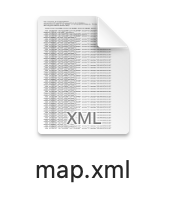
**Multi-Transportation models from OD matrix for Oakville Example**

**1. Get network file – two ways**

Method 1- download from OSM

Step 1: network (roads)download from osm → map.xml



Step 2: map.xml → netconvert → .net.xml

netconvert --osm 1\_map.xml -o file.net.xml.gz

A close-up of a document

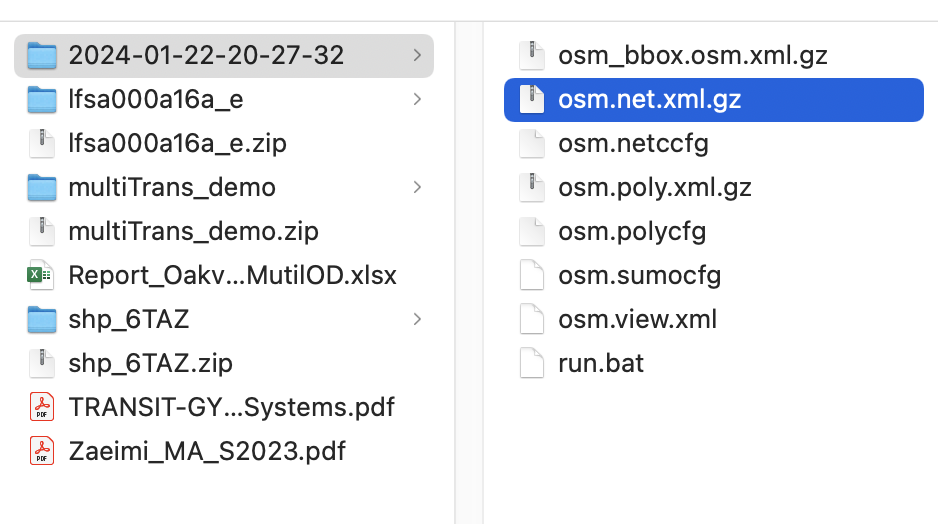
Description automatically generated

Method 2- download from OSM web wizard for SUMO

.net.xml.gz

A screenshot of a map

Description automatically generated



**2 TAZ**

2.1 Get shapefile

Step 1: Download FSA shapefile from government website.

A map of canada with black lines

Description automatically generated

A map of a city

Description automatically generated

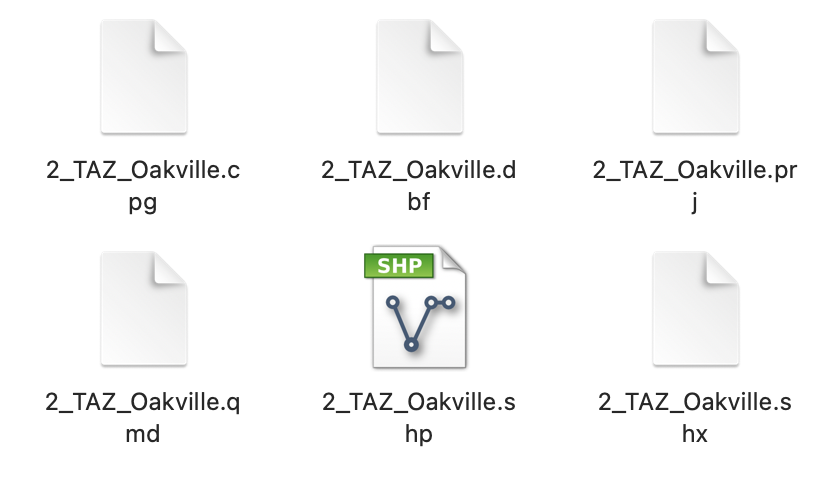
Step 2: edit the zone in QGIS

**Vector ‣ Geoprocessing Tools-Difference**

5 + 1.shp → 2\_TAZ\_Oakville

A yellow map with blue text

Description automatically generated



2.2 transfer the shapefile to sumo poly.xml file

shapefile.shp → polyconvert → taz.poly.xml

polyconvert --shapefile-prefix TAZ\_Oakville --shapefile.guess-projection true --shapefile.traditional-axis-mapping true --shapefile.id-column CFSAUID -n 1\_net.net.xml.gz -o 2\_taz.poly.xml

A close-up of a document

Description automatically generated

2.3 SUMO generates TAZ file from poly.xml file

taz.poly.xml → edgesInDistricts.py →TAZ.xml

A screenshot of a computer

Description automatically generated

MAC

python E:\\Sheridan\\Sumo\\sumo/tools/edgesInDistricts.py -n 1\_net.net.xml.gz -t 2\_taz.poly.xml -o 2\_TAZ.xml

WINDOWS

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools/edgesInDistricts.py" -n 1\_net.net.xml.gz -t 2\_taz.poly.xml -o 2\_TAZ.xml

**3 Trip (with o & d)**

**step 1: prepare OD matrix files** **A white paper with a corner curled up

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**step 2: generate trip form taz.xml + od-matrix**

od2trips -v --taz-files 2\_TAZ.xml --vtype vType --prefix vType --od-matrix-files OD\_matrix\_vType.od -o vType.odtrips.xml

od2trips -v --taz-files 2\_TAZ.xml --vtype drive --prefix drive --od-matrix-files 3\_OD\_matrix\_drive.od -o 3\_drive.odtrips.xml

od2trips -v --taz-files 2\_TAZ.xml --vtype bus --prefix bus --od-matrix-files 3\_OD\_matrix\_bus.od -o 3\_bus.odtrips.xml

od2trips -v --taz-files 2\_TAZ.xml --vtype pudo --prefix pudo --od-matrix-files 3\_OD\_matrix\_pudo.od -o 3\_pudo.odtrips.xml

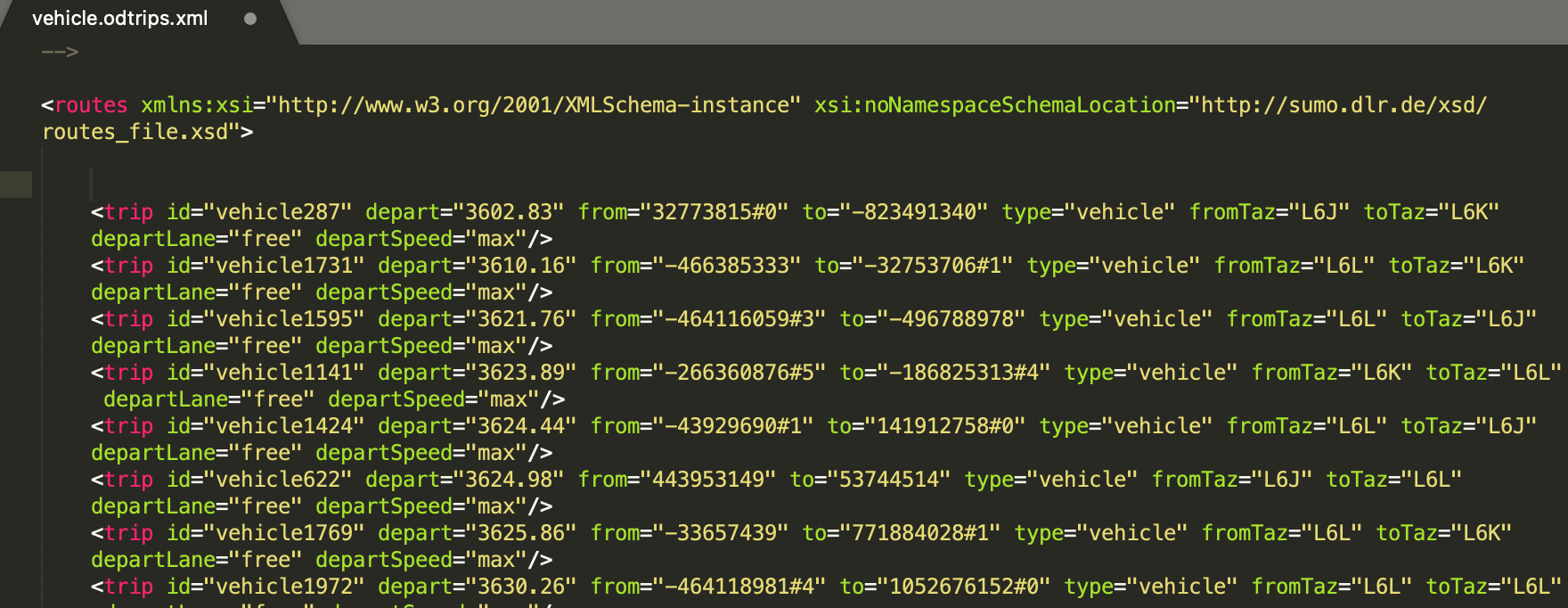
od2trips -v --taz-files 2\_TAZ.xml --vtype pedestrian --prefix pedestrian --od-matrix-files 3\_OD\_matrix\_pedestrian.od -o 3\_pedestrian.odtrips.xml

od2trips -v --taz-files 2\_TAZ.xml --vtype bike --prefix bike --od-matrix-files 3\_OD\_matrix\_bike.od -o 3\_bike.odtrips.xml

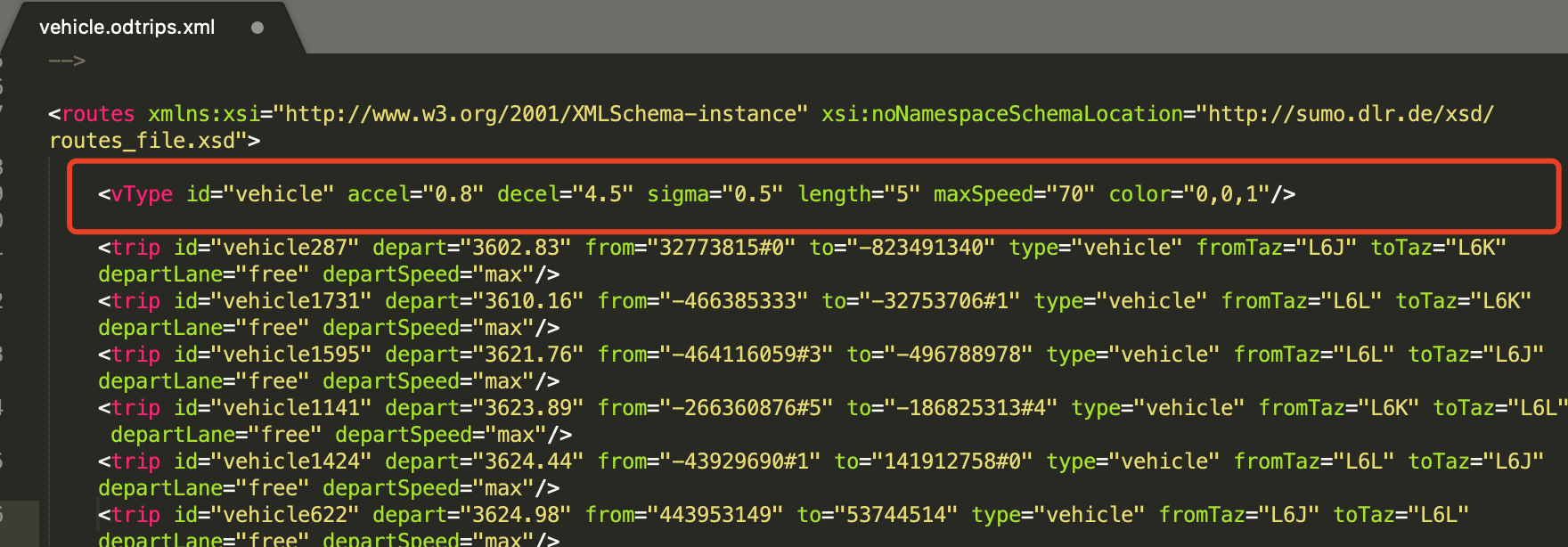
#od2trips -v --taz-files 2\_TAZ.xml --vtype carpool --prefix carpool --od-matrix-files 3\_OD\_matrix\_carpool.od -o 3\_carpool.odtrips.xml

A close-up of a document

Description automatically generated



**step 3: must define vtype in vType.odtrips.xml**

****

<https://sumo.dlr.de/docs/Vehicle_Type_Parameter_Defaults.html>

<vType id="drive" accel="0.8" decel="4.5" sigma="0.5" length="5" maxSpeed="70" color="0,0,1"/>

<vType id="bus" vClass="bus" color="1,1,0.5"/>

<vType id="pudo" vClass="bicycle" color="1,1,0"/>

<vType id="pedestrian" vClass="pedestrian" color="1,0,1"/>

<vType id="bike" vClass="bicycle" color="1,1,0"/>

<vType id="passenger" vClass="passenger" color="0,1,1"/>

**4 generate routes (with every edge) for specific requirement**

osm.sumocfg → duarouter →od\_file.odtrips.rou.xml

tripvaludation

to find the shortest path to assign to each vehicle in the simulation.

duarouter -c duarcfg\_file.trips2routes.duarcfg -o multiTrans.rou.xml

duarouter -v -n sumo\_net.net.xml --route-files vehicle.odtrips.xml,bicycle.odtrips.xml,bus.odtrips.xml,pedestrian.odtrips.xml --xml-validation never --no-step-log true -o multiTrans.rou.xml

duarouter -v -n sumo\_net.net.xml --route-files vehicle.odtrips.xml --xml-validation never --no-step-log true -o vehicle.rou.xml

duarouter -v -n sumo\_net.net.xml --route-files bicycle.odtrips.xml --xml-validation never --no-step-log true -o bicycle.rou.xml

duarouter -v -n sumo\_net.net.xml --route-files bus.odtrips.xml --xml-validation never --no-step-log true -o bus.rou.xml

duarouter -v -n sumo\_net.net.xml --route-files pedestrian.odtrips.xml --xml-validation never --no-step-log true -o pedestrian.rou.xml

**5 run simulation in SUMO**

<?xml version="1.0" encoding="UTF-8"?>

<configuration>

<input>

<net-file value="1\_net.net.xml.gz " />

<route-files value="vehicle.odtrips.xml, bicycle.odtrips.xml, bus.odtrips.xml, pedestrian.odtrips.xml"/>

<!--route-files value="vehicle.rou.xml, bicycle.rou.xml, bus.rou.xml, pedestrian.rou.xml"/-->

<additional-files value="taz3.poly.xml"/>

</input>

<output>

<output-file value="od\_route\_file.odtrips.rou.xml" />

</output>

<report>

<xml-validation value="never" />

<no-step-log value="true" />

</report>

</configuration>

file: osm.sumocfg

<input>

<net-file value="osm.net.xml.gz"/>

<route-files value="osm.bicycle.trips.xml,osm.bus.trips.xml,osm.motorcycle.trips.xml,osm.pedestrian.trips.xml"/>

<additional-files value="osm.poly.xml.gz"/>

</input>

oakville

* -rand.grid

**6 generate od matrix from route/trips and taz file**

generate od-matrix from rou.xml + taz.xml by using route2OD.py

python PATH\\route2OD.py -r vType.odtrips.xml -a TAZ.xml -o vType\_ODGenerate.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r 3\_vehicle.odtrips.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_vehicle.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r 3\_bicycle.odtrips.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_bicycle.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r 3\_pedestrianodtrips.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_pedestrian.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r 3\_bus.odtrips.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_bus.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r vehicle.rou.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_vehicleFromRou.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r bicycle.rou.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_bicycleFromRou.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r pedestrian.rou.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_pedestrianFromRou.xml

python "C:\\Program Files (x86)\\Eclipse\\Sumo\\tools\\route\\route2OD.py" -r bus.rou.xml -a 2\_TAZ.xml -o 4\_ODGenerate\_busFromRou.xml

PATH: MAC → E:\\Sheridan\\Sumo\\sumo/tools/route/ WINDOWS -> C:\\Program Files (x86)[\\Eclipse\\Sumo\\tools\\route](file://Eclipse//Sumo//tools//route)

A close-up of a document

Description automatically generated

A screenshot of a computer program

Description automatically generated