



SUMO Tutorial

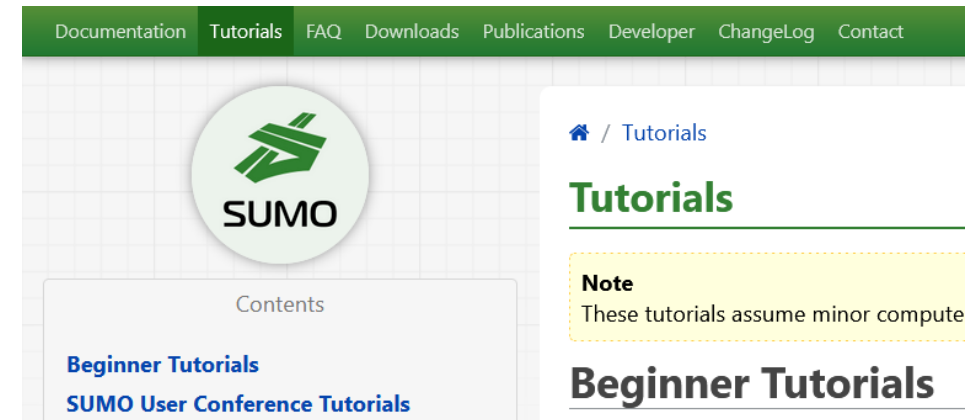
Jakob Erdmann

SUMO2025, Berlin

Outline



- 3-Click scenario generation with osmWebWizard.py
- Railway simulation
 - GTFS
 - Passengers
- Importing detector data
- Electric vehicles and automatic charging

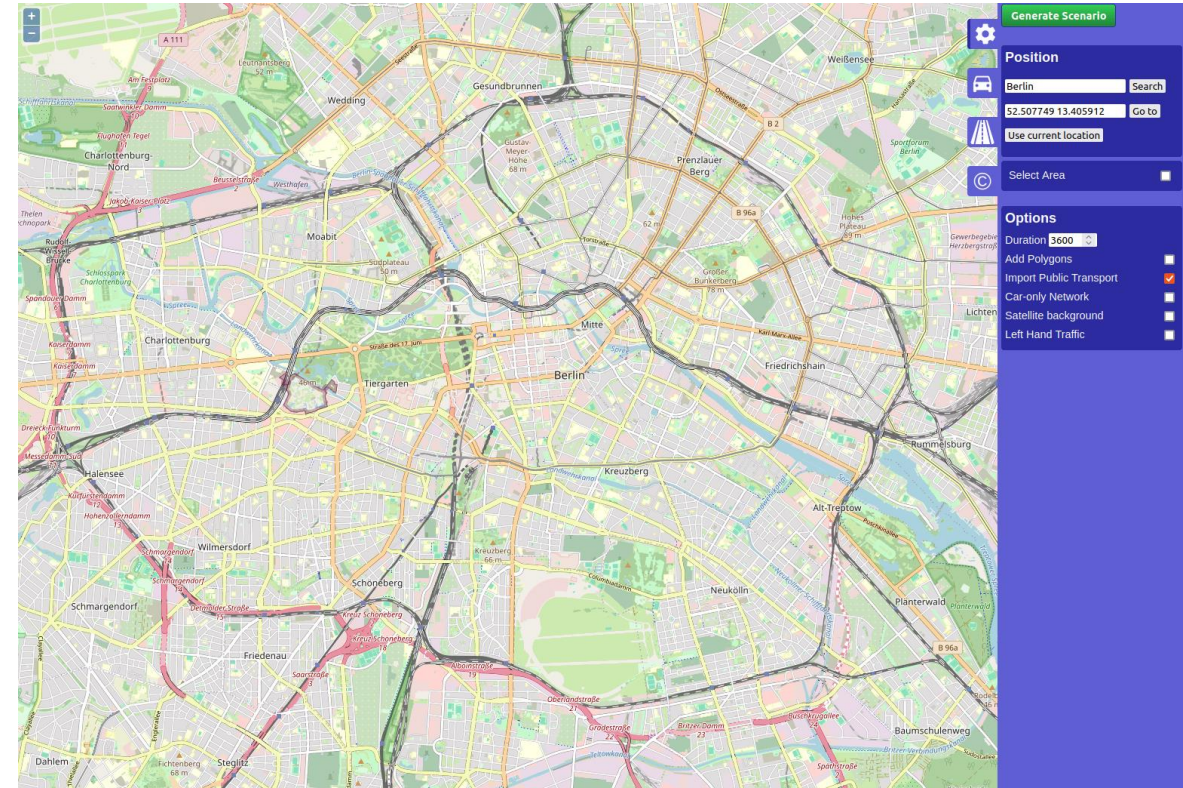


- **Prerequisites**
 - SUMO 1.23.0+
 - Python: python.org/download/
 - Data files: sumo.dlr.de/daily/sumo2025_tutorial.zip

osmWebWizard – Berlin S-Bahn



- [tools/osmWebWizard.py](#)
- OpenStreetMap network data
- **Random traffic**
- Configure
 - Area: inner ring
 - road types: Railway: light_rail
 - ~~Traffic modes~~
 - ~~Traffic volume~~
 - ~~Fraction of through-traffic~~
 - Public Transport
 - Scenario duration
 - ~~Building Shapes and Points-of-Interest (cosmetic)~~
- Generated files allow rebuilding and adapting the scenario
- Example data in 0_wizard



Berlin S-Bahn simulation - Signals



- Warning: **Block ... exceeds maximum length**
 - Only relevant for [single-track operations](#)
 - Can be suppressed with **--railsignal.max-block-length 30000**
- Warning: **Teleporting vehicle ... waited too long**
 - Signals divide track network into blocks
 - Train may only enter a signal block that isn't occupied by another train
 - (except when [portion working](#))
 - **Signals perform many additional safety and deadlock-prevention checks!**
 - OSM is missing some signals → longer blocks → unrealistic wait times
 - Can be fixed with **netconvert** option **--railway.signal.guess.by-stops**
 - Places a signal before and after every public transport stop where missing
 - Call **build.bat** to adapt train routes (network was split to place signals)
- `0_wizard/2.netccfg`

Berlin S-Bahn simulation - Schedule



- Webwizard only uses OSM data: no schedule timing, only routes and service interval
 - Spacing between trains of different lines is random/wrong
- Use GTFS! (provided by [VBB](#))
 - **General Transit Feed Specification:** data interchange standard for public transport schedules established by Google.
 - Tutorial contains trimmed-down version with S-Bahn data only (see file `trim_gtfs.sh` for more information)
- **SUMO/tools/import/gtfs/gtfs2pt.py**
 - gtfs GTFS_sbahn_berlin.zip**
 - date 20250512**
 - network osm.net.xml.gz**
 - stops osm_stops.add.xml**
- **Loaded Stops establish the default track direction / track choice**
 - `1_gtfs/import.bat`
 - `1_gtfs/gtfs.sumocfg`

Berlin S-Bahn simulation - Schedule



- **Problem:** VBB includes latest disruptions whereas OSM does not
 - Current disruption forces some tracks to be used in the non-standard direction
 - osmWebWizard only ensured usable track direction/stops for the OSM routes
 - **--railway.topology.all-bidi** (after running netconvert, rerun build.bat)
 - Define 2 extra stops (**disruption_stops.add.xml**)
 - **SUMO/tools/import/gtfs/gtfs2pt.py**
 - gtfs GTFS_sbahn_berlin.zip**
 - date 20250512**
 - network osm.net.xml.gz**
 - stops osm_stops.add.xml, disruption_stops.add.xml**
 - **Problem 2:** Loaded Stops **not always** establish the default track direction / track choice. Some tracks are ambiguously used in both directions
 - Invalid track choice degrades network performance / punctuality
 - Manual fixing still required
 - Example data in 1_gtfs
- 0_wizard/3.netccfg
 - 0_wizard/build.bat
 - 1_gtfs/import.bat
 - 1_gtfs/gtfs.sumocfg

Berlin S-Bahn simulation - Passengers



- Passengers may need to change trains
- Network needs some infrastructure to connect platforms
 - **SUMO/tools/net/generateStationEdges.py**
 - **netconvert -c stations.netccfg**
 - (see build.bat)
- Generate random persons that use public transport

```
<configuration>
  <begin value="7:0:0"/>
  <end value="8:0:0"/>
  <net-file value="stations.net.xml.gz"/>
  <additional-files
value="stations.add.xml, ../1_gtfs/vtypes.xml, ../1_gtfs/gtfs_routes.add.xml"/>
  <route-file value="persons.rou.xml"/>
  <persontrips value="True"/>
  <from-stops value="busStop"/>
  <to-stops value="busStop"/>
  <trip-attributes value="modes='public'"/>
  <validate value="True"/>
</configuration>
```

Berlin S-Bahn simulation - Passengers



- SUMO/tools/route/analyzePersonPlans.py:

...

156: stop walk public

157: stop walk public walk public public walk

159: stop walk public public walk

161: stop walk public walk public walk public walk

163: stop public walk

211: stop public walk public walk

217: stop walk public walk public

656: stop walk public walk

813: stop walk public walk public walk

- New: Passengers default to taking any line that takes them to their destination
- Example data in `2_passengers`

osmWebWizard limitations for Berlin S-Bahn simulation

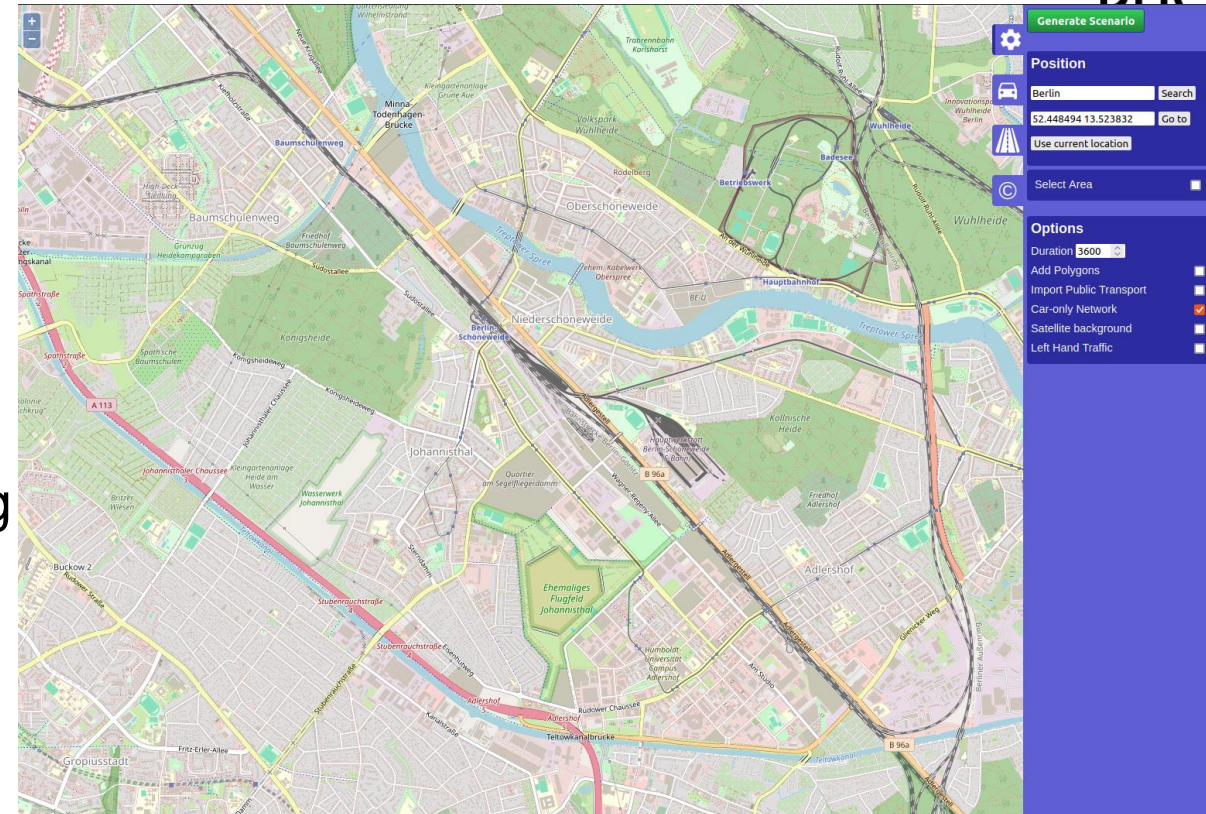


- Filtering by “light_rail” retrieves too little: some tracks are shared with regional trains and marked as “rail” (does not affect the inner ring scenario)
- Filtering by “light_rail” retrieves too much: service yards, industrial usage, ...
- Better to retrieve more tracks and additional usage information. Example:
<https://github.com/DLR-TS/sumo-berlin> (from build.py)
 - --type-files ...,sumo/data/typemap/osmNetconvertRailUsage.typ.xml
 - --keep-edges.by-type railway.light_rail|usage.main,railway.rail|usage.main. ...
 - --keep-edges.by-vclass rail_urban

osmWebWizard – Berlin, Treptow-Köpenick



- [tools/osmWebWizard.py](#)
- OpenStreetMap network data
- ~~Random traffic~~
- Configure
 - Area: part of Berlin, includes DLR
 - road types: Only roads
 - Car-only Network
- Generated files allow rebuilding and adapting
- Extra options in 2.netccfg:
 - `<osm.turn-lanes value="true"/>`
 - `<parking-output value="parking.add.xml"/>`
- Example data in 3_wizard



Berlin, counting data



- Public traffic data from the city of Berlin: [Digitale Plattform Stadtverkehr Berlin](#)
 - Location of detectors (xlsx -> save as .csv)
 - Hourly Counting data (csv, one file per month)

G308 zwischen Kanalstraße und AS Stubenrauchstraße														
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	MQ_KURZNAME	DET_NAME_ALT	DET_NAME_NEU	DET_ID15	MQ_ID15	STRASSE	POSITION	POS_DETAIL	RICHTUNG	SPUR	annotation	LANGE (WGS84)	BREITE (WGS84)	INBETRIEBN
2	TE001	TEU00002_Det0	TE001_Det_HF1	100101010000167	1.002E+14	A115	AS Spanische	AK Zehlend	Südwest	HF_R	Hauptfahrb	13.19257786	52.43386833	2/18/2003
3	TE001	TEU00002_Det1	TE001_Det_HF2	100101010000268	1.002E+14	A115	AS Spanische	AK Zehlend	Südwest	HF_2vR	Hauptfahrb	13.19257786	52.43386833	2/18/2003
4	TE002	TEU00002_Det2	TE002_Det_HF1	100101010000369	1.002E+14	A115	AS Spanische	AD Funktur	Nordost	HF_R	Hauptfahrb	13.1927467	52.43381256	2/18/2003
5	TE002	TEU00002_Det3	TE002_Det_HF2	100101010000470	1.002E+14	A115	AS Spanische	AD Funktur	Nordost	HF_2vR	Hauptfahrb	13.1927467	52.43381256	2/18/2003
6	TE004	TEU00004_Det0	TE004_Det_HF1	100101010000874	1.002E+14	Clayallee	zwischen Scha	Potsdamer	Süd	HF_R	Hauptfahrb	13.26130077	52.43664151	2/18/2003
7	TE004	TEU00004_Det1	TE004_Det_HF2	100101010000975	1.002E+14	Clayallee	zwischen Scha	Potsdamer	Süd	HF_2vR	Hauptfahrb	13.26130077	52.43664151	2/18/2003
8	TE005	TEU00005_Det0	TE005_Det_HF1	100101010001076	1.002E+14	Berliner Stra	zwischen Seeh	Clayallee	West	HF_R	Hauptfahrb	13.26310552	52.43511242	2/19/2003
9	TE005	TEU00005_Det1	TE005_Det_HF2	100101010001177	1.002E+14	Berliner Stra	zwischen Seeh	Clayallee	West	HF_2vR	Hauptfahrb	13.26310552	52.43511242	2/19/2003
10	TE006	TEU00006_Det0	TE006_Det_HF1	100101010001278	1.002E+14	Teltower Da	zwischen Kirch	Berliner Stra	Nord	HF_R	Hauptfahrb	13.25988057	52.43374142	2/18/2003

```
mq_name;tag;stunde;qualitaet;q_kfz_mq_hr;v_kfz_mq_hr;q_pkw_mq_hr;v_pkw_mq_hr;q_lkw_mq_hr;v_lkw_mq_hr
TE001;13.05.2024;0;1.0;195;61;166;61;29;68
TE001;13.05.2024;1;1.0;114;57;98;55;16;70
TE001;13.05.2024;2;1.0;101;66;76;64;25;75
TE001;13.05.2024;3;1.0;166;62;122;59;44;71
```

Berlin, counting data import



- Trim data to a single day: `grep "13.05.2024\"mq_name" mq_hr_2024_05.csv`
- Replaced Umlaut in column name to avoid Windows encoding issues
- Build traffic scenario (4_counts/build.bat):
 - `tools/detector/mapDetectors.py -n osm.net.xml.gz -d Stammdaten.csv -o det.add.xml -i MQ_KURZNAME -x "LAENGE (WGS84)" -y "BREITE (WGS84)" --max-radius 20 --delimiter , --write-params`
 - `tools/detector/edgeDataFromFlow.py -d det.add.xml -f mq_hr_2024_05_13.csv -q q_pkw_mq_hr,q_lkw_mq_hr --time-column stunde --id-column mq_name --time-scale 3600 -o countdata.xml -i 1`
 - `tools/randomTrips.py -n ../3_wizard/osm.net.xml.gz --min-distance 2000 -e 2:0:0 -r random.rou.xml --fringe-factor 10`
 - `tools/routeSampler.py -r random.rou.xml -d countdata.xml --edgedata-attribute q_pkw_mq_hr -o sampled.rou.xml -I -b 6:0:0 -a "departSpeed=\"avg\" departLane=\"best\""`

Berlin, routes from counts



- 21600: Wrote 2624 routes (568 distinct) achieving total count 3136 (100.00%) at 17 locations. GEH<5 for 100.00%
- 25200: Wrote 2971 routes (590 distinct) achieving total count 3537 (100.00%) at 17 locations. GEH<5 for 100.00%
- ...
- avg interval input count: count 18, min 1066.00 (82800.0), max 3537.00 (25200.0), mean 2152.56, Q1 1419.00, median 2296.00, Q3 2738.00
- avg interval written vehs: count 18, min 839.00 (82800.0), max 2971.00 (25200.0), mean 1804.89, Q1 1303.00, median 1811.00, Q3 2266.00
- avg interval underflow: count 18, min 0.00 (21600.0), max 0.00 (21600.0), mean 0.00, Q1 0.00, median 0.00, Q3 0.00
- avg interval overflow: count 18, min 0.00 (21600.0), max 0.00 (21600.0), mean 0.00, Q1 0.00, median 0.00, Q3 0.00
- avg interval GEH%: count 18, min 100.00 (21600.0), max 100.00 (21600.0), mean 100.00, Q1 100.00, median 100.00, Q3 100.00
- Example data in 4_counts

```
<vType id="DEFAULT_VEHTYPE">  
  <param key="has.battery.device" value="true"/>  
  <param key="device.battery.chargeLevel" value="normc(5000,3000,1000,35000)"/>  
</vType>
```

```
Warning: Battery of vehicle '21600_1017' is depleted, time=23712.00.  
Warning: Battery of vehicle '21600_2077' is depleted, time=25370.00.  
Warning: Battery of vehicle '21600_2134' is depleted, time=25434.00.  
Warning: Battery of vehicle '25200_200' is depleted, time=26259.00.  
Warning: Battery of vehicle '25200_1874' is depleted, time=28375.00.
```

- Configured initial charge to ensure that some vehicles need to charge within the scenario
 - Device params support distributions!
- Device parameters can be used for coloring, text annotations

Electric Vehicles - Automated charging



- Build charging stations for the network (build.bat)
 - Charging stations are attached to 3% of parking areas

```
<configuration>
  <net-file value="../../../3_wizard/osm.net.xml.gz"/>
  <add-files value="../../../3_wizard/parking.add.xml"/>
  <output-file value="cstations.add.xml"/>
  <output-parking-file value="parking2.add.xml"/>
  <probability value="0.03"/>
  <power value="75000"/>
</configuration>
```

- Configure stationfinder device (run2.sumocfg)

```
<device.stationfinder.probability value="1"/>
<device.stationfinder.emptyThreshold value="0.0"/>
<device.stationfinder.needToChargeLevel value="0.1"/>
<device.stationfinder.radius value="600"/>
<device.stationfinder.reserveFactor value="1.25"/>
<device.stationfinder.saturatedChargeLevel value="0.3"/>
```

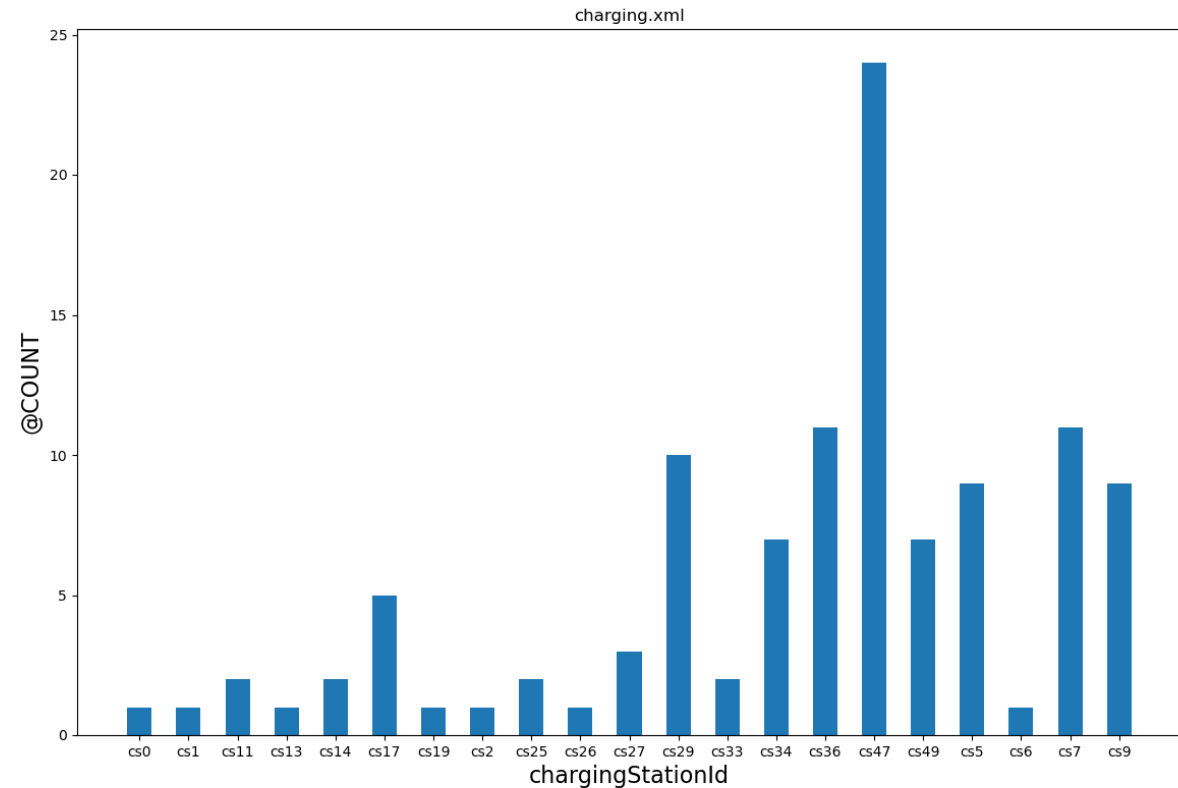
Defaults:

0.05
0.4
180 (s)
1.1
0.8

Electric Vehicles - Automated charging

- No more Battery depletion events
- 111 stops at chargingStation
- Possible outputs
 - --stop-output stopinfos.xml
 - --chargingstations-output charging.xml
 - --chargingstations-output.aggregated
- 21 out of 53 chargingStations were used

- Example data in 5_electric



Conclusion

- Use netedit to discover all the extra tools
- Use [tools/osmWebWizard.py](#) to get a quick start
- Read the documentation / FAQ at <http://sumo.dlr.de/docs>
- Report any bugs you find to sumo-user@eclipse.org
- Share your scenarios and results
- Talks to us. We are always looking for project partners! sumo@dlr.de

