



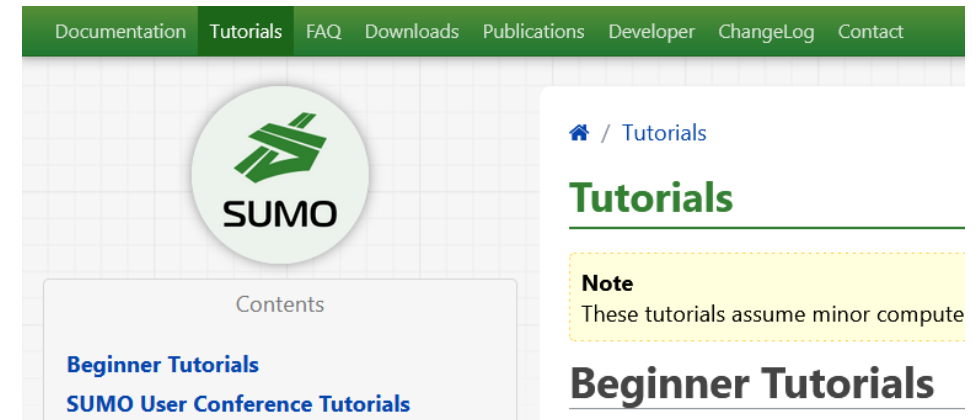
SUMO Tutorial

Jakob Erdmann

SUMO2026, Berlin

Outline

- Podcars
- State Saving/Loading
- Routing Preferences



- **Prerequisites**
 - SUMO 1.27.0+
 - Python: python.org/download/
 - Data files: sumo.dlr.de/daily/sumo2026_tutorial.zip

Podcars / Personal Rapid Transit (PRT)



- small, automated, on-demand public transport
- require less space and resources than roads and cars per passenger/km



[Das Cabinentaxi - Krauss Maffei Transurban \(1974\)](#),
not deployed

source: youtube



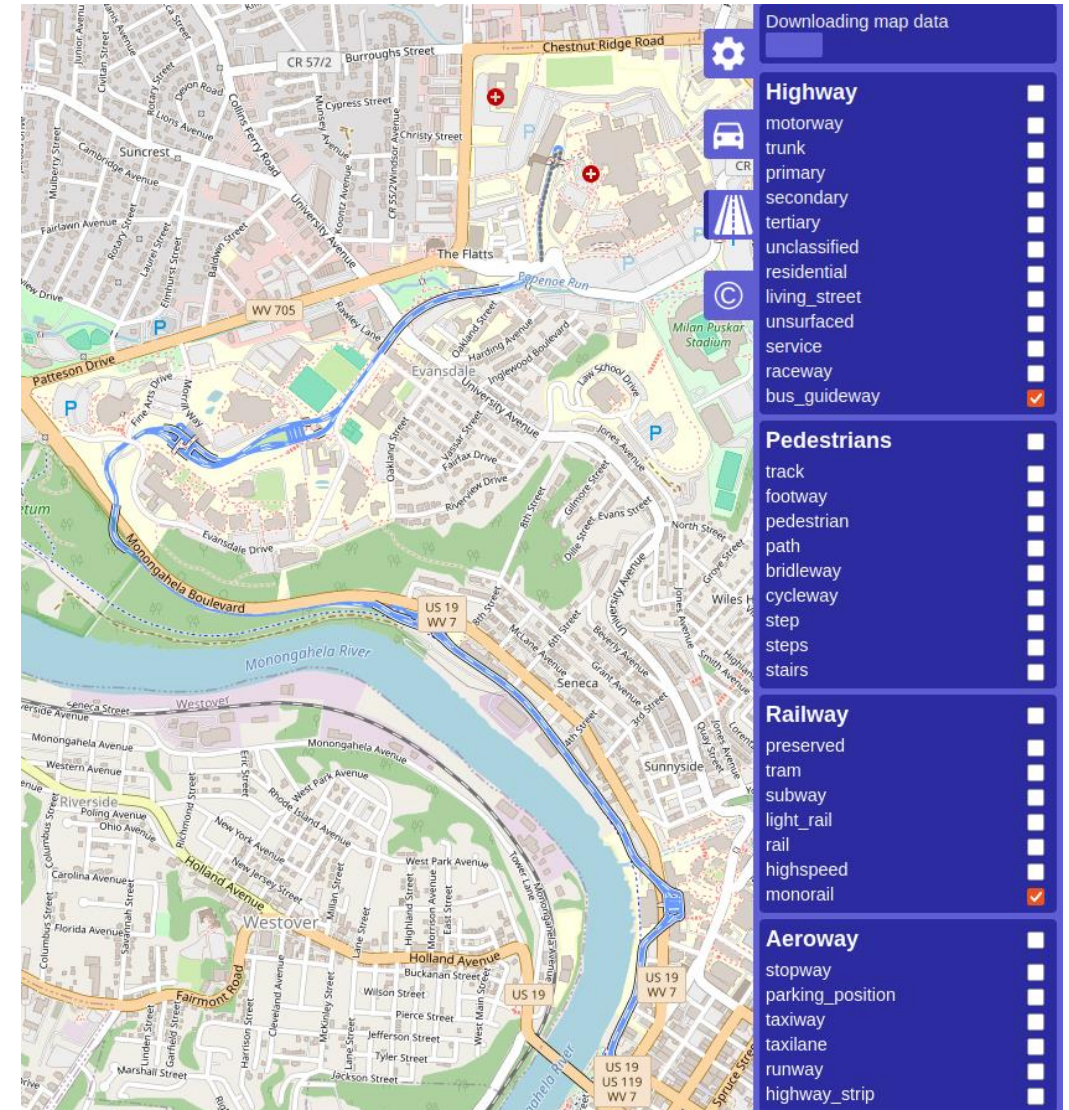
[Morgantown PRT](#)
operational since 1975
73 vehicles, 5.8km, 5 stations

source: wikipedia

osmWebWizard – Morgantown



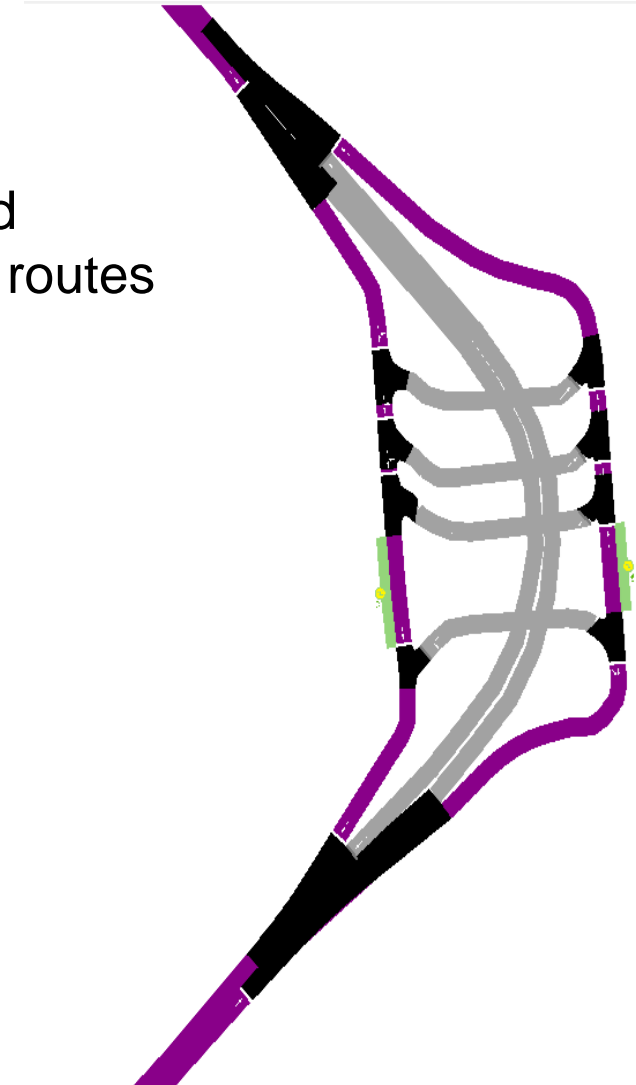
- [tools/osmWebWizard.py](#)
- OpenStreetMap network data
- Configure
 - Area: Morgantown
 - road types: bus_guideway, monorail
 - ~~Traffic modes~~
 - ~~Traffic volume~~
 - ~~Fraction of through-traffic~~
 - Public Transport
 - Scenario duration
 - ~~Building Shapes and Points-of-Interest (cosmetic)~~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



One-click scenario limitations

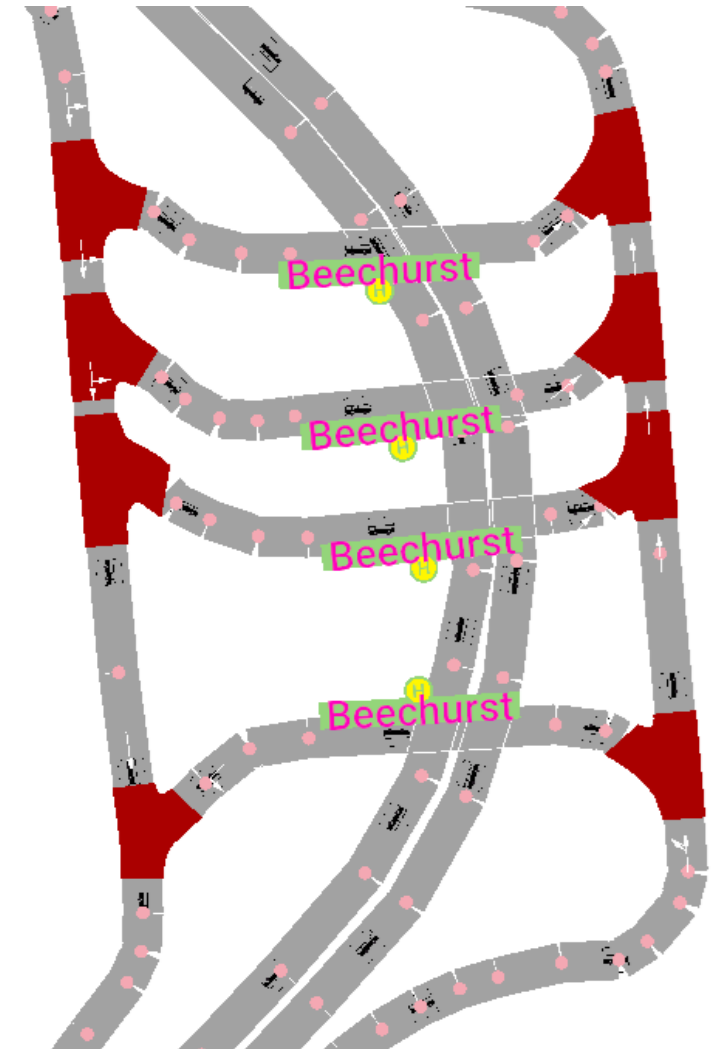


- OSM:
 - Inconsistent edge types (bus_guideway, monorail)
 - Wrong placement of stops: Off-line stations are not mapped
 - Public transport line does not reflect dynamic (on-demand) routes
 - Speeds not mapped
- Wizard
 - Vehicles disappear/spawn after driving one way (--join)
 - Line-based vehicles are the wrong model anyway



Using netedit to fix infrastructure

- Load scenario in netedit (Ctrl+T from sumo-gui)
 - Selection mode (S)
 - Select all edges: shift-drag rectangle or *apply selection*
 - Inspection mode (I)
 - Inspect arbitrary edge -> inspects the selection
 - Allow dialog: bus
 - Speed: 14.75
 - Clear selection (Esc in selection mode, shift + Esc in any mode)
 - Additional mode (A): busStop
 - Create new Off-line busStops
 - Copy the name from the original stops!
 - Delete mode (D): remove old On-line stops
 - save
-
- Example data in `1_netedit`



Modelling PRT Vehicles

- SUMO provides taxi device for modelling demand-responsive public transport
- First presented during the 2020 tutorial, new features since then:
 - Mix of arbitrary vClasses in intermodal routing (drones, rail, ...)
 - Different persons restricted to different vehicle fleets
 - Idle taxis waiting at taxi stand
 - instant re-dispatch once a taxi becomes idle
 - Reservations (pre-booking)
 - State-saving and loading
 - many stability/algorithm fixes



- Example data in `2_taxis`





Defining PRT Vehicles in Netedit

- Start with a fresh sumocfg: open only network in netedit (Ctrl+O)
 - Alternative: Ctrl+Shift+T in sumo-gui
 - load osm_stops.add.xml (Ctrl+A)
- Demand super-mode: F3
- Define a taxi type

Edit parameters

Parameters	
key	value
has.taxi.device	true
device.taxi.dropOffDuration	0
device.taxi.parking	false

Edit vType 'taxi'

Vehicle Type attributes			
vClass		guiShape	
 bus		bus	
id	taxi	probability	1.00
 color		personCapacity	8
length	4.72	containerCapacity	0

Types

Type Editor

Create Type



Delete Type

Copy Type

Current Type

taxi

Internal attributes

id	taxi
vClass	 bus
 color	
length	4.72
minGap	1.00
maxSpeed	15.00
desiredMaxSpeed	2777.78
parkingBadges	
probability	1.00
accel	0.6
decel	3
sigma	0
tau	1.00



- Example data in 2_taxis

Defining PRT Vehicles in Netedit

- Demand super-mode: F3
- Define a taxi type
- Define taxis that initialize all over the net:
 - Define a long route (R), (bus)
 - Repeated clicks on waypoint edges
 - <enter> to accept



- Example data in 2_taxis

Defining PRT Vehicles in Netedit

- Demand super-mode: F3
- Define a taxi type
- Define taxis that initialize all over the net:
 - Define a long route (R), (bus)
 - Define a flow (V) that uses the route, departEdge="random", departPos="free"
 - Use the new taxi type
- Save as taxi.rou.xml

- Example data in 2_taxis

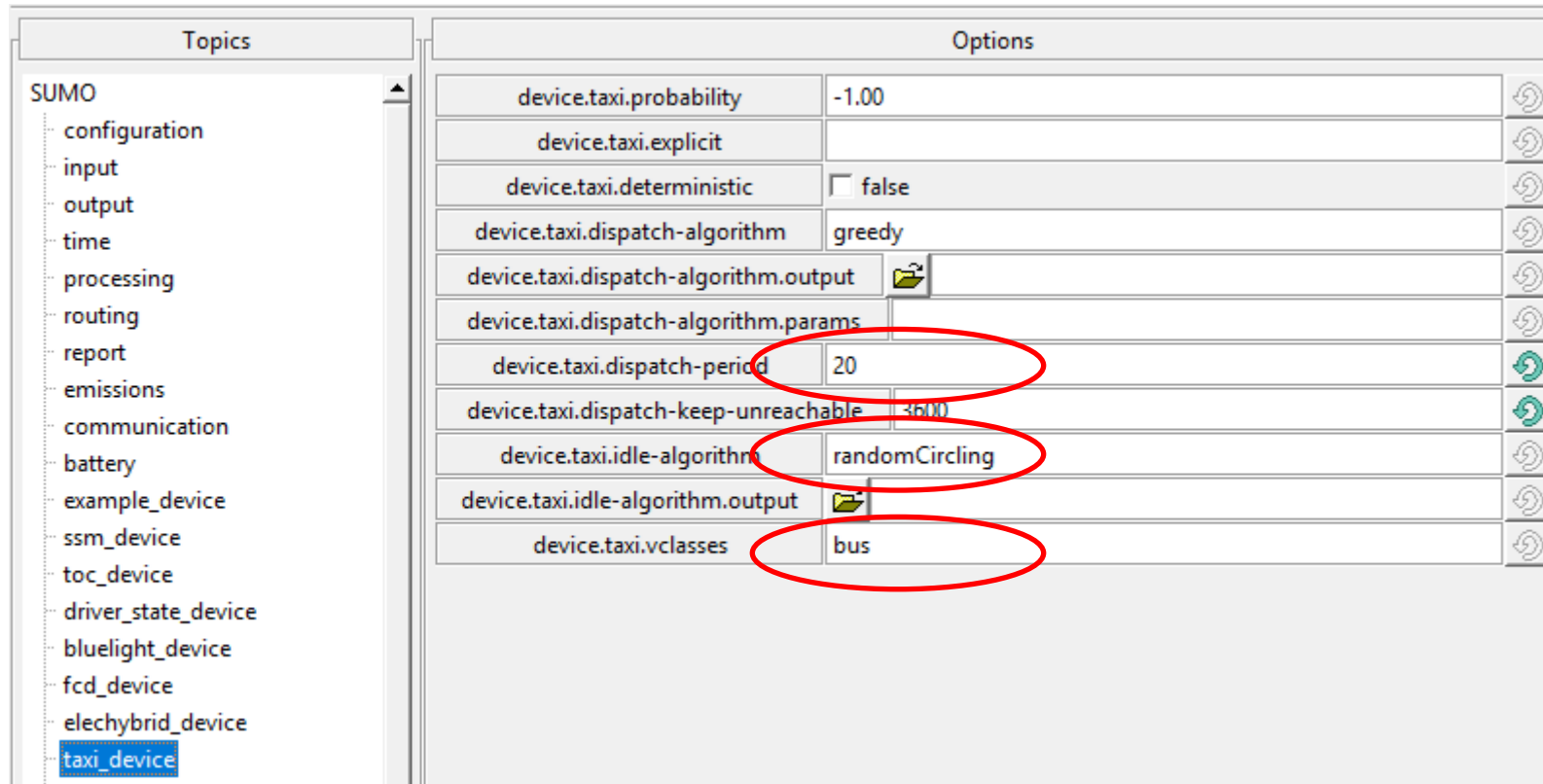


The screenshot shows the 'Vehicles' configuration window in Netedit. It is set to 'flow (over route)' mode. The 'Parent vType' is 'taxi'. Under 'Internal attributes', several fields are highlighted with red circles: 'departEdge' is set to 'random', 'departPos' is set to 'free', and in the 'Flow attributes' section, 'terminate' is set to 'end-number', 'end' is set to '0.00', and 'number' is set to '69'.

Vehicles	
Vehicles	
flow (over route)	
Parent vType	
taxi	
Internal attributes	
id	pod
color	yellow
departEdge	random
arrivalEdge	
departLane	first
departPos	free
departSpeed	avg
arrivalLane	current
arrivalPos	max
arrivalSpeed	current
line	
personNumber	0
containerNumber	0
departPosLat	center
arrivalPosLat	center
insertionChecks	all
begin	0.00
Flow attributes	
terminate	end-number
end	0.00
number	69

Defining PRT Vehicles in Netedit

- Configure further taxi-related options (shift-F10)

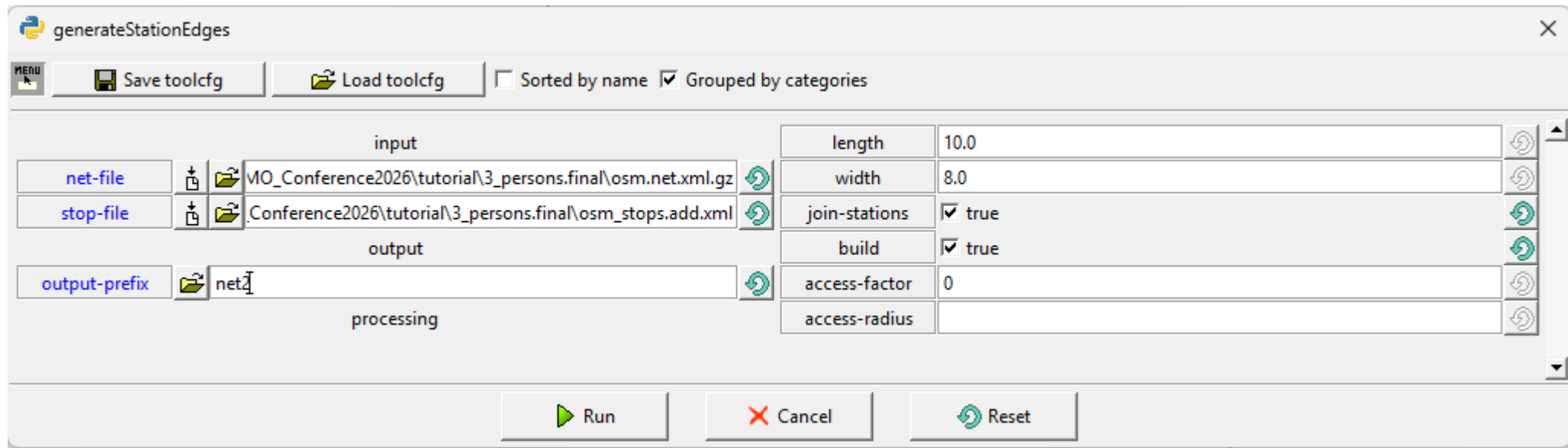


- Save sumocfg
- Example data in 2_taxis

Defining Passenger demand in Netedit



- randomTrips to define demand
 - Only guideways, no edges for pedestrians
 - Start at stops? Random stop likely wrong for the desired direction!
- generateStationEdges.py

The screenshot shows the 'generateStationEdges' tool window. It has a menu icon, 'Save toolcfg', 'Load toolcfg', and checkboxes for 'Sorted by name' and 'Grouped by categories'. The 'input' section contains 'net-file' (MO_Conference2026\tutorial\3_persons.final\osm.net.xml.gz) and 'stop-file' (Conference2026\tutorial\3_persons.final\osm_stops.add.xml). The 'output' section has 'output-prefix' (net2). The 'processing' section includes a table of parameters: length (10.0), width (8.0), join-stations (checked true), build (checked true), access-factor (0), and access-radius (empty). At the bottom are 'Run', 'Cancel', and 'Reset' buttons.

input	
net-file	MO_Conference2026\tutorial\3_persons.final\osm.net.xml.gz
stop-file	Conference2026\tutorial\3_persons.final\osm_stops.add.xml

output	
output-prefix	net2

processing	
length	10.0
width	8.0
join-stations	<input checked="" type="checkbox"/> true
build	<input checked="" type="checkbox"/> true
access-factor	0
access-radius	

- Open generated network and stops (net2.net.xml.gz, net2_stops.add.xml)
- Load demand again (taxi.rou.xml)
- Overwrite sumocfg (other options are kept)
- Example data in 3_persons

Defining Passenger demand in Netedit

- randomTrips to define demand
 - Disable validation because duarouter disallows bus edges for taxi (by default)
 - Netedit doesn't yet support passing extra duarouter options (--taxi.vclasses)
- Load persons.rou.xml in netedit (Ctrl+D)
- Set taxi-transfer options (shift-F10)
- Set ride sharing option

routing	
report	
emissions	
communication	
battery	
example_device	
ssm_device	
toc_device	
driver_state_device	
bluelight_device	
fcd_device	
	weights.priority-factor 0.00
	weights.separate-turns 0.00
	astar.all-distances
	astar.landmark-distances
	persontrip.walkfactor 0.75
	persontrip.transfer.car-walk parkingAreas
	persontrip.transfer.taxi-walk ptStops
	persontrip.transfer.walk-taxi ptStops
	persontrip.default.group sharing

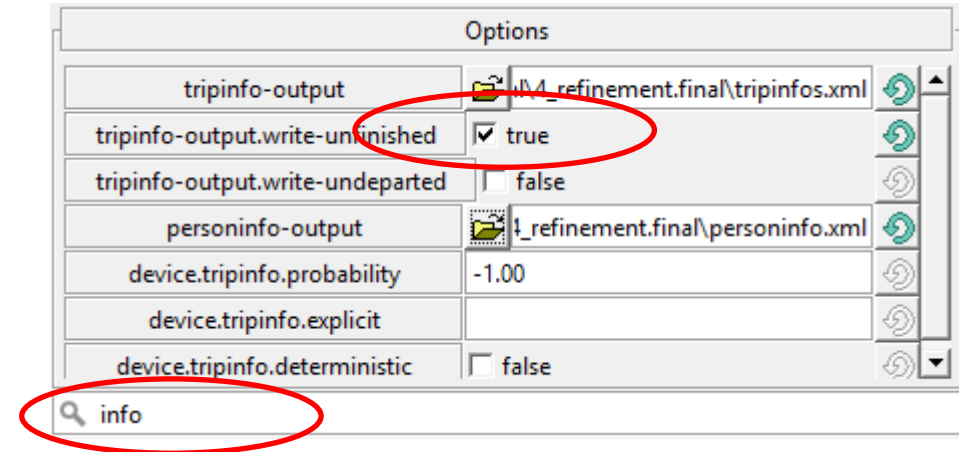
- Set simulation end to 4200 (taxis would keep circling otherwise)
- Save sumocfg
- Example data in 3_persons

input	
net-file	VO_Conference2026\tutorial\3_persons.final\net2.net.xml.gz
additional-files	Conference2026\tutorial\3_persons.final\net2_stops.add.xml
weights-prefix	
edge-type-file	
output	
output-trip-file	persons.rou.xml
route-file	
vtype-output	
weights-output-prefix	
error-log	
processing	
vehicle-class	
seed	42
random	<input type="checkbox"/> false
min-distance	100
min-distance.fringe	
max-distance	
intermediate	0
jtrouter	<input type="checkbox"/> false
maxtries	100
remove-loops	<input type="checkbox"/> false
random-routing-factor	1
marouter	<input type="checkbox"/> false
validate	<input type="checkbox"/> false
threads	
min-success-rate	0.1
verbose	<input type="checkbox"/> false
poisson	<input type="checkbox"/> false
persons	
persons	<input checked="" type="checkbox"/> true
personrides	
persontrip.modes	taxi

Scenario Evaluation

- Add options for statistic-output, log-output, tripinfo-output (including write-unfinished), personinfo-output, stop-output, (including write-unfinished)
- Use search box to find options quickly
- Don't forget to save the .sumocfg (Ctrl+Shift+S)
- Run simulation
- log.txt:

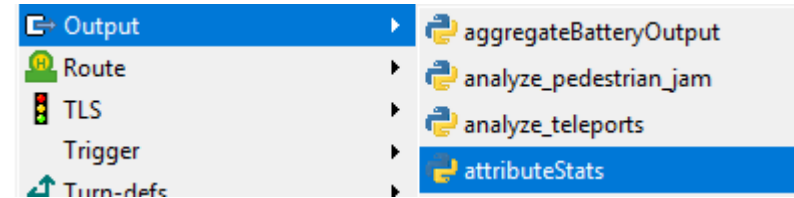
```
Ride Statistics (avg of 3600 rides):  
WaitingTime: 66.58  
RouteLength: 2540.36  
Duration: 198.53
```



- Example data in 4_refinement

Scenario Evaluation

- Call attributeStats.py on tripinfos.xml



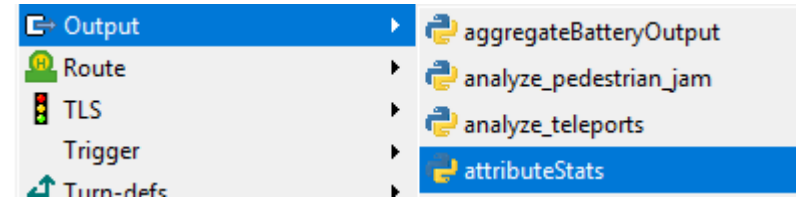
```
Python Tool
Console output

python "D:\erdm_ja\sumo-dev\tools\output\attributeStats.py" "D:\erdm_ja\sip-svn-trunk\papers\DLR\SUMO_Conference2026\tutorial\4_refinement.final\tripinfo
starting process...
taxi customers: count 69, min 31.00, max 77.00, mean 52.17, Q1 46.00, median 52.00, Q3 56.00, stdDev 8.57
taxi occupiedDistance: count 69, min 21357.93, max 36264.83, mean 26495.38, Q1 24819.10, median 26330.23, Q3 27858.94, stdDev 2528.31
taxi occupiedTime: count 69, min 1771.00, max 2787.00, mean 2118.28, Q1 1996.00, median 2126.00, Q3 2212.00, stdDev 174.52
tripinfo arrival: count 69, min -1.00 (pod.0), max -1.00 (pod.0), mean -1.00, Q1 -1.00, median -1.00, Q3 -1.00, stdDev 0.00
tripinfo arrivalPos: count 69, min -1.00 (pod.0), max -1.00 (pod.0), mean -1.00, Q1 -1.00, median -1.00, Q3 -1.00, stdDev 0.00
tripinfo arrivalSpeed: count 69, min -1.00 (pod.0), max -1.00 (pod.0), mean -1.00, Q1 -1.00, median -1.00, Q3 -1.00, stdDev 0.00
tripinfo depart: count 69, min 0.00 (pod.0), max 20.00 (pod.33), mean 1.33, Q1 0.00, median 0.00, Q3 0.00, stdDev 3.37
tripinfo departDelay: count 69, min 0.00 (pod.0), max 20.00 (pod.33), mean 1.33, Q1 0.00, median 0.00, Q3 0.00, stdDev 3.37
tripinfo departPos: count 69, min 0.00 (pod.0), max 148.71 (pod.55), mean 22.52, Q1 0.00, median 0.00, Q3 49.57, stdDev 38.15
tripinfo departSpeed: count 69, min 9.85 (pod.46), max 14.75 (pod.0), mean 14.27, Q1 14.75, median 14.75, Q3 14.75, stdDev 1.05
tripinfo duration: count 69, min 4180.00 (pod.33), max 4200.00 (pod.0), mean 4198.67, Q1 4200.00, median 4200.00, Q3 4200.00, stdDev 3.37
tripinfo rerouteNo: count 69, min 54.00 (pod.21), max 112.00 (pod.50), mean 78.39, Q1 72.00, median 78.00, Q3 85.00, stdDev 11.41
tripinfo routeLength: count 69, min 49056.66 (pod.24), max 54309.84 (pod.38), mean 51839.25, Q1 51395.17, median 51898.47, Q3 52430.31, stdDev 1043.99
tripinfo speedFactor: count 69, min 1.00 (pod.0), max 1.00 (pod.0), mean 1.00, Q1 1.00, median 1.00, Q3 1.00, stdDev 0.00
tripinfo stopTime: count 69, min 34.00 (pod.28), max 86.00 (pod.42), mean 56.80, Q1 52.00, median 56.00, Q3 62.00, stdDev 8.83
tripinfo timeLoss: count 69, min 207.45 (pod.19), max 548.59 (pod.24), mean 345.06, Q1 292.56, median 341.33, Q3 394.13, stdDev 69.20
tripinfo waitingCount: count 69, min 3.00 (pod.58), max 14.00 (pod.45), mean 7.90, Q1 6.00, median 8.00, Q3 9.00, stdDev 2.32
tripinfo waitingTime: count 69, min 6.00 (pod.57), max 250.00 (pod.24), mean 81.16, Q1 52.00, median 75.00, Q3 99.00, stdDev 45.83
process finished
```

- Example data in 4_refinement

Scenario Evaluation

- Call attributeStates.py on stopinfos.xml



```
Python Tool
Console output

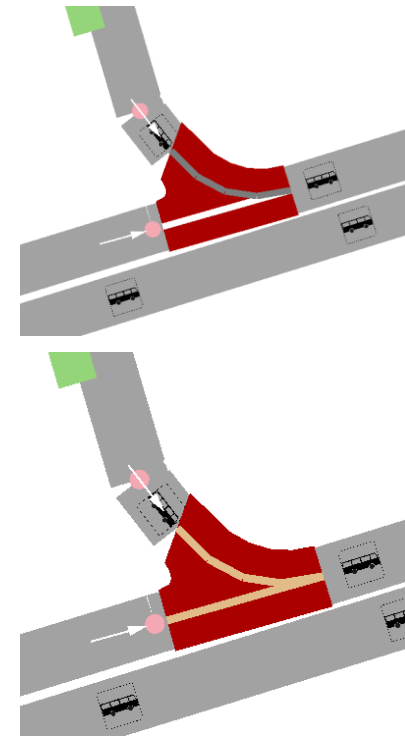
python "D:\erdm_ja\sumo-dev/tools/output/attributeStats.py" "D:\erdm_ja\sip-svn-trunk\papers\DLR\SUMO_Conference2026\tutorial\4_refinement.final\stopinfos.xml"
starting process...
stopinfo blockedDuration: count 1480, min 0.00 (pod.68), max 12.00 (pod.66), mean 0.01, Q1 0.00, median 0.00, Q3 0.00, stdDev 0.41
stopinfo busStop: count 421, min 153314536.00 (pod.66), max 2505524893.00 (pod.55), mean 918760661.67, Q1 153314536.00, median 153314536.00, Q3 2505524893.00, stdDev 1102081001.21
stopinfo ended: count 1480, min 31.00 (pod.68), max 4118.00 (pod.16), mean 1947.30, Q1 1022.00, median 1964.00, Q3 2853.00, stdDev 1068.73
stopinfo initialContainers: count 1480, min 0.00 (pod.68), max 0.00 (pod.68), mean 0.00, Q1 0.00, median 0.00, Q3 0.00, stdDev 0.00
stopinfo initialPersons: count 1480, min 0.00 (pod.68), max 8.00 (pod.41), mean 2.43, Q1 0.00, median 1.00, Q3 5.00, stdDev 2.92
stopinfo loadedContainers: count 1480, min 0.00 (pod.68), max 0.00 (pod.68), mean 0.00, Q1 0.00, median 0.00, Q3 0.00, stdDev 0.00
stopinfo loadedPersons: count 1480, min 0.00 (pod.31), max 8.00 (pod.53), mean 2.43, Q1 0.00, median 1.00, Q3 5.00, stdDev 2.92
stopinfo parking: count 1480, min 0.00 (pod.68), max 0.00 (pod.68), mean 0.00, Q1 0.00, median 0.00, Q3 0.00, stdDev 0.00
stopinfo pos: count 1480, min 10.65 (pod.36), max 34.59 (pod.13), mean 26.68, Q1 24.03, median 25.32, Q3 28.87, stdDev 4.91
stopinfo started: count 1480, min 30.00 (pod.68), max 4116.00 (pod.16), mean 1944.67, Q1 1020.00, median 1961.00, Q3 2850.00, stdDev 1068.68
stopinfo unloadedContainers: count 1480, min 0.00 (pod.68), max 0.00 (pod.68), mean 0.00, Q1 0.00, median 0.00, Q3 0.00, stdDev 0.00
stopinfo unloadedPersons: count 1480, min 0.00 (pod.68), max 8.00 (pod.41), mean 2.43, Q1 0.00, median 1.00, Q3 5.00, stdDev 2.92
process finished
```

- Example data in 4_refinement

Scenario Refinement



- Smart conflict management: junction type "zipper"
 - Netedit Selection mode (S)
 - Select all junctions (any attribute, empty value, ENTER)
 - Or Shift+J and button "Select All"
 - Inspect mode (I)
 - Inspect any junction, set type to "zipper"
 - Save network as net3.net.xml
 - Set option output-suffix 2 (Shift-F10)
 - All output files will have '2' appended before the .xml extension
 - Save sumocfg-as 2.sumocfg



```
Ride Statistics (avg of 3600 rides):  
WaitingTime: 66.58  
RouteLength: 2540.36  
Duration: 198.53
```

```
Ride Statistics (avg of 3600 rides):  
WaitingTime: 63.15  
RouteLength: 2539.40  
Duration: 189.28
```

- Example data in 4_refinement

State saving and loading



- Motivation
 - Run multiple what-if scenarios from a specific time T without repeating the slow warm-up to reach T
 - Diverging randomness after loading state (default): have scenarios evolve differently from the exact same starting point onwards (using different seeds)
 - Exact replication of behavior (**--save-state.rng**) when testing models that do not affect vehicle behavior (i.e. emission models)
 - Monitoring a long-running simulation by periodically saving states (**--save-state.period**)
 - Caveat: Simulation state is not guaranteed to be compatible across releases (in contrast to all other input files where we aim to guarantee compatibility)
- Example data in 5_state

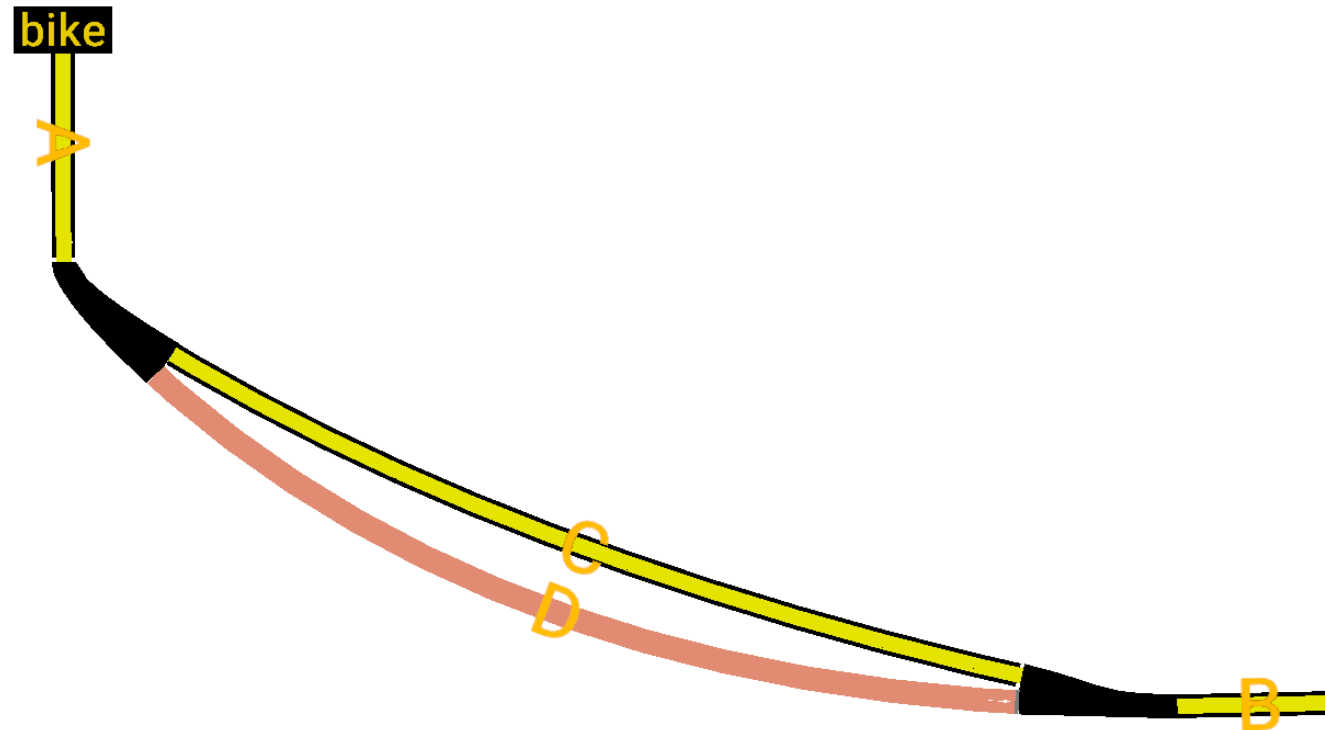
State saving and loading



- Load 2.sumocfg and edit (F10)
 - save-state.times = 1000 (can be a list)
 - save-state.files = state.xml (optional, defaults to state_TIME.xml.gz)
 - save-state.transportables
- Compare tripinfos, personinfos
- Result: state does not replicate exact prior behavior
 - save-state.rng
 - save-state.precision 16 (low values may show divergence)
- fcd-output: matches
- vehroute-output: matches
- tripinfo-output: mostly matches
 - remaining minor variations in routeLength
 - access stages that were finished before state-saving time are missing in tripinfo output of state-loading simulation.
- Example data in 5_state

Routing Preferences

- Vehicles in SUMO prefer the fastest route (by default)
- Realistically, a bicycle rider would prefer a bike path to a road even if it takes slightly longer

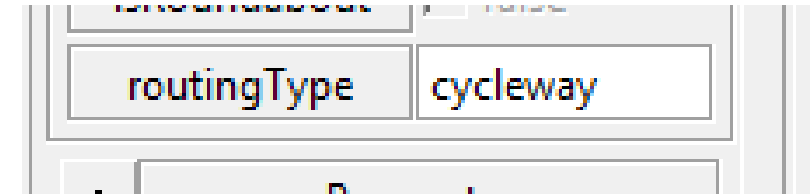


- Example data in 6_preferences

Routing Preferences



- New edge attribute **routingType** (arbitrary string), defaults to its edge type



- New feature: routing preferences (not yet in netedit 🙄)
 - Assumed travel time is divided by the configured priority for matching vType or vClass on edges according to their routingType
 - Loaded from additional file

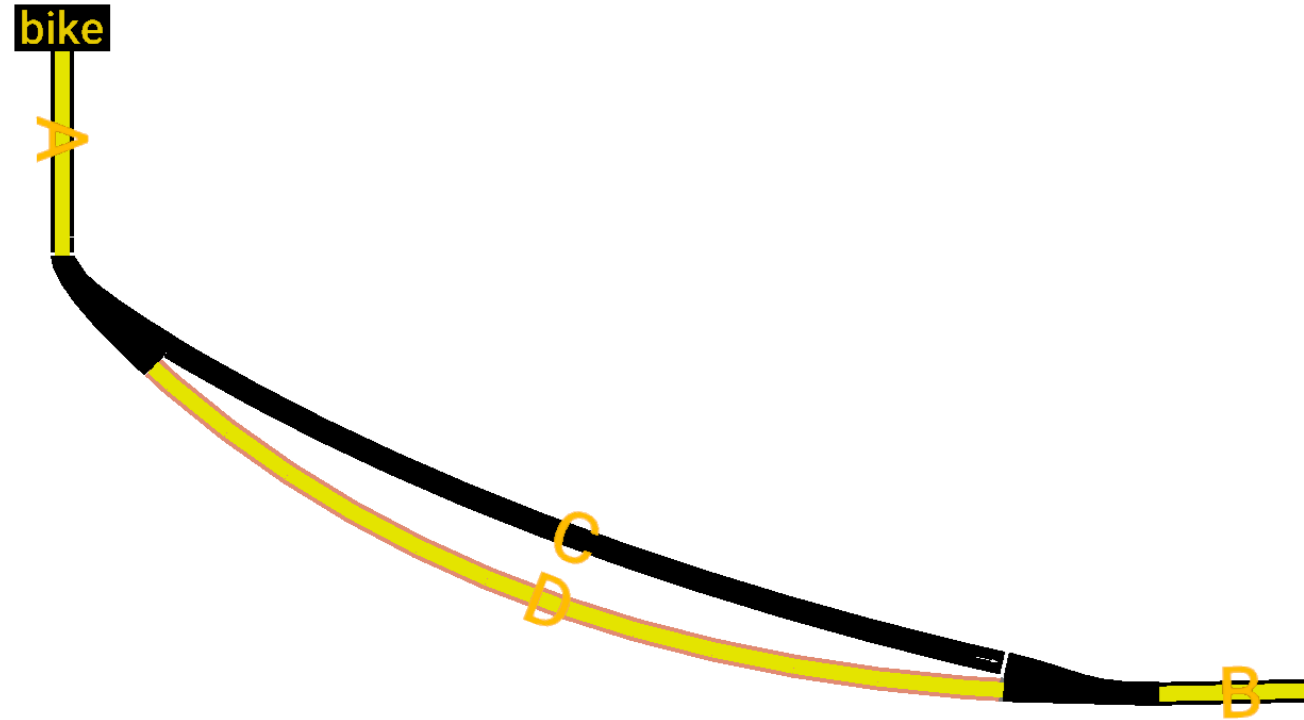
```
<preference routingType="cycleway" priority="2" vClasses="bicycle"/>
<preference routingType="bad_road" priority="0.5" vTypes="type1 type2"/>
```

- Example data in 6_preferences

Routing Preferences



```
<additional>  
  <preference routingType="cycleway" priority="2" vClasses="bicycle"/>  
</additional>
```



- Example data in 6_preferences

Conclusion

- Use [tools/osmWebWizard.py](#) to get a quick start
- Use netedit to discover all the extra tools
- Use LLMs to get an overview of capabilities and components
 - But don't trust every word they generate
- Use <https://sumo.dlr.de/extractTest.php> to get input examples
- 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

