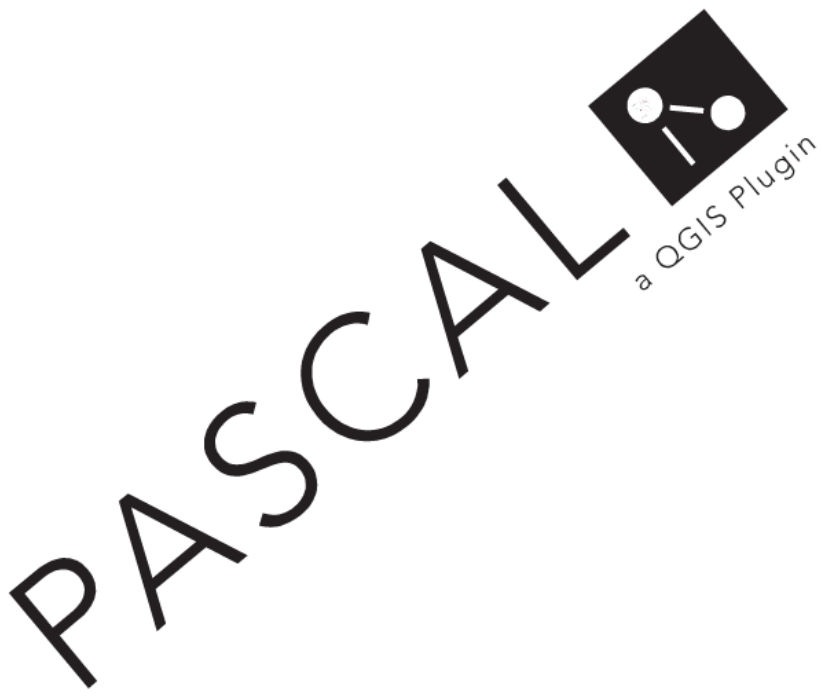
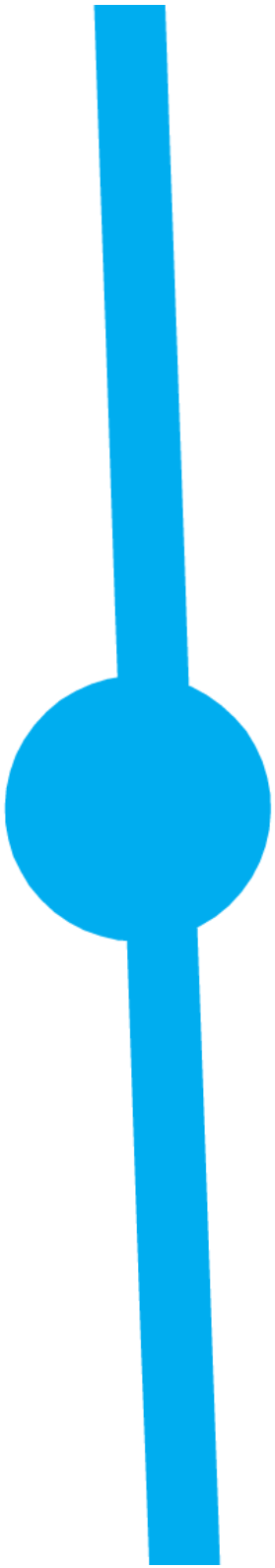

PASCAL
a QGIS Plugin

PASCAL

a QGIS Plugin



Problem?

In Amsterdam North, the *noord-zuidlijn* will be introduced in 2017

> Upcoming and growing area —> need for more and accessible transport nodes.

> Accessible public transport will result in higher use, eliminating car use

However, public transport can't reach everywhere —> so what locations are in need of new nodes, where to place them?

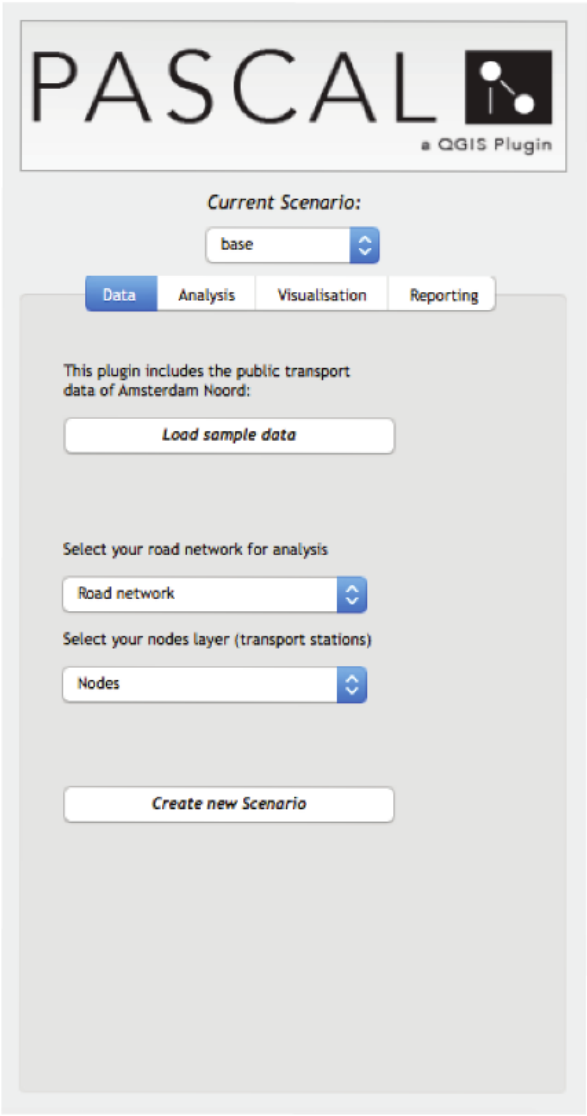
User requirements

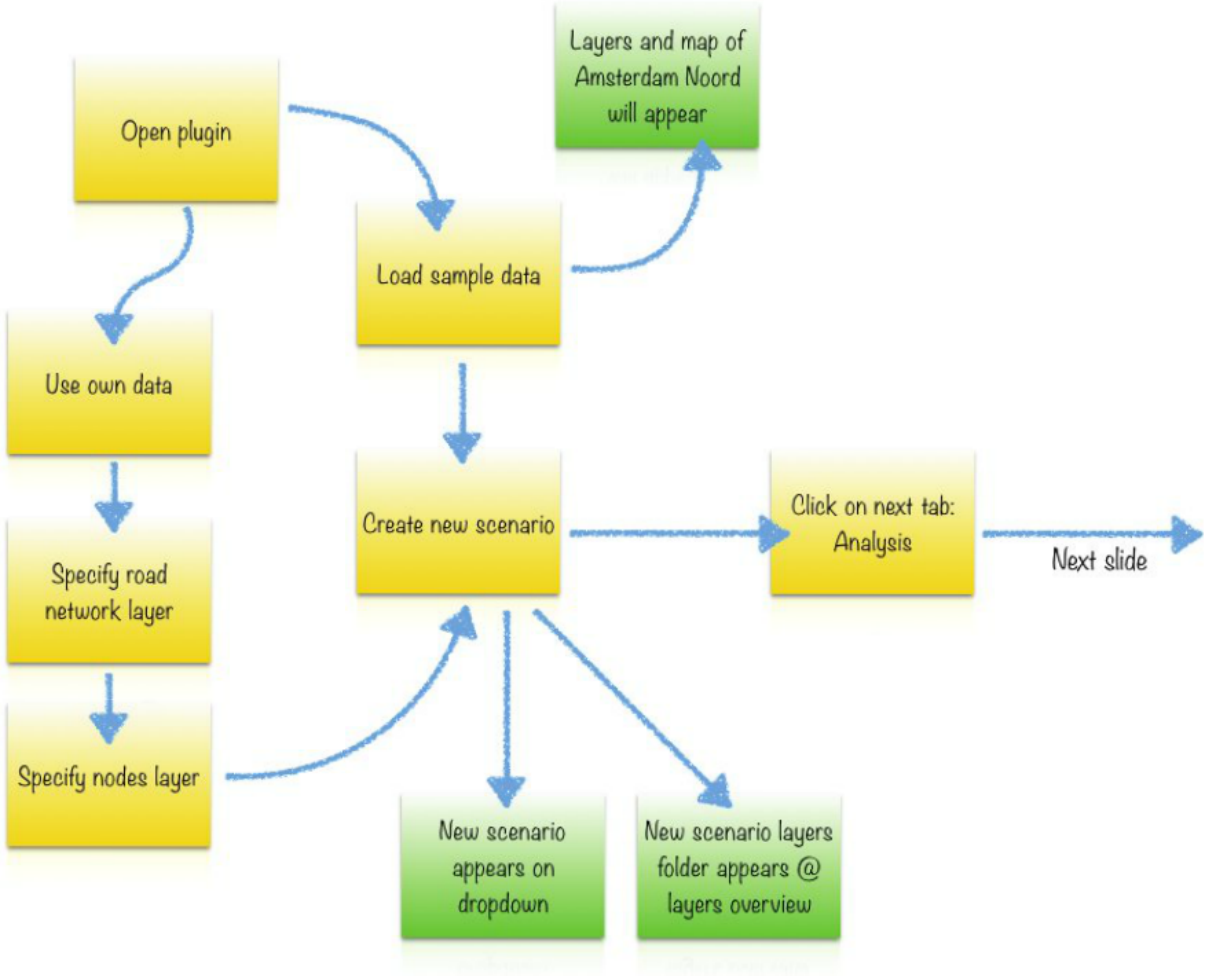
User requirements of the urban planner in the spatial-decision process placed in the three phases given by Simon (1960)

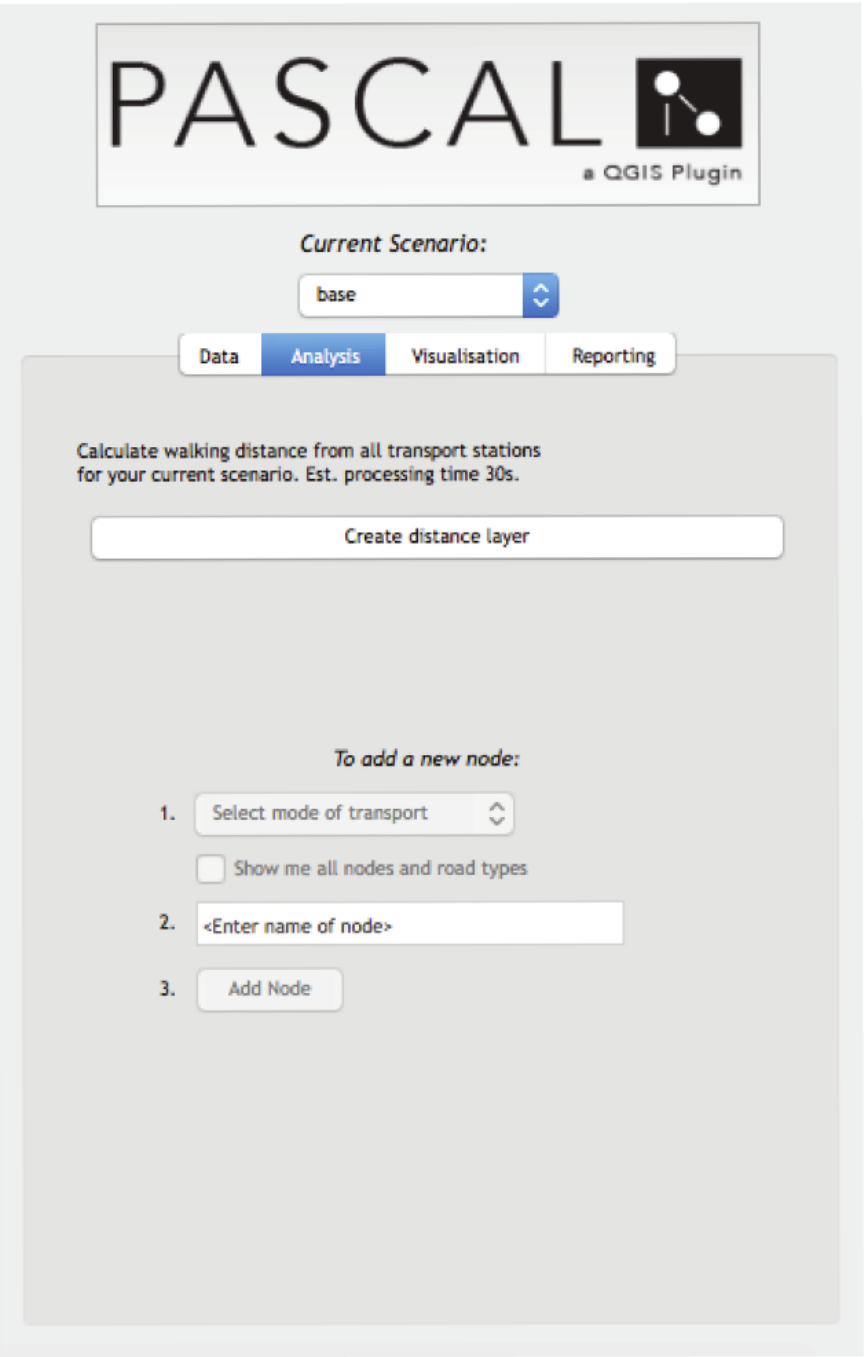
- **Intelligence**, to identify the issue, the urban planner needs to retrieve relevant information about the transport network and population
- **Design**, to work towards a solution, the urban planner needs new knowledge about the accessibility of the transit nodes, which can be produced by spatial analysis. This information can be used to work towards a solution
- **Choice**, to make a selection from different alternatives, the urban planner needs to compare and evaluate different scenarios

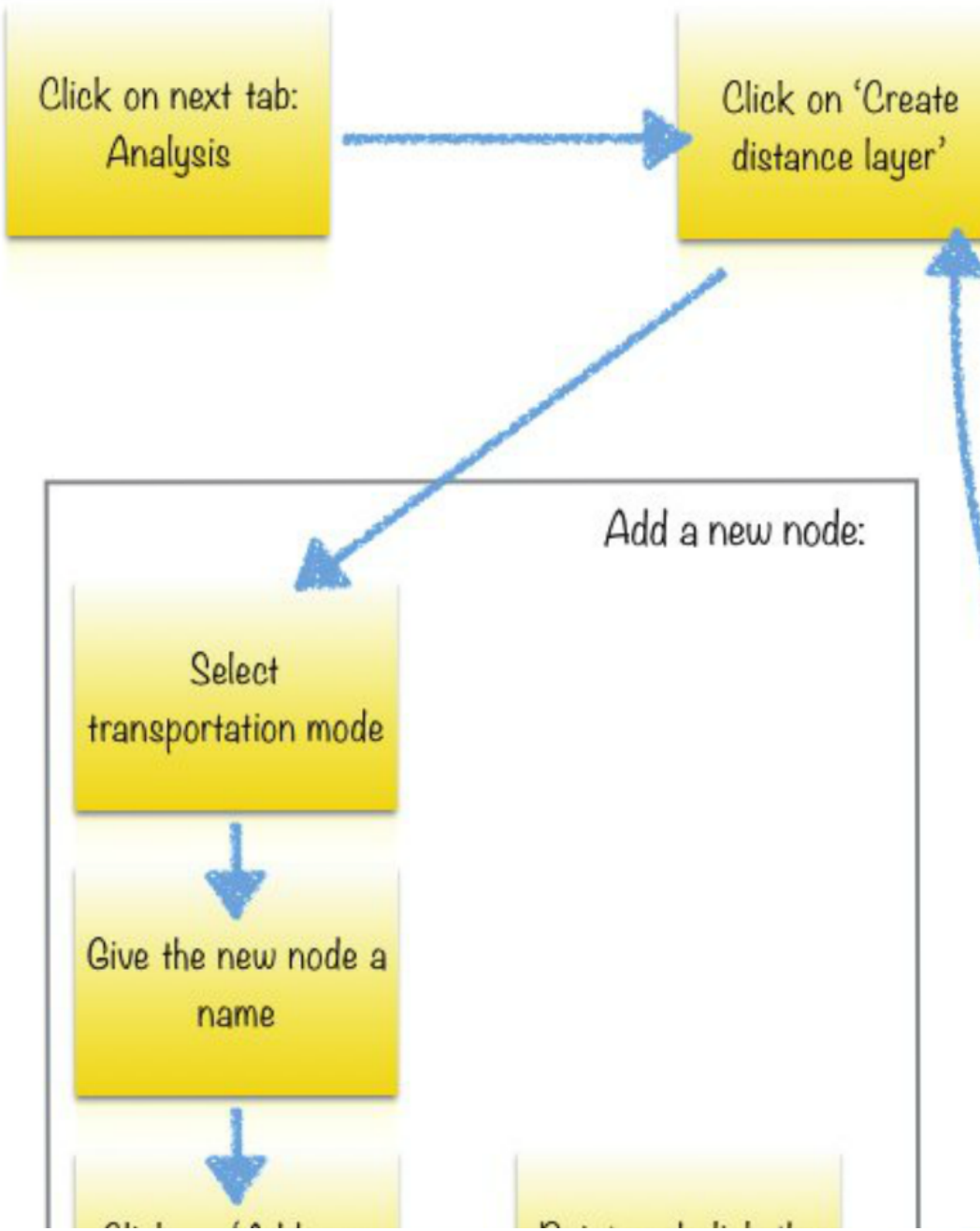
The urban planner can **iteratively** go through these phases. After generating new knowledge or ideas, he can go back to a previous phase.

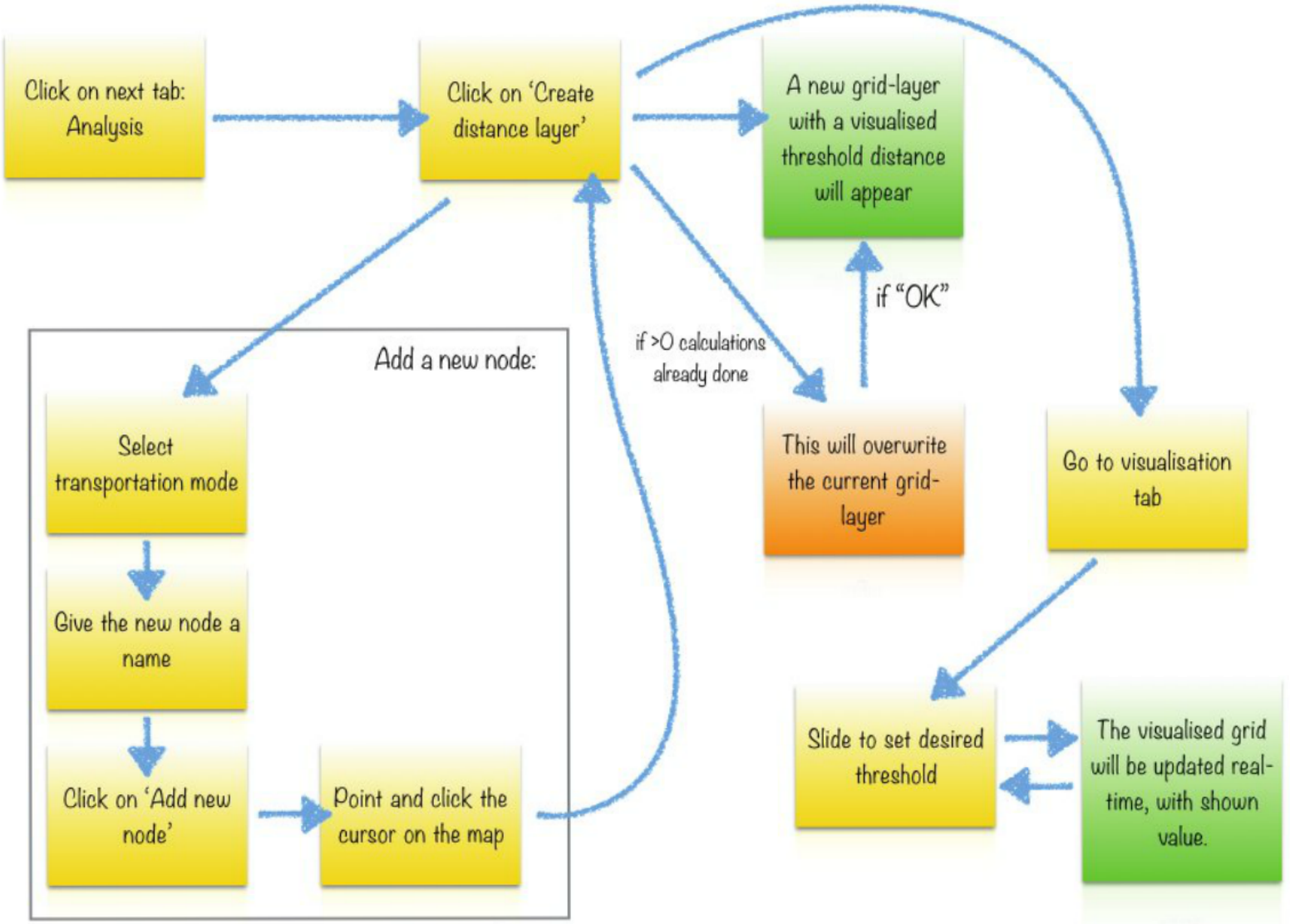
UX / Workflow

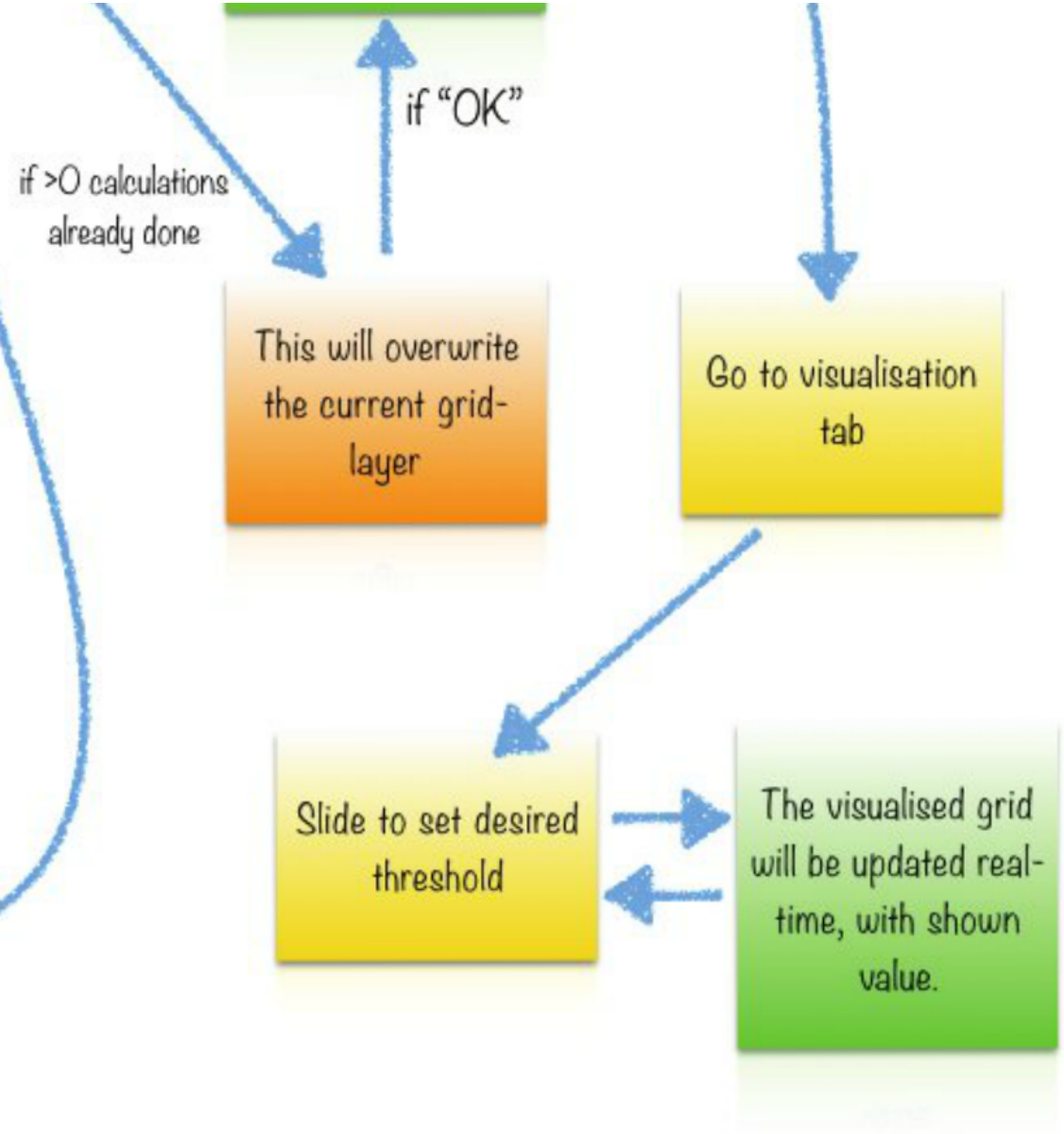

PASCAL
a QGIS Plugin
Current Scenario:
base
Data
Analysis
Visualisation
Reporting
This plugin includes the public transport data of Amsterdam Noord:
Load sample data
Select your road network for analysis
Road network
Select your nodes layer (transport stations)
Nodes
Create new Scenario

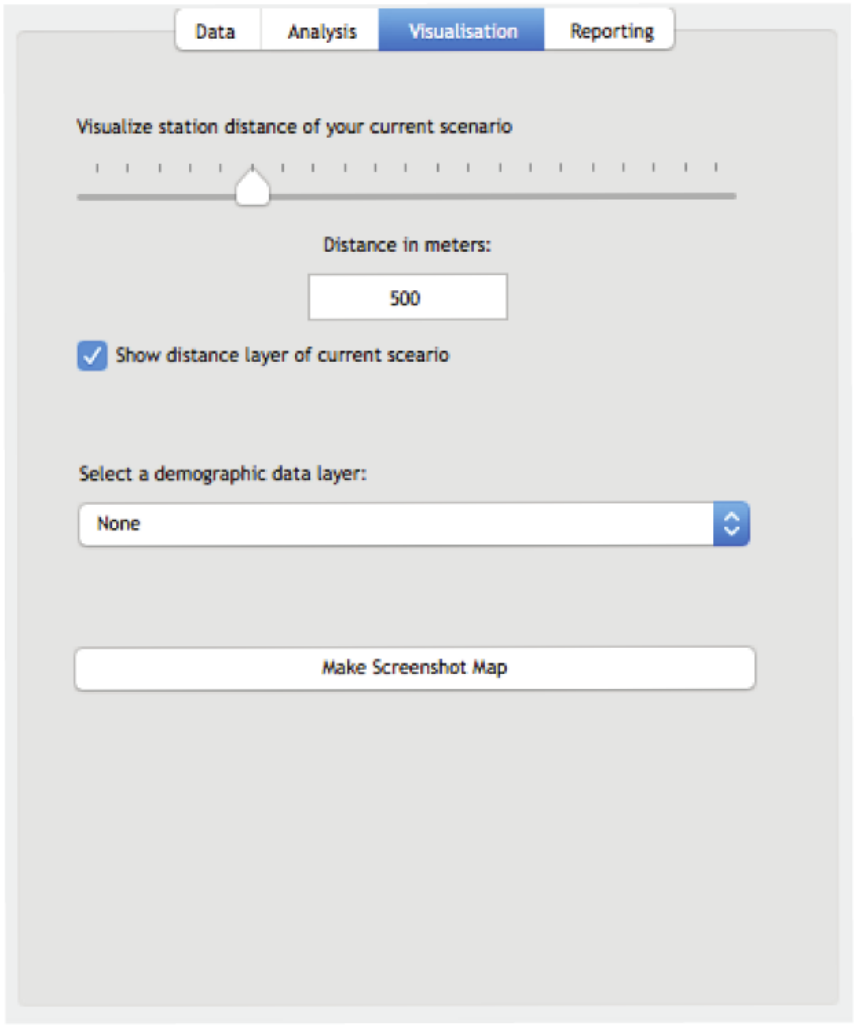

Open plugin
Layers and map of Amsterdam Noord will appear
Load sample data
Use own data
Create new scenario
Click on next tab: Analysis
Next slide
Specify road network layer
Specify nodes layer
New scenario appears on dropdown
New scenario layers folder appears @ layers overview

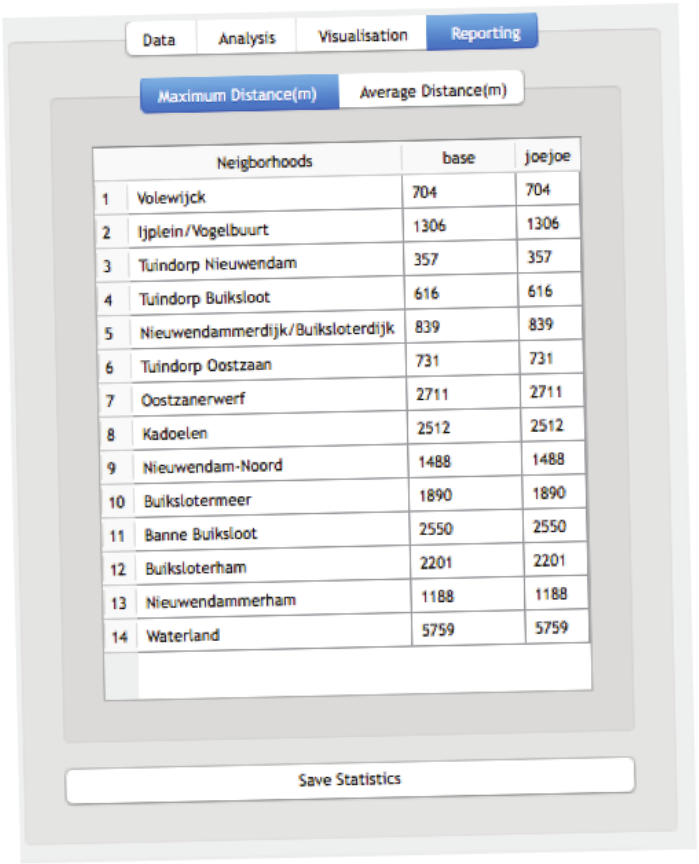

PASCAL
a QGIS Plugin
Current Scenario:
base
Data
Analysis
Visualisation
Reporting
Calculate walking distance from all transport stations
for your current scenario. Est. processing time 30s.
Create distance layer
To add a new node:
1.
Select mode of transport
Show me all nodes and road types
2.
<Enter name of node>
3.
Add Node


Click on next tab: Analysis
Click on 'Create distance layer'
Add a new node:
Select transportation mode
Give the new node a name

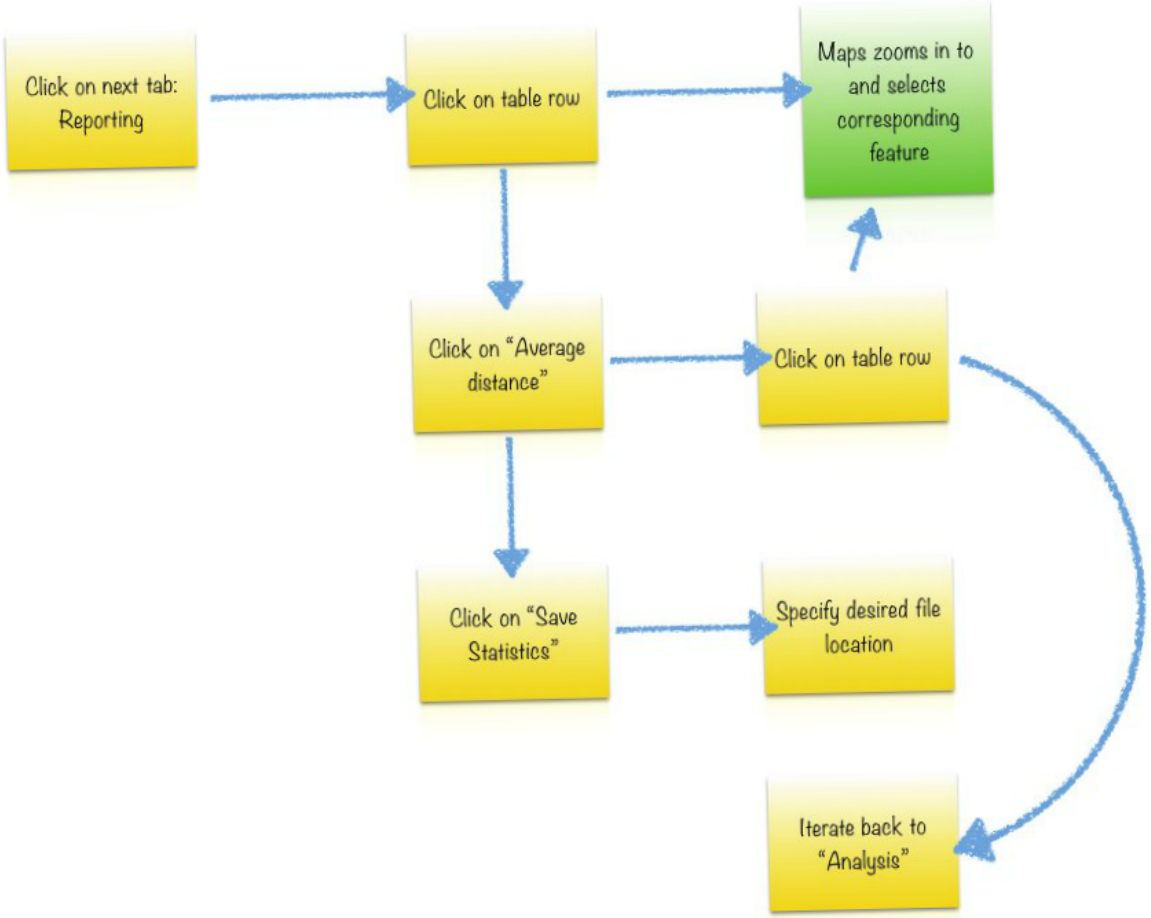

Click on next tab: Analysis
Click on 'Create distance layer'
A new grid-layer with a visualised threshold distance will appear
if "OK"
if >0 calculations already done
This will overwrite the current grid-layer
Go to visualisation tab
Add a new node:
Select transportation mode
Give the new node a name
Click on 'Add new node'
Point and click the cursor on the map
Slide to set desired threshold
The visualised grid will be updated real-time, with shown value.


if "OK"
if >0 calculations already done
This will overwrite the current grid-layer
Go to visualisation tab
Slide to set desired threshold
The visualised grid will be updated real-time, with shown value.


Data
Analysis
Visualisation
Reporting
Visualize station distance of your current scenario
Distance in meters:
500
Show distance layer of current sceario
Select a demographic data layer:
None
Make Screenshot Map



	Neigborhoods	base	joejoe
1	Volewijck	704	704
2	Ijplein/Vogelbuurt	1306	1306
3	Tuindorp Nieuwendam	357	357
4	Tuindorp Buiksloot	616	616
5	Nieuwendammerdijk/Buiksloterdijk	839	839
6	Tuindorp Oostzaan	731	731
7	Oostzanerwerf	2711	2711
8	Kadoelen	2512	2512
9	Nieuwendam-Noord	1488	1488
10	Buikslotermeer	1890	1890
11	Banne Buiksloot	2550	2550
12	Buiksloterham	2201	2201
13	Nieuwendammerham	1188	1188
14	Waterland	5759	5759



Data

PASCAL stores topological and thematic data layers to support cartographic display and spatial analysis

Road network

- Linestring vector data
- Attributes: Name, road type
- Spatial analysis —> Graph

Transport nodes

- Point vector data
- Attributes: Name,Transport type
- Spatial analysis —> Graph, tied points
- Allow user to manipulate —> add new features

Neighborhoods

- Polygon vector data

Transport nodes

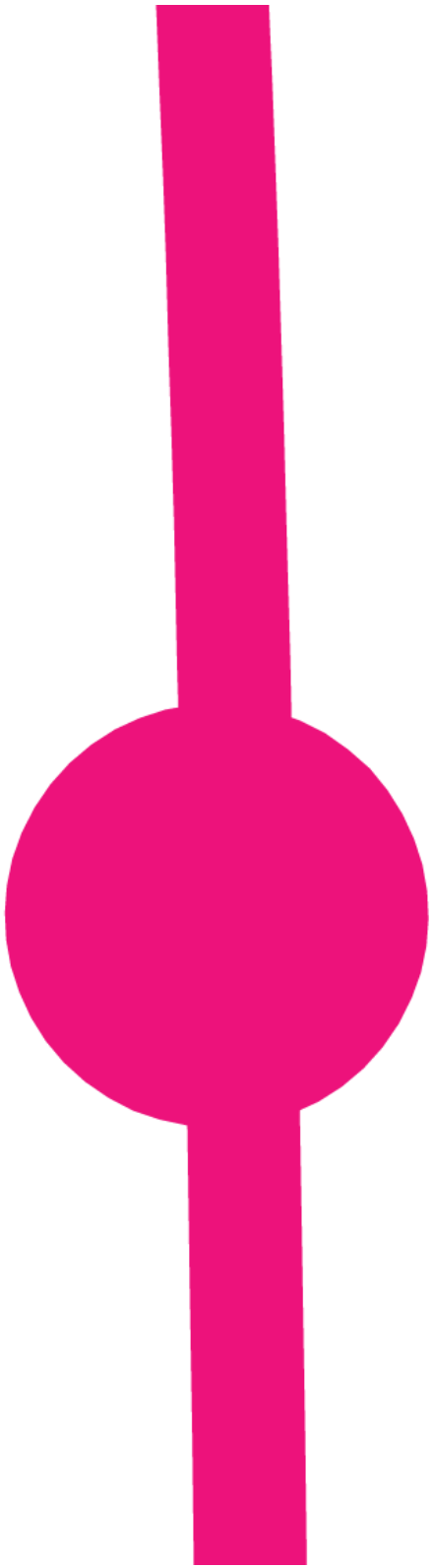
- Point vector data
- Attributes: Name,Transport type
- Spatial analysis —> Graph, tied points
- Allow user to manipulate —> add new features

Neighborhoods

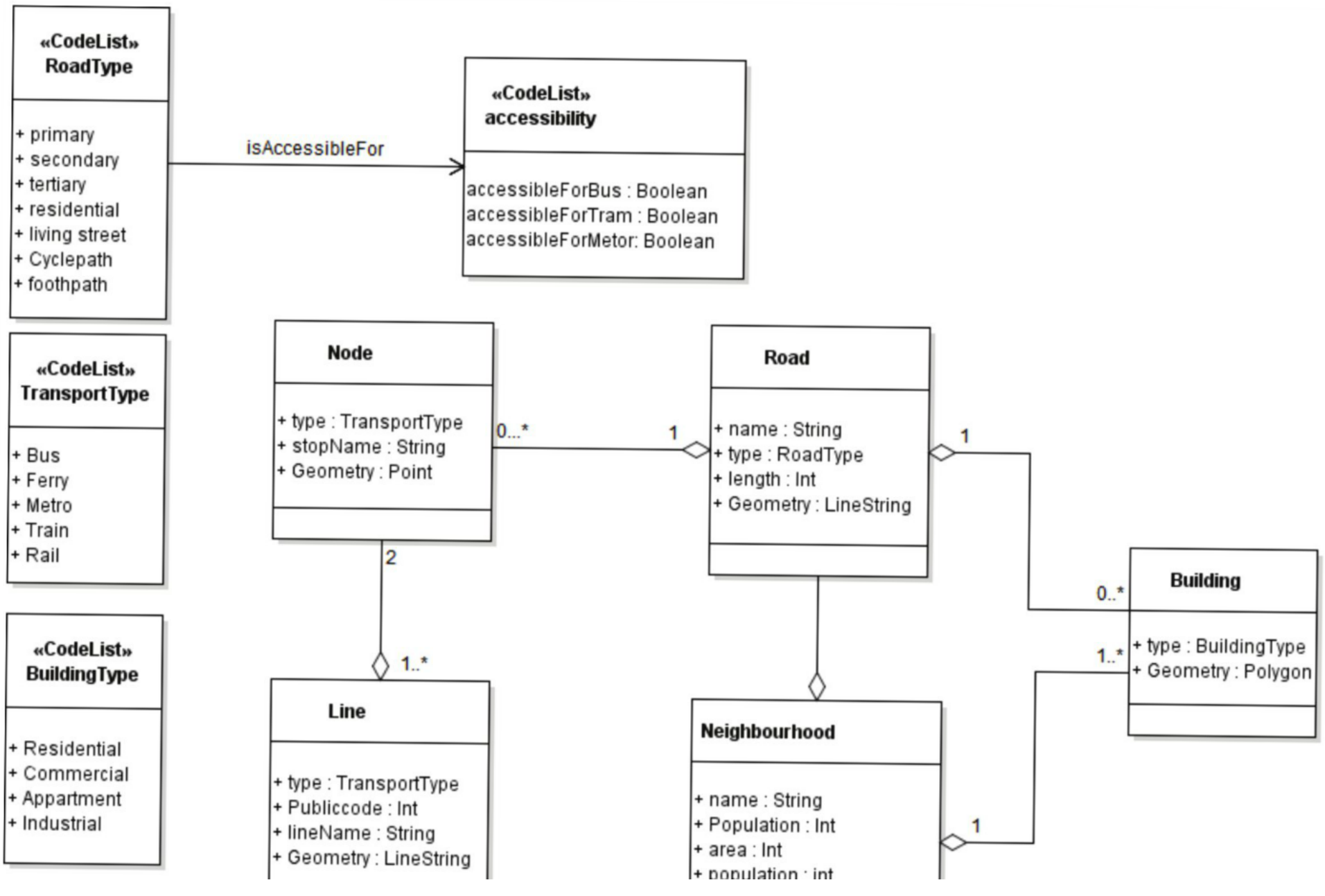
- Polygon vector data
- Attributes: Name, demographic data
- Spatial analysis —> Raster statistics per polygon
- Pre-processing —> thematic data layers: Population, Population density, Cars per Household, Elderly

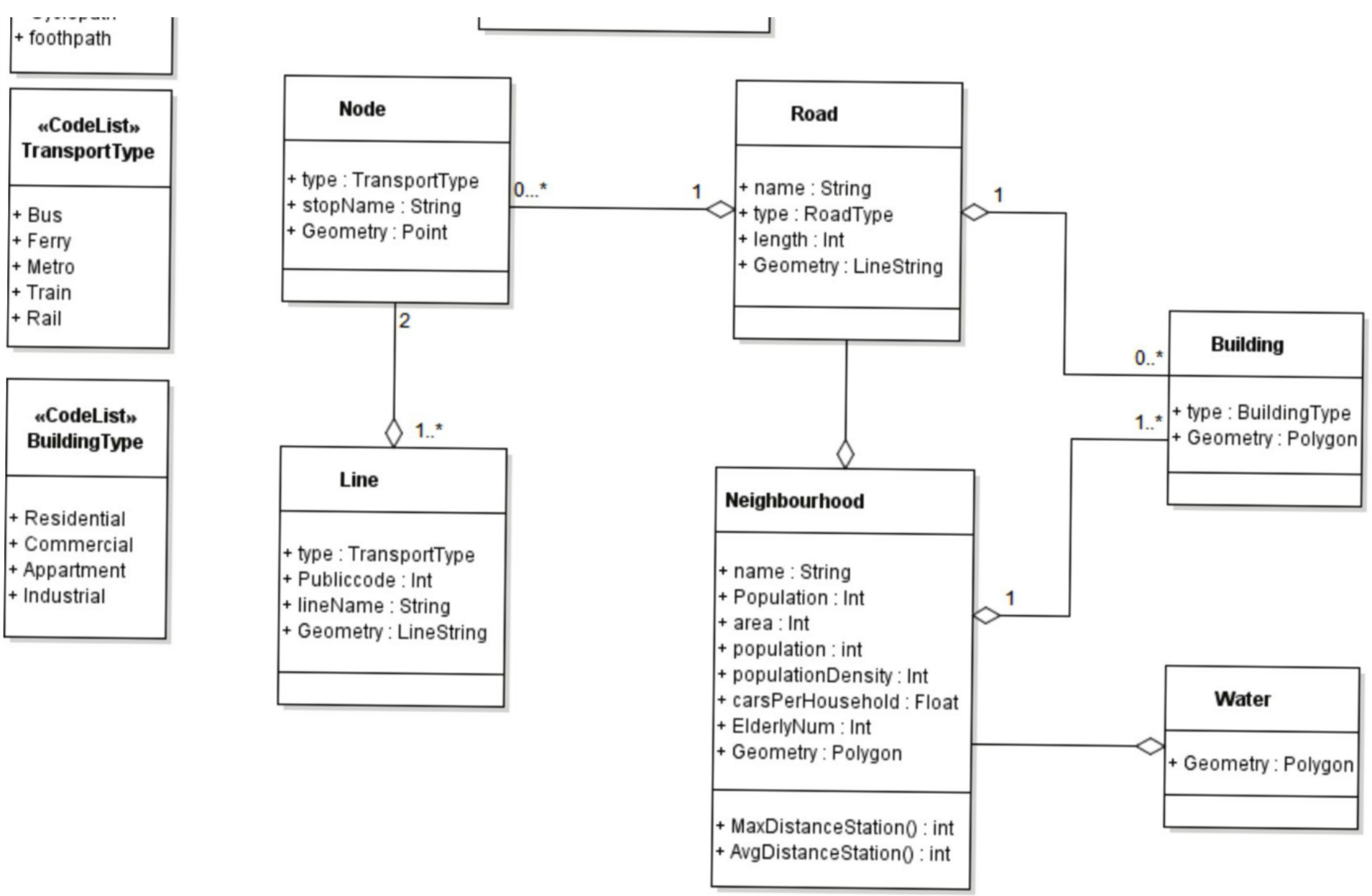
Carthographic layers

- Buildings, water, public transport lines



UML


«CodeList»
RoadType
+ primary
+ secondary
+ tertiary
+ residential
+ living street
+ Cyclepath
+ foothpath
isAccessibleFor
«CodeList»
accessibility
accessibleForBus : Boolean
accessibleForTram : Boolean
accessibleForMetor: Boolean
«CodeList»
TransportType
+ Bus
+ Ferry
+ Metro
+ Train
+ Rail
Node
+ type : TransportType
+ stopName : String
+ Geometry : Point
0...*
1
Road
+ name : String
+ type : RoadType
+ length : Int
+ Geometry : LineString
1
2
0..*
Building
+ type : BuildingType
+ Geometry : Polygon
1..*
«CodeList»
BuildingType
+ Residential
+ Commercial
+ Appartment
+ Industrial
1..*
Line
+ type : TransportType
+ Publiccode : Int
+ lineName : String
+ Geometry : LineString
Neighbourhood
+ name : String
+ Population : Int
+ area : Int
+ population : int
1


+ foothpath
«CodeList»
TransportType
+ Bus
+ Ferry
+ Metro
+ Train
+ Rail
«CodeList»
BuildingType
+ Residential
+ Commercial
+ Appartment
+ Industrial
Node
+ type : TransportType
+ stopName : String
+ Geometry : Point
0...*
1
Road
+ name : String
+ type : RoadType
+ length : Int
+ Geometry : LineString
1
2
1..*
Line
+ type : TransportType
+ Publiccode : Int
+ lineName : String
+ Geometry : LineString
0..*
Building
+ type : BuildingType
+ Geometry : Polygon
1..*
Neighbourhood
+ name : String
+ Population : Int
+ area : Int
+ population : int
+ populationDensity : Int
+ carsPerHousehold : Float
+ ElderlyNum : Int
+ Geometry : Polygon
+ MaxDistanceStation() : int
+ AvgDistanceStation() : int
1
Water
+ Geometry : Polygon

Analysis

Distance to station grid

> Building graph of road network and tie nodes to it.

> Calculate service area for all stations and selecting the minimum value for each vertex

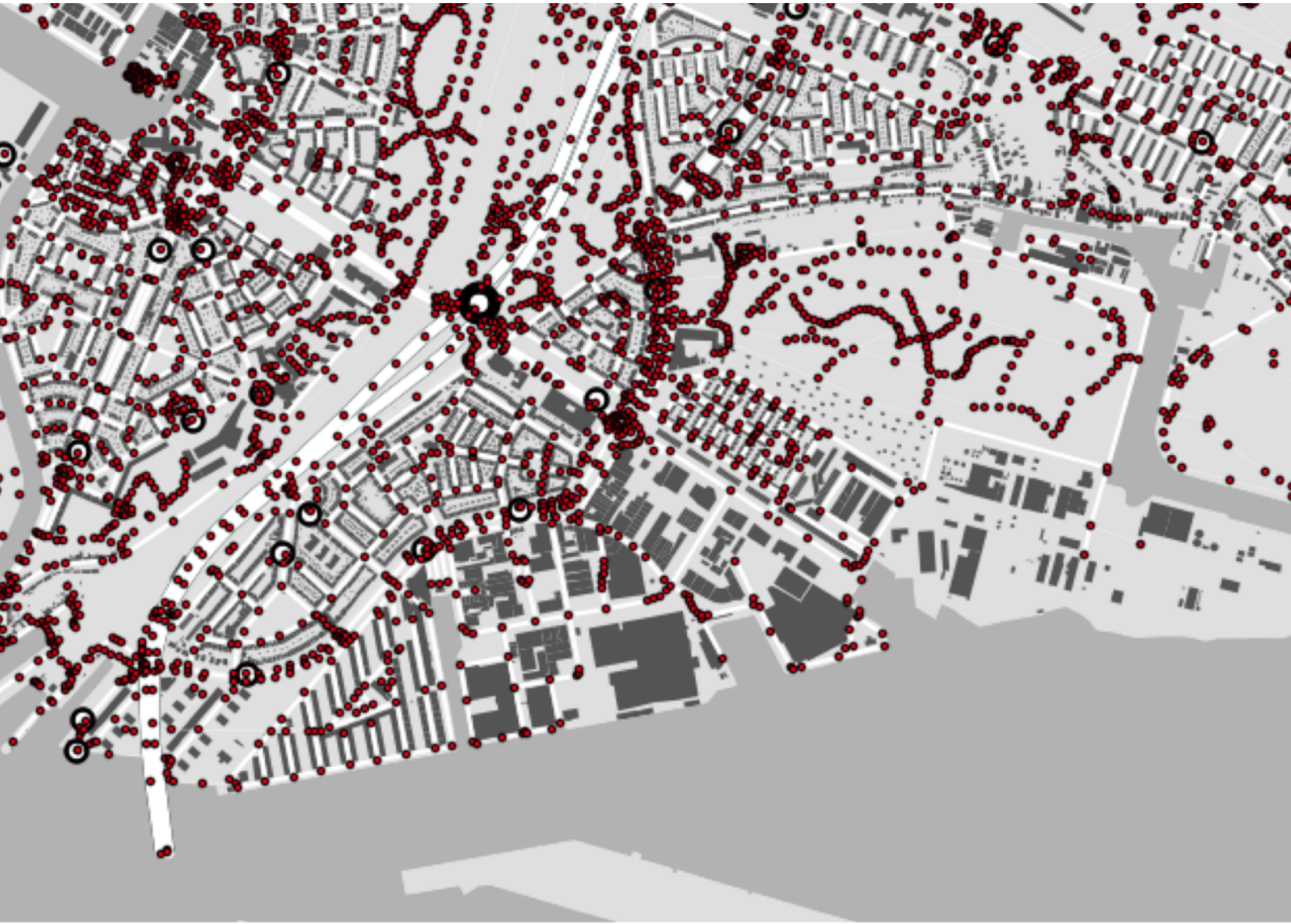
Service area: *'Distance from a station to each vertex in the road network'*.

> Interpolate (**IDW**) between vertices to get distance to station grid.

Grid Statistics

> Intersect all neighborhood polygons with the *'distance to station'* grid.

> Retrieve **max** and **average** distance for each neighborhood



Visualisation

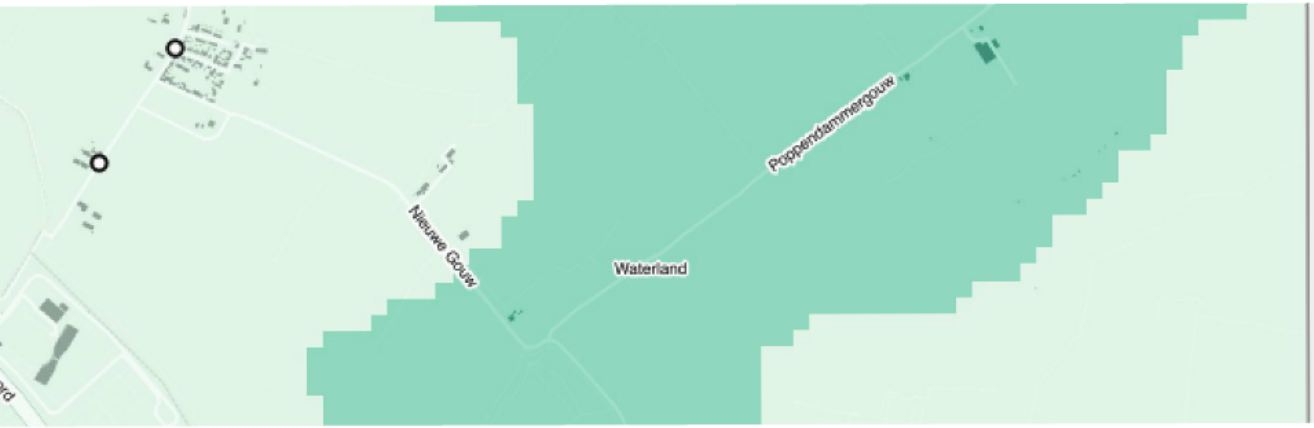
Data | Analysis | Visualisation | Reporting

Visualize station distance of your current scenario

Distance in meters:

500

Show distance layer of current sceario



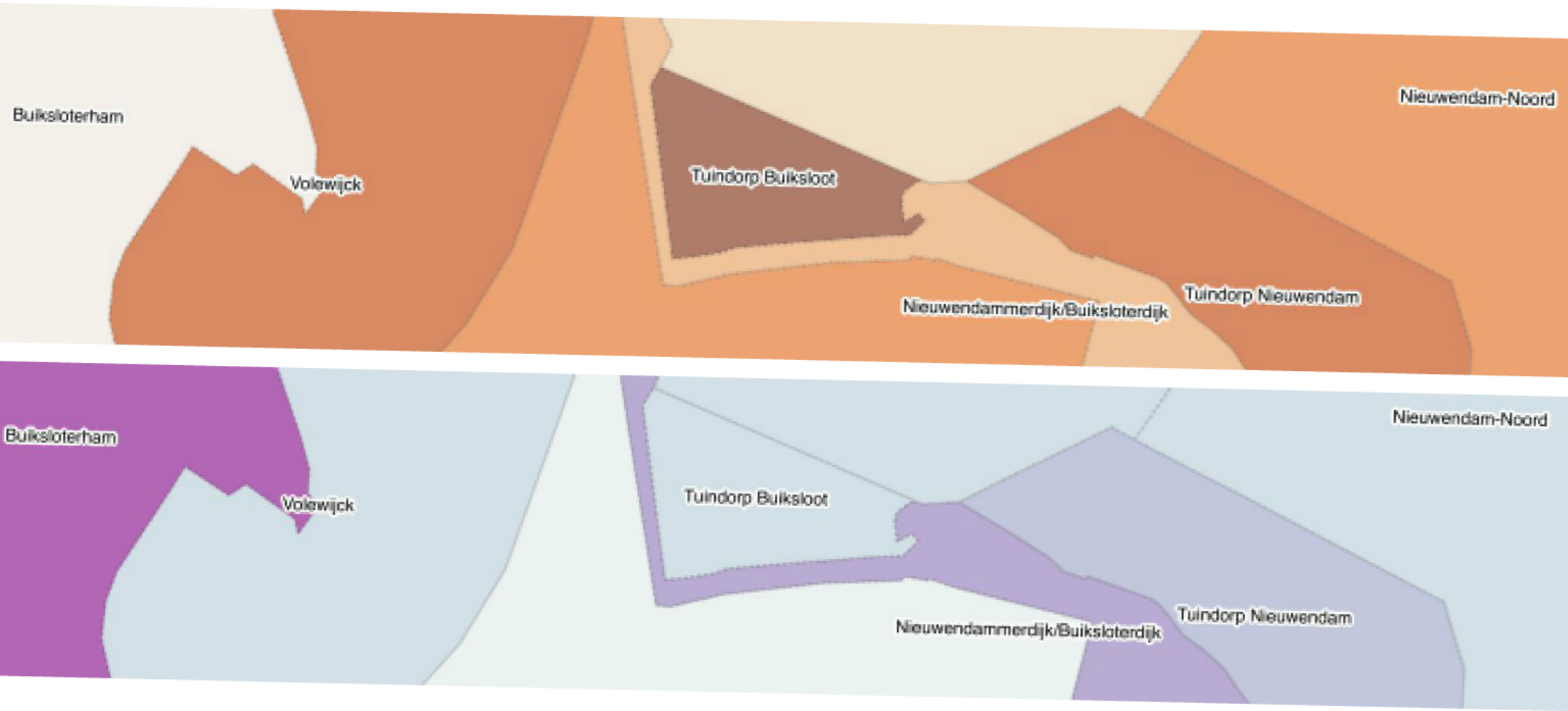
Data | Analysis | Visualisation | Reporting

Visualize station distance of your current scenario

Distance in meters:

1124

Show distance layer of current sceario

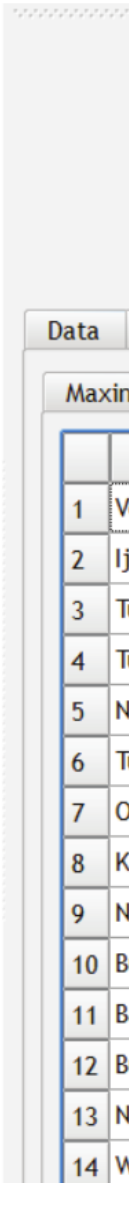

Buiksloterham
Volewijck
Tuindorp Buiksloot
Nieuwendam-Noord
Nieuwendammerdijk/Buiksloterdijk
Tuindorp Nieuwendam
Buiksloterham
Volewijck
Tuindorp Buiksloot
Nieuwendam-Noord
Nieuwendammerdijk/Buiksloterdijk
Tuindorp Nieuwendam


Uitkade
Van der Pekstraat
Meidoornweg
Buiksloterweg
Nieuwe Leeuwarderweg
Adelaarsweg
Meeuwenlaan
Zamenhofstraat
Johan van Hasseltweg
Zwarte Pad
Nieuwendammerdijk
Uitpromenade
Laanweg
Buiksloterweg
Meeuwenlaan
Sixhavenweg
Uplein
Uplein
Noordwal

Reporting

Average and maximum station distance of the various neighborhoods can be compared for the different scenarios that are created.

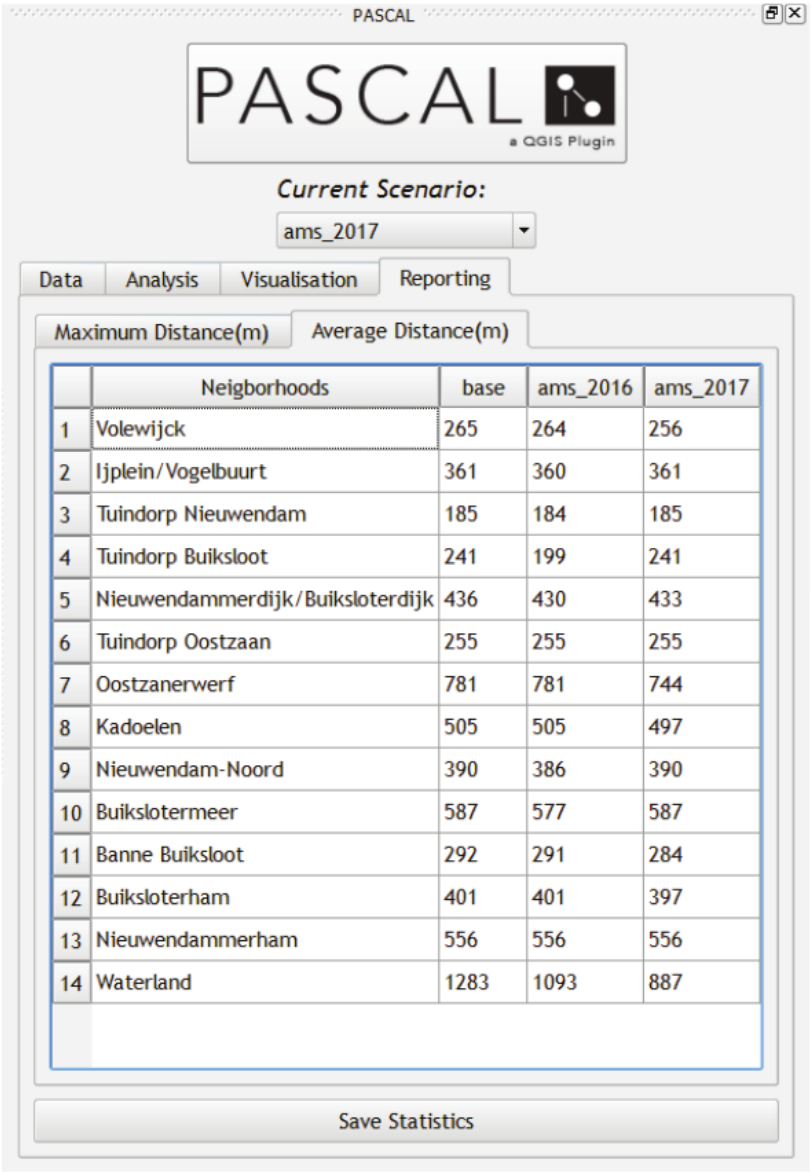
> The neighborhood statistics can be saved to a .csv file



d maximum station distance of the various neighborhoods

pared for the different scenarios that are created.

borhood statistics can be saved to a .csv file



	Neigborhoods	base	ams_2016	ams_2017
1	Volewijck	265	264	256
2	Ijplein/Vogelbuurt	361	360	361
3	Tuindorp Nieuwendam	185	184	185
4	Tuindorp Buiksloot	241	199	241
5	Nieuwendammerdijk/Buiksloterdijk	436	430	433
6	Tuindorp Oostzaan	255	255	255
7	Oostzanerwerf	781	781	744
8	Kadoelen	505	505	497
9	Nieuwendam-Noord	390	386	390
10	Buikslotermeer	587	577	587
11	Banne Buiksloot	292	291	284
12	Buiksloterham	401	401	397
13	Nieuwendammerham	556	556	556
14	Waterland	1283	1093	887

To conclude

What was the problem: Does **PASCAL** support the urban planner in the decision-making process of placing new nodes?

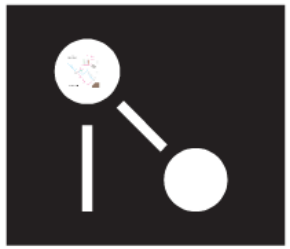
- **PASCAL** is able to support the urban planner —> localizing potential areas for new transit nodes, virtually placing nodes and comparing different scenarios
- All components of a **SDSS** are addressed —> iterative analysis of distance, interactive visualization and reporting, …

Discussion

- More overlap with other stakeholders —> implementation of cost analysis
- Frequency, capacity of public transport
- Connecting destinations (points of interest)

QUESTIONS?

PASCAL



a QGIS Plugin