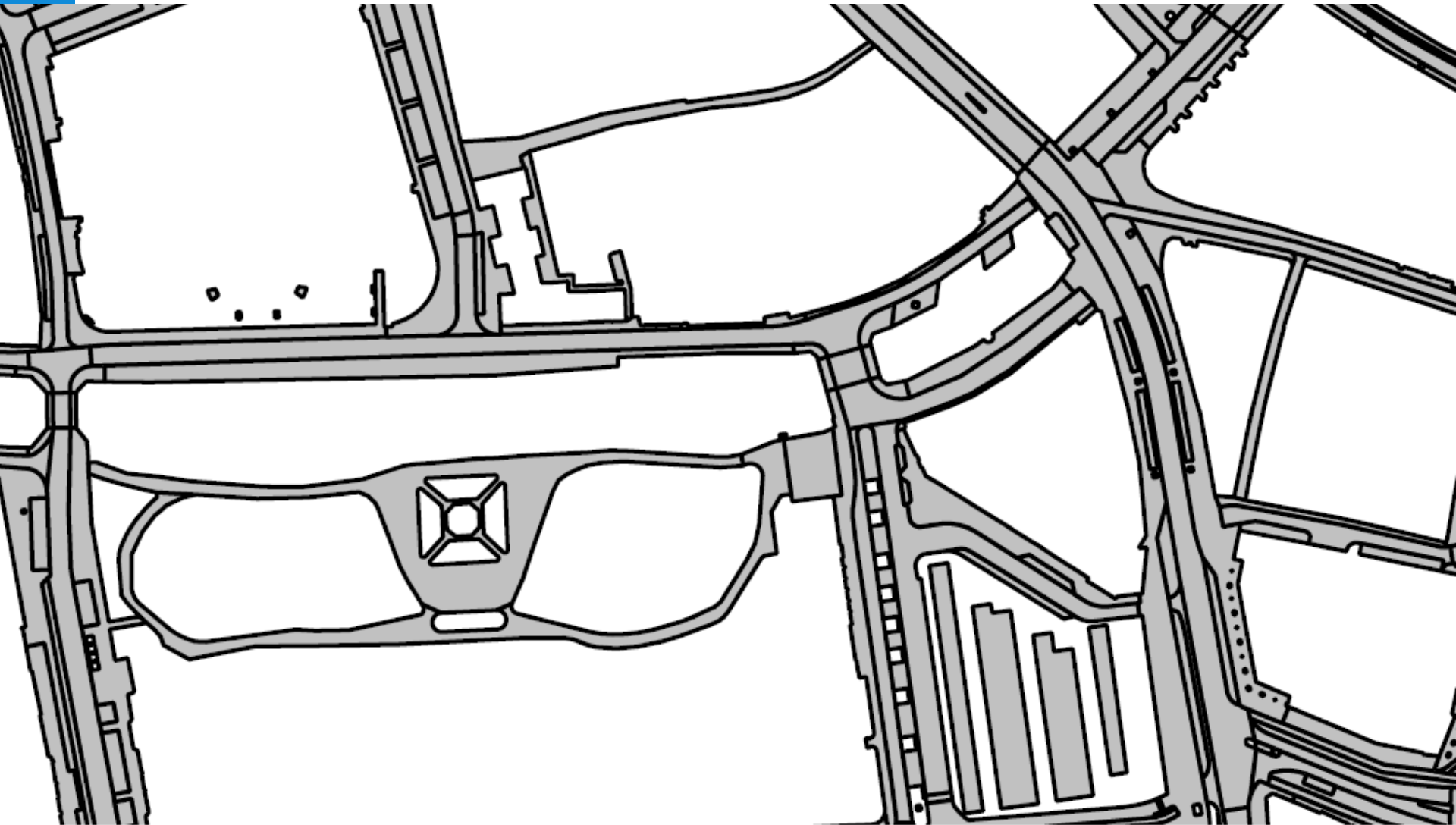


Roads are lines (edges in tGAP)

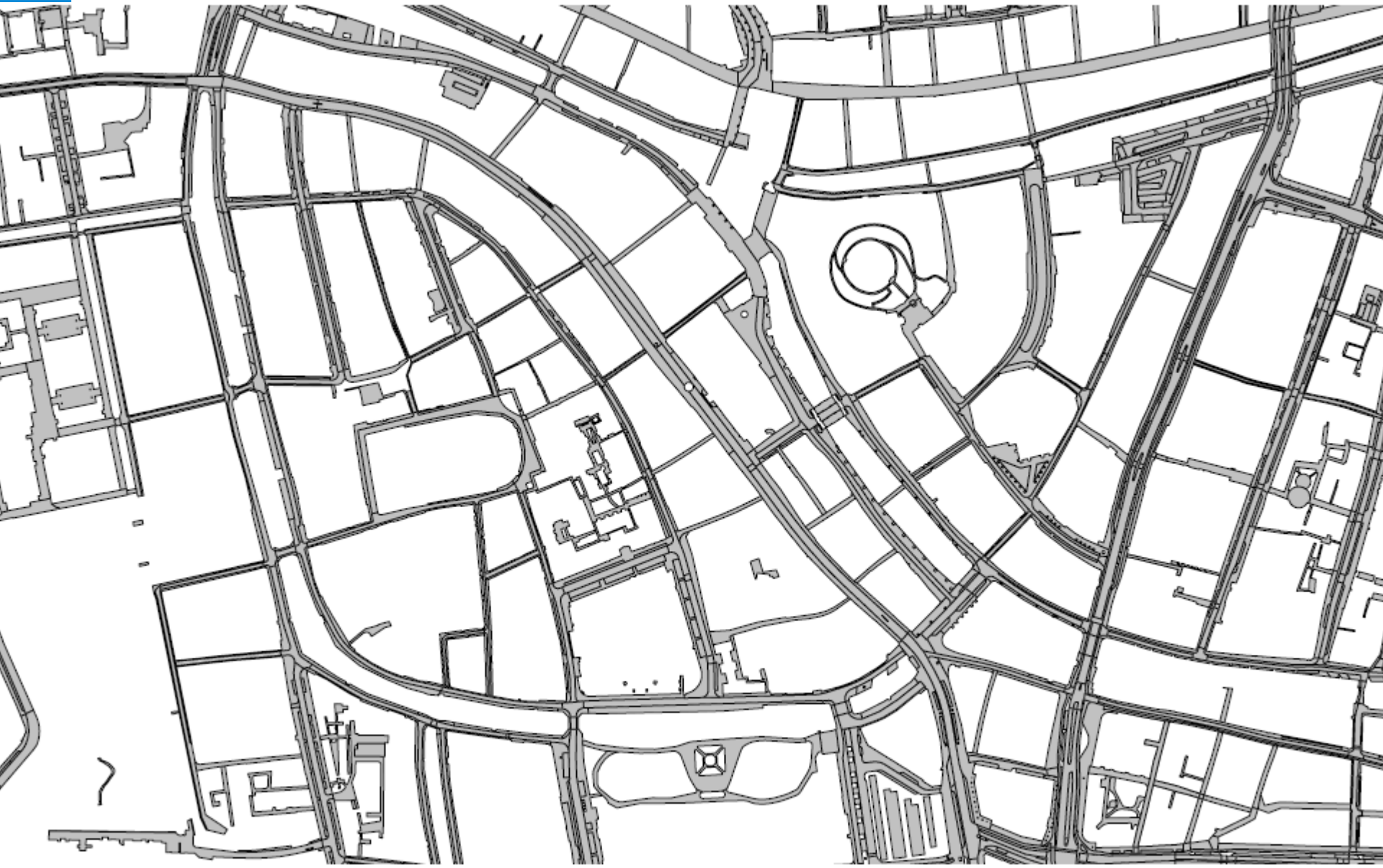
Main idea:

merge terrains (road cycles) faces  
remove shared edge (road)

Road as areas (from Dutch base map:  
to be used at 1:500 – 1:5,000)



# Base map, zoomed out



Roads as lines (NWB 1:25,000) → shock



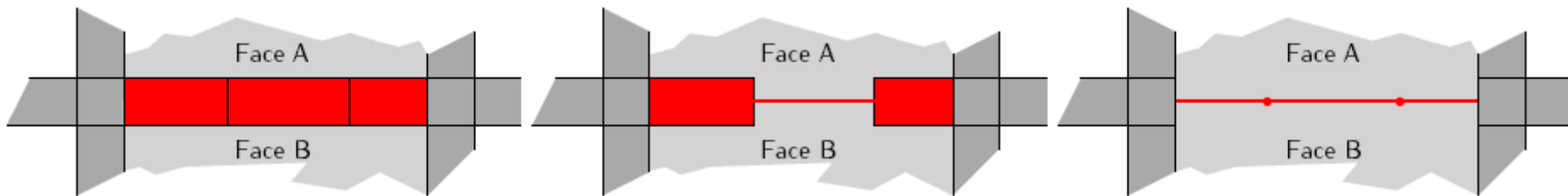
# Road network generalization: levels of granularity

Granularity indicates number of elements involved per step:

- *Coarse:* one or more complete classes is generalized;  
e.g. all highways
- *Medium:* group of features are generalized together;  
e.g. road consists of multiple parts
- *Fine:* individual part, object or road segment;  
e.g. segment between junctions (or different speed/material)

# Pro's and con's of granularity levels

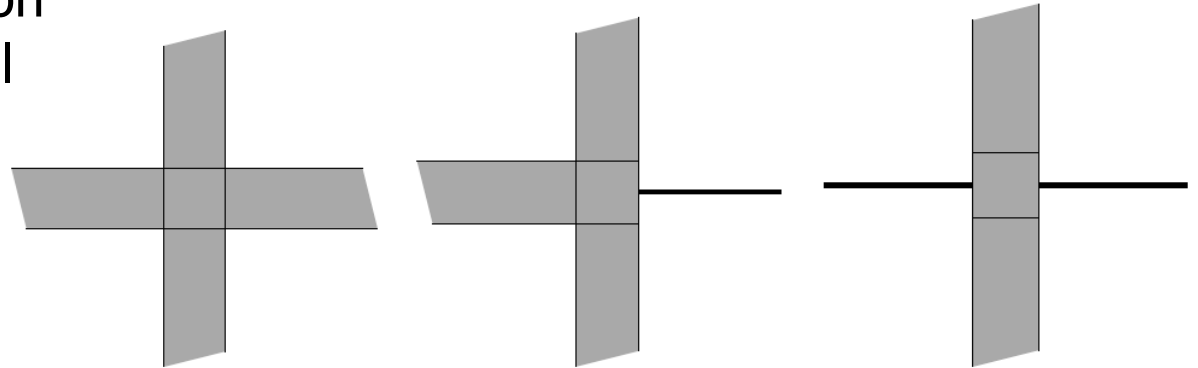
- Coarse/medium  
pro: easy to read  
con: content shocks, computational expensive, complex problem
- Fine  
pro: more gradual, easier to compute, history of steps, feature links  
con: can be 'disturbing'



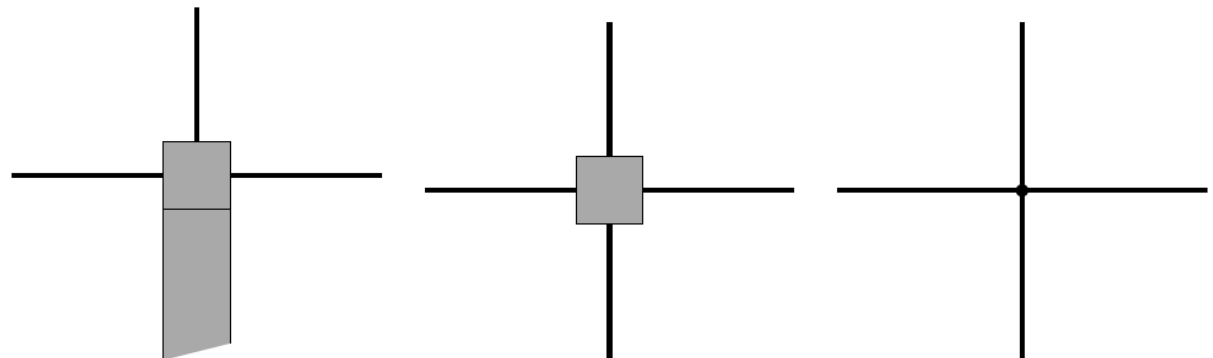


# Fine granularity, (no) overall recipe

- Assume: connection road class of equal importance and junction class of higher importance



- Normal tGAP algorithm → remove least important feature (collapse)



- tGAP with both areas and lines features

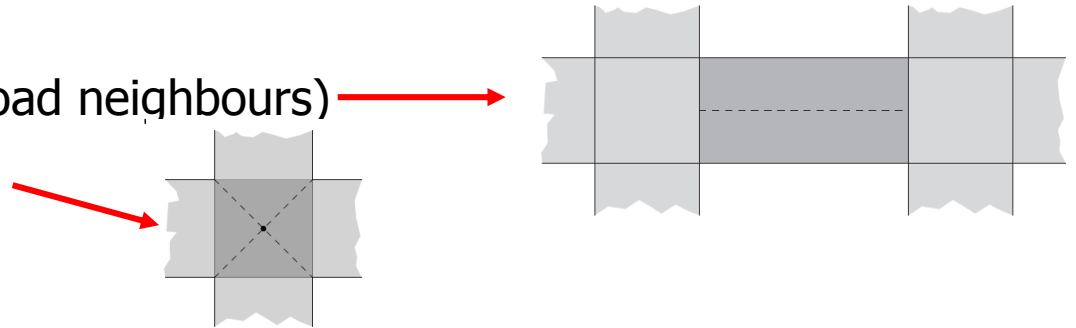
# tGAP creation (road network)

- Start with area partition consisting of 3 types of faces:

1. non-road

2. road connection (2 road neighbours) →

3. road junction (more) →

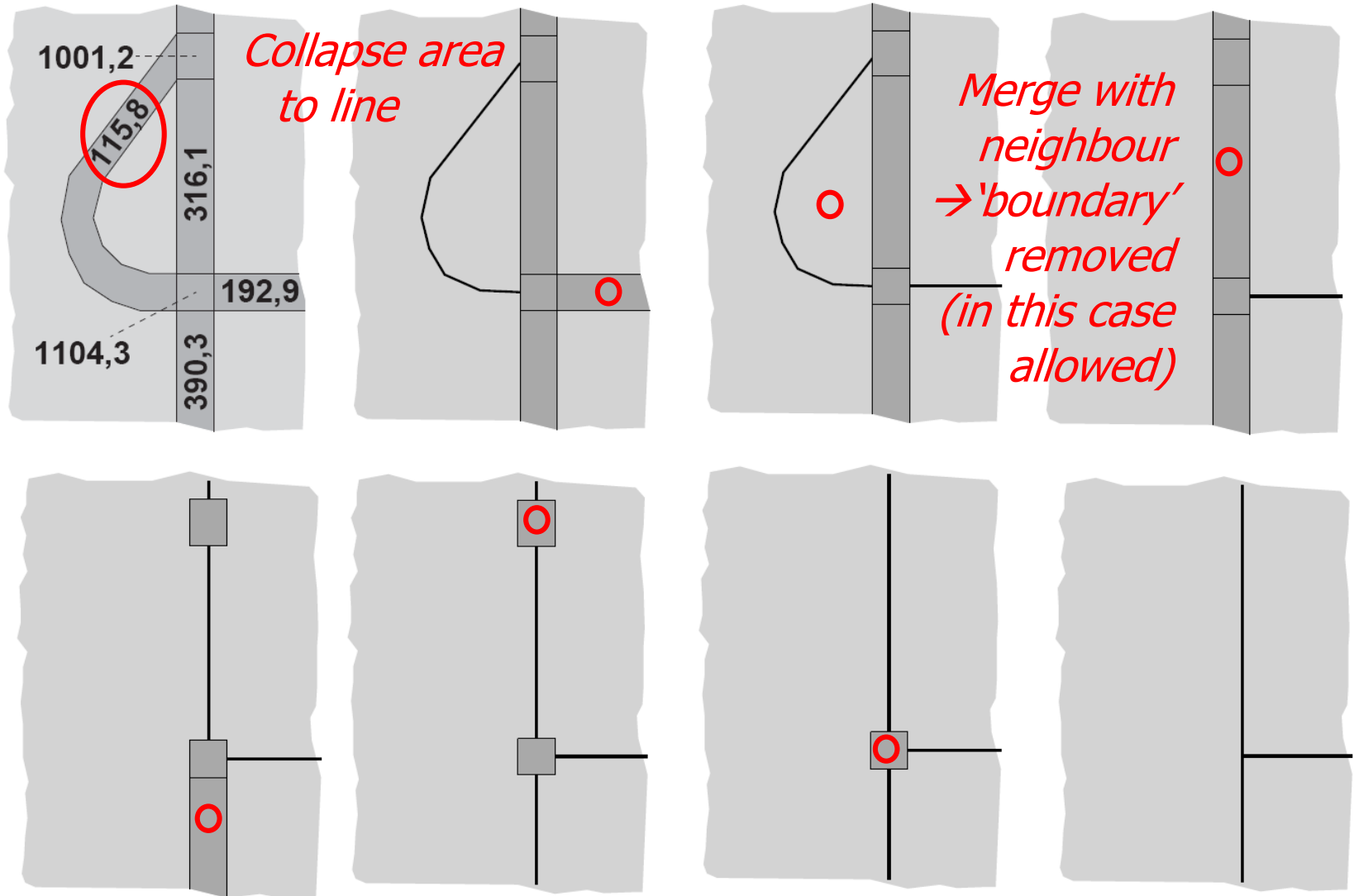


- Select least important face, then in case:

1. face is non-road, when possible merge with best adjacent non-road;  
When not possible, raise importance and back in queue
2. face is connection and there exists neighbour connection, then merge;  
Otherwise collapse road connection face to line
3. face is junction, only collapse if adjacent road connections are lines;  
When not the case, raise importance and back in queue

- Continue (at least) until road features are all lines

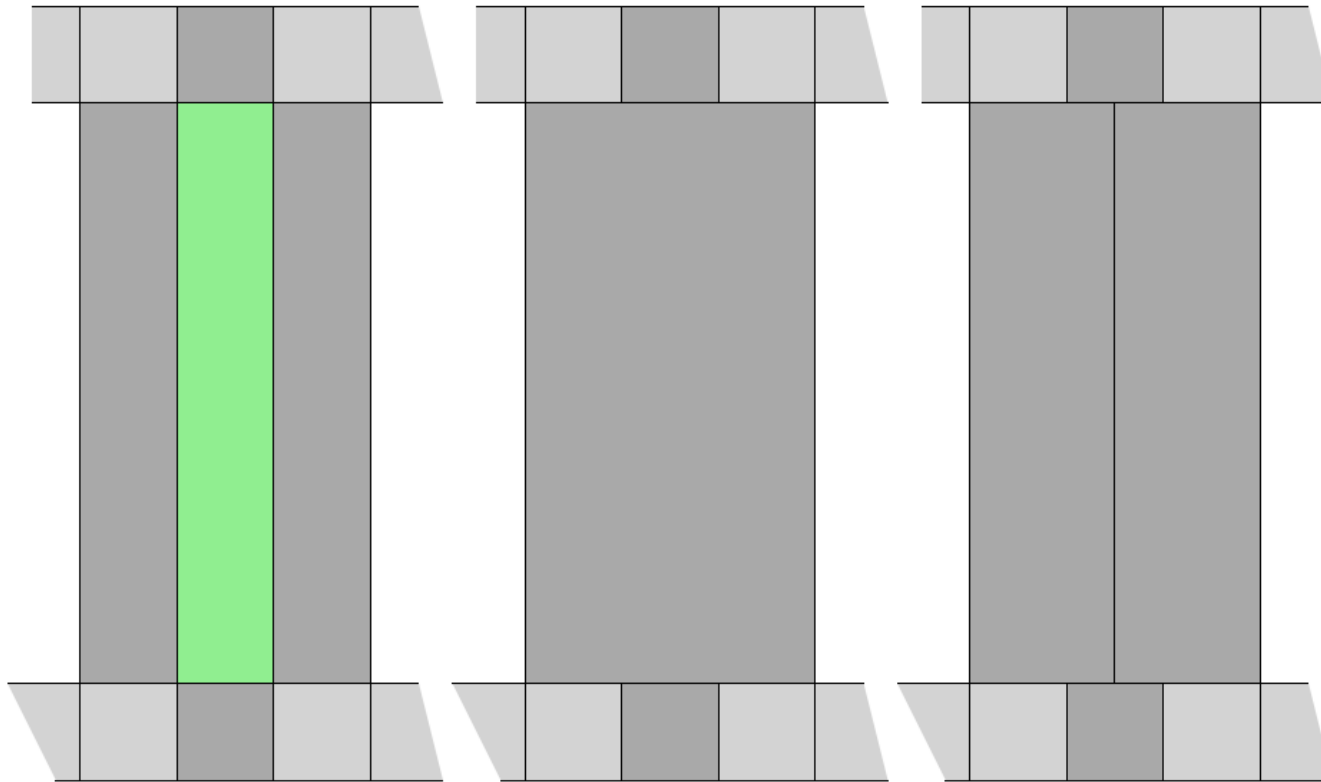
# Remove least important feature



# Remove least imp face (non-road)

Problem: all neighbors road faces...

2 options to continue: 2 road merges or... 1 split+merge parts



(green=non-road, dark grey=connection, light grey= junction)

# Some discussion

- Fine granularity indeed preferred by users? → usability test
- Just ideas presented, basic operations (split, merge) available, but is overall process steering Ok?
- Features not too often put back in queue with higher imp?
- Now only one non-read class, but in reality more classes (with different imp and characteristics) → more challenging
- Can road subclass granularity level be obtained by simple part-by-part tGAP approach?  
(first all minor roads collapsed to lines, before next level)
- Line road segment only removed when 2 neighbour non-road faces merge, because one was least imp  
(alternative: also select least imp line features (road) in tGAP?)

# Thanks

- for your attention!
- for support by the Dutch Technology Foundation STW (project number 11185)
- Suggestions, Questions?

