1. **OpenStreetMap地图导入并生成仿真文件：**

初始仿真地图与交通流等文件由SUMO中的osmWebWizard.py生成。说明网址：<https://sumo.dlr.de/docs/Tools/Import/OSM.html>

osmWebWizard.py可以在生成地图等文件时设置初始状态下的车辆类型，交通流数据：

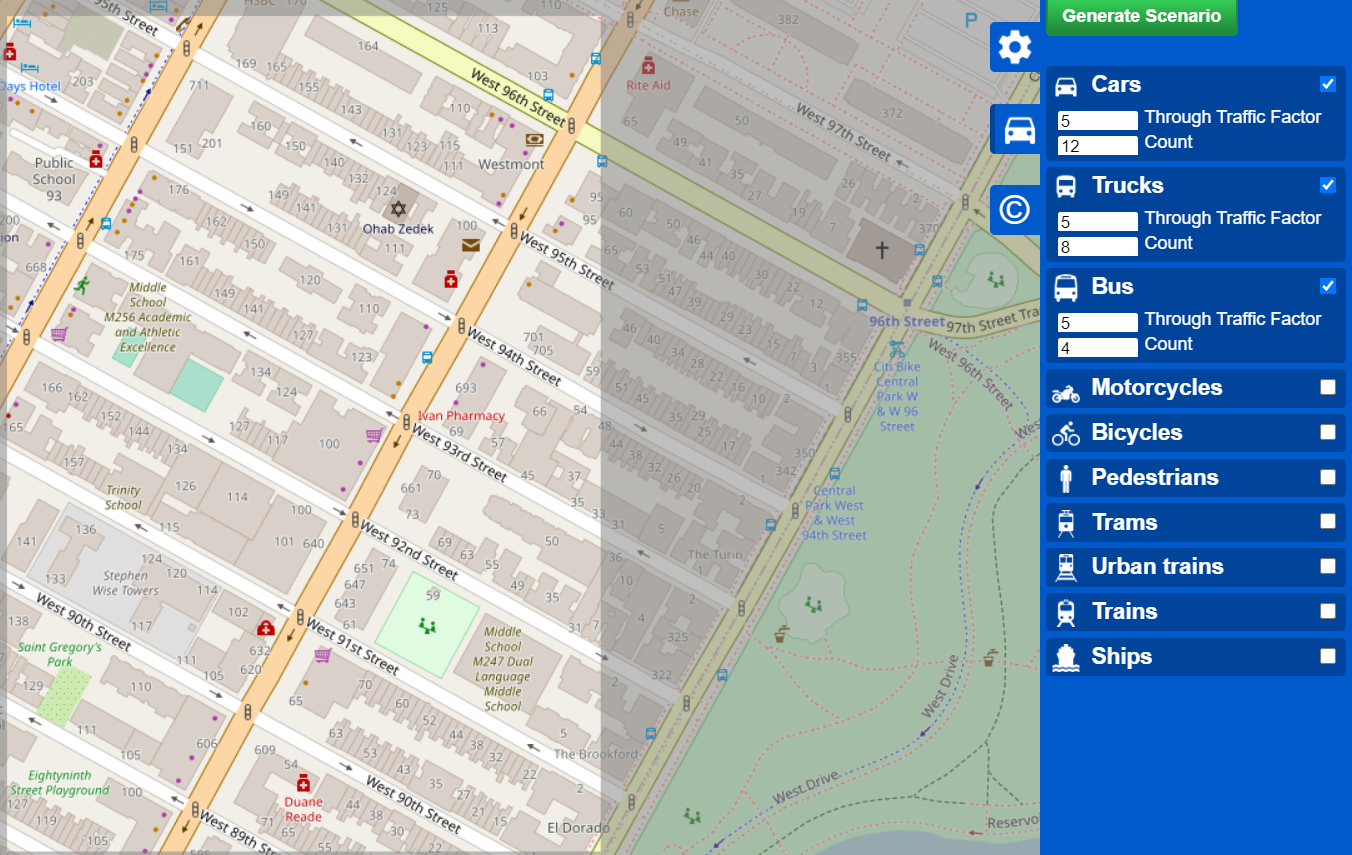
<https://sumo.dlr.de/docs/Tutorials/OSMWebWizard.html#demand_generation>

如下图所示，初始情况下交通流与车辆行驶状态由Through Traffic Factor和count两个参数决定。

Through Traffic Factor：车辆产生时，OSM Web Wizard会随机为车辆选择出发与到达的道路。该参数值越大，说明相比完全在地图中的整条道路，更多的车辆会从地图边缘的道路出发/离开，即过境交通量越大。

Count: 该参数结合道路长度与车道数，可用于计算某种车辆的小时交通量。

以car为例，假设地图中car可以行驶的道路总长度为5km，每条道路有2条车道，car的count数值设为90，则每小时将会生成5\*2\*90=900辆。相当于用randomTrips.py产生交通流时，将参数p设为4（=3600/900），即每4秒出发一辆新car。



1. **路网随机交通流生成：**

randomTrips.py：生成trips.xml

--route-file：生成rou.xml

相关说明网页：

<https://sumo.dlr.de/docs/Tools/Trip.html>

命令行：

python randomTrips.py -n osm.net.xml -o Trips.trips.xml -p 10 --route-file newRoutes.rou.xml --validate

其中osm.net.xml为仿真所用net.xml文件，可由OpenStreetMap生成，Trips.trips.xml为输出的trip文件，-p参数代表每x秒生成一辆新车，--route-file为根据random trips生成route文件，并去除不连续的trips。--validate命令用于生成validate trips。

可增加--vehicle-class bus --trip-attributes="maxSpeed=\"27.8\"”等以增加车辆类型，速度等属性。

1. **SUMO 交通信号灯状态获取与修改：**

TraCI/Change Traffic Lights State：

<https://sumo.dlr.de/docs/TraCI/Change_Traffic_Lights_State.html>

Traci Traffic Light相关函数：

<https://sumo.dlr.de/pydoc/traci._trafficlight.html#TrafficLightDomain-getIDList>

利用traci可获取交通信号灯信息包括：

IDList，

IDCount(场景中交通信号灯总数)，

programLogic

PhaseDuration

PhaseName

PhaseIndex

controlledLanes

controlledLinks等

1. **SUMO交叉口信息获取：**

TraCI/Junction Value Retrieval：

<https://sumo.dlr.de/docs/TraCI/Junction_Value_Retrieval.html>

利用traci可获取junction信息包括：

IDList: 场景中所有junctionID

IDCount: 场景中所有junction总数

Position：指定junction的位置坐标

Shape：指定junction的形状描述坐标点集

1. **获取Plain XML文件**

使用netedit.exe打开osm.net.xml并保存Plain XML文件，可以获得仿真场景中所有edge, node, connection, traffic light相关信息的文件。

（或使用命令行netconvert --osm my\_osm\_net.xml --plain-output-prefix osm\_net）

1. **SUMO坐标转换与导出**

<https://sumo.dlr.de/docs/Geo-Coordinates.html>

使用以下命令导出含经纬度坐标文件

netconvert --sumo-net-file myNet.net.xml --plain-output-prefix plain --proj.plain-geo

1. **SUMO仿真中坐标相关操作**

Tools/Sumolib：

坐标转换，路网中node, edge等对象位置获取等

<https://angelobanse.github.io/sumo-documentation2/Tools/Sumolib.html>

<https://sumo.dlr.de/daily/pydoc/sumolib.html>

1. **SUMO Importing additional Polygons**

<https://sumo.dlr.de/docs/Networks/Import/OpenStreetMap.html>

1. **后台调用sumo.exe并在仿真结束后自动退出程序(--quit-on-end)**

sumoBinary = "D:/SUMO 1.5.0/bin/**sumo.exe**" # sumo-gui.exe for starting GUI

sumoCmd = [sumoBinary, "-c", "D:\ShuyangDongDocument\\UVA\\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\\NewYork\\NewYork.sumocfg","--queue-output", "SUMODataOutput\\queue.xml","**--quit-on-end**"]

1. **OSM地图转化为SUMO地图**

<https://sumo.dlr.de/docs/Networks/Import/OpenStreetMap.html>

netconvert --osm-files berlin.osm.xml -o berlin.net.xml

Recommended netconvert Options：

--geometry.remove --ramps.guess --junctions.join --tls.guess-signals --tls.discard-simple --tls.join --tls.default-type actuated

命令行示例：

D:\SUMO 1.5.0\bin:

**Convert OSM map:**

netconvert --osm-files D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkMap.osm -o D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkMap.net.xml --geometry.remove --ramps.guess --junctions.join --tls.guess-signals --tls.discard-simple --tls.join --tls.default-type actuated

netconvert --osm-files D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\map.osm -o D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\map.net.xml --geometry.remove --ramps.guess --junctions.join --tls.guess-signals --tls.discard-simple --tls.join --tls.default-type actuated

**Importing additional Polygons (Buildings, Water, etc.):**

polyconvert --xml-validation never --net-file D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkMap.net.xml --osm-files D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkMap.osm --type-file D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\osmPolyconvert.typ.xml -o D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkMap.poly.xml

polyconvert --xml-validation never --net-file D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYork.net.xml --osm-files D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\mappolygon.osm --type-file D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\osmPolyconvert.typ.xml -o D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYork11.poly.xml

**Write sumo configuration file:**

<https://sumo.dlr.de/docs/Basics/Using_the_Command_Line_Applications.html>

**Generate random trips:**

<https://sumo.dlr.de/docs/Tools/Trip.html>

To let n vehicles depart between times t0 and t1 set the options

-b t0 -e t1 -p ((t1 - t0) / n)

D:\SUMO 1.5.0\tools

python randomTrips.py -n D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYork.net.xml -o D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkTrips.trips.xml -b 0 -e 3600 -p 0.8 --fringe-factor 10 --route-file D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkRoutes.rou.xml --validate

<https://www.eclipse.org/lists/sumo-user/msg02295.html>

--flows 100

python randomTrips.py -n D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYork.net.xml -o D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkTrips.trips.xml --route-file D:\ShuyangDongDocument\UVA\UVA-Research\SmartCitySimulation\SUMOSimulation\SmartCitySUMOSim\NewYork\NewYorkRoutes.rou.xml --fringe-factor 100 --flows 10 --validate