

New Features in SUMO

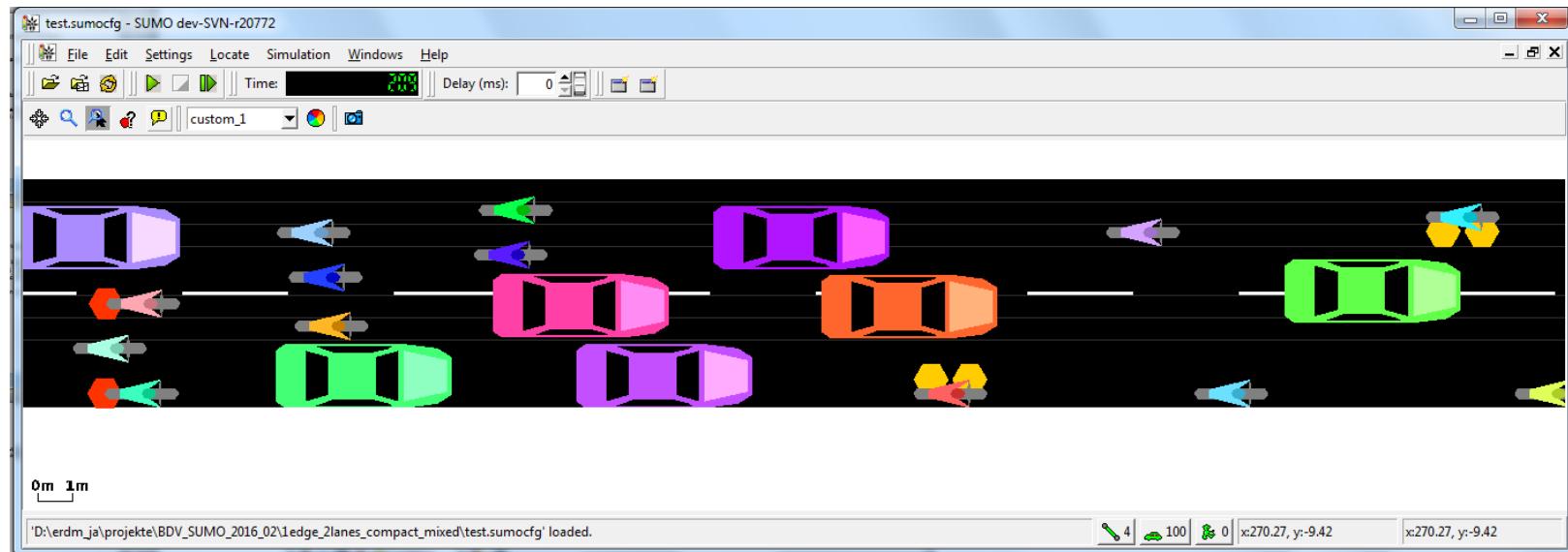
SUMO User Conference 2016



Knowledge for Tomorrow

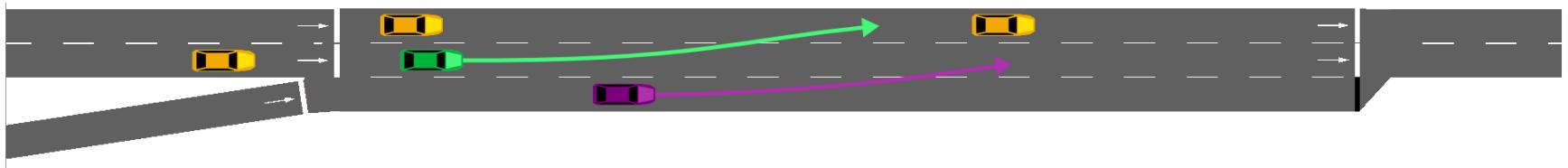
Sublane Model

- Configurable lateral resolution for car-following and lane-changing
- Continuous positioning in x,y (pos, posLat)
- New lane-changing model to accommodate lateral dynamics
 - maxSpeedLat, minGapLat, latAlignment, lateral encroachment (lcPushy)
- Allow modelling of Asian traffic characteristics (flexible lane use, large proportion of two-wheelers)
- Improved modelling of car/bicycle interactions (overtaking on a single lane)



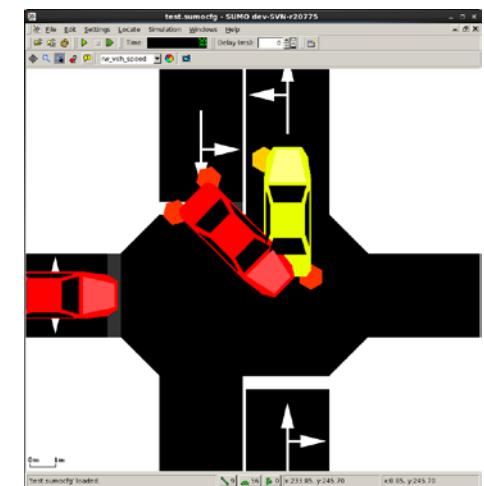
New Parameters for all lane-changing models

- One vType parameter for each changing reason
 - IcStrategic
 - IcCooperative
 - IcSpeedGain
 - IcKeepRight
- Control the likelihood (or eagerness) to perform lane changing for the respective reason
 - Public busses should be less likely to perform cooperative lane-changing that might put them at a disadvantage. (And should instead expect cooperation from everyone else)



Collision detection and handling

- Collisions are part of SUMO
 - Originally, bugs in the collision-free model (not all of them fixed)
 - Dangerous traffic light configuration
 - Intentionally unsafe car-following parameterization
 - new model for driver errors planned
 - TraCI
- So far, only detected along contiguous lanes
- New option for detecting collision on junctions
 - Detect invalid positioning of internal junctions
- New option for configuring collision handling
 - Teleport rear vehicle (current default)
 - Remove both vehicles
 - Warning only
 - Further extensions planned (i.e. vehicles block the road for some time before removal)



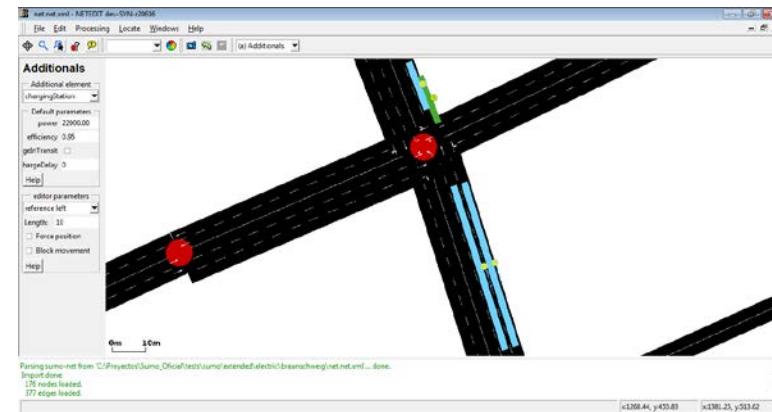
TraCI

- New vehicle command *nextTLS* to retrieve upcoming traffic lights
 - Returns variable length list [(tlsID, tlsLinkIndex, distance, linkState),]
- Improved coverage of the C++ client library
 - Vehicle add, remove, moveToXY
 - Variable subscriptions
 - Context subscriptions
 - ~90% coverage now (lots of additions already in 0.26.0)



Netedit support for additional network infrastructure *(still in branch)*

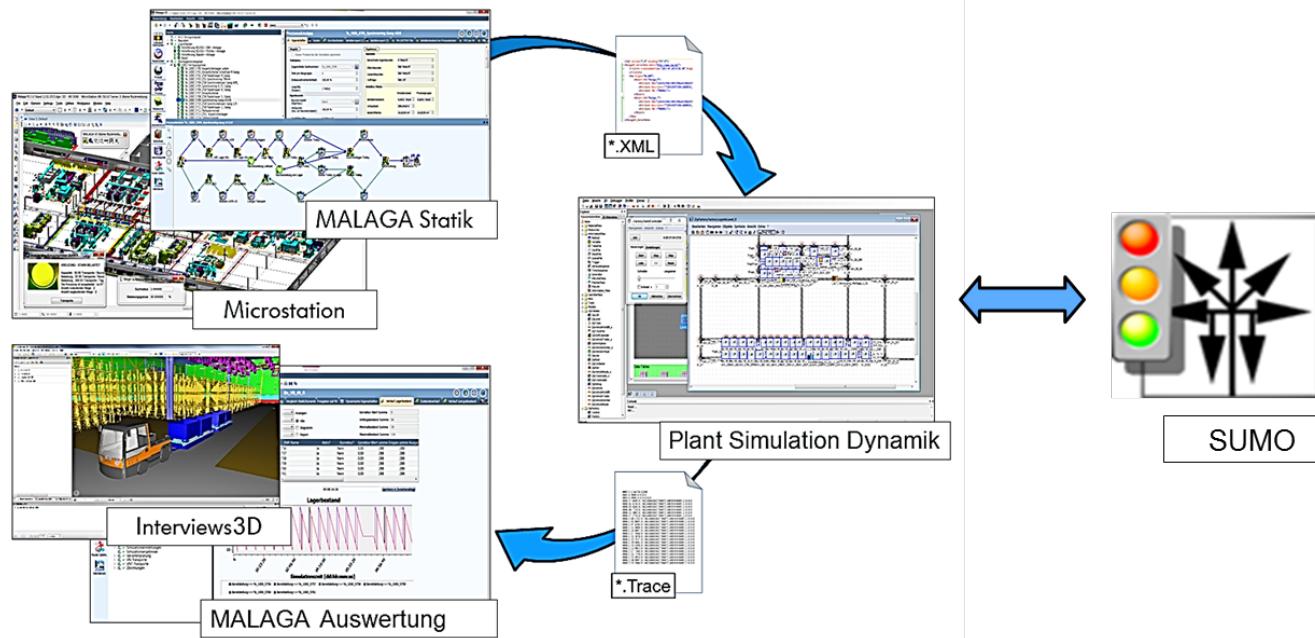
- Load, define, configure and save the following objects
 - Detectors (E1, E2, E3)
 - Rerouters
 - Stopping places (busStop, containerStop, chargingStation)
 - Calibrators
 - Variable Speed signs
 - RouteProbe detectors



Passing Blockage with Lane Changes

SUMO in Production Logistics

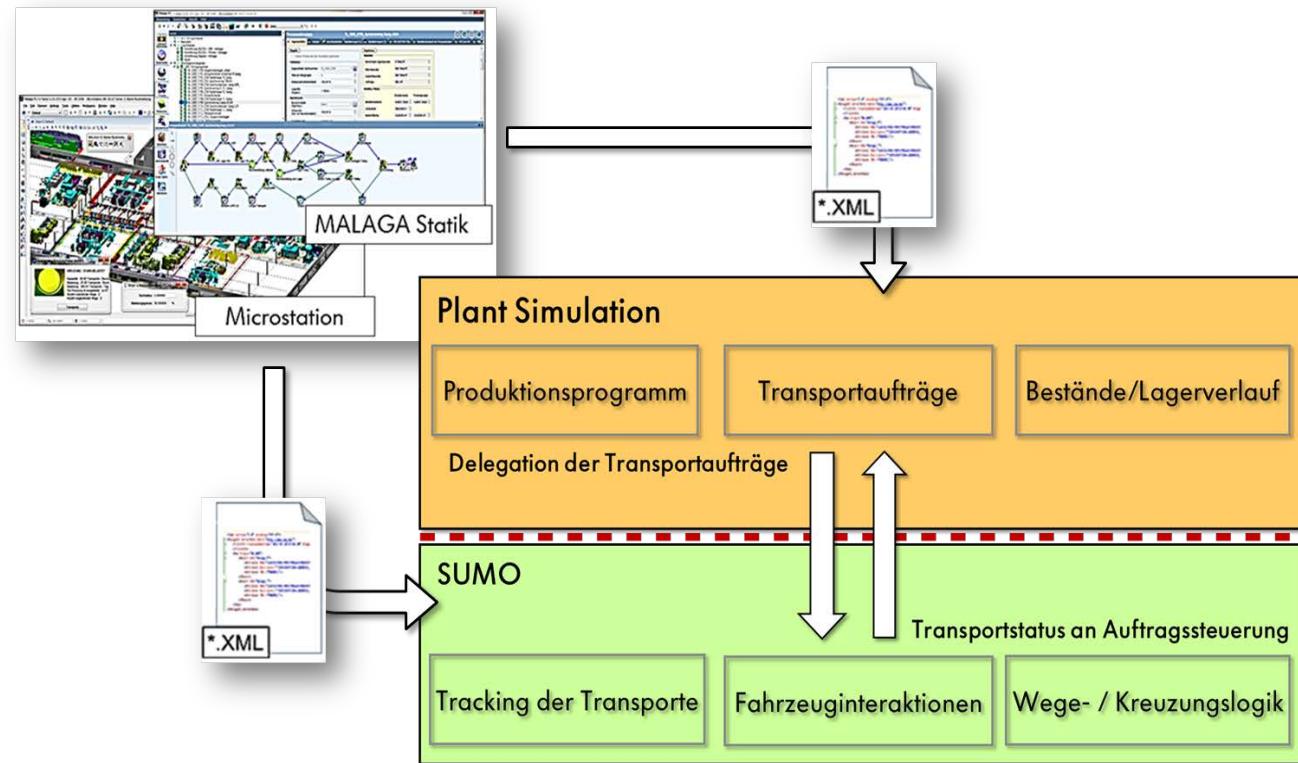
- Respecting „real“ dynamics in virtual inhouse logistic
- Coupling SUMO to existing material flow simulation
- Respecting oncoming traffic and lane changes



Passing Blockage with Lane Changes

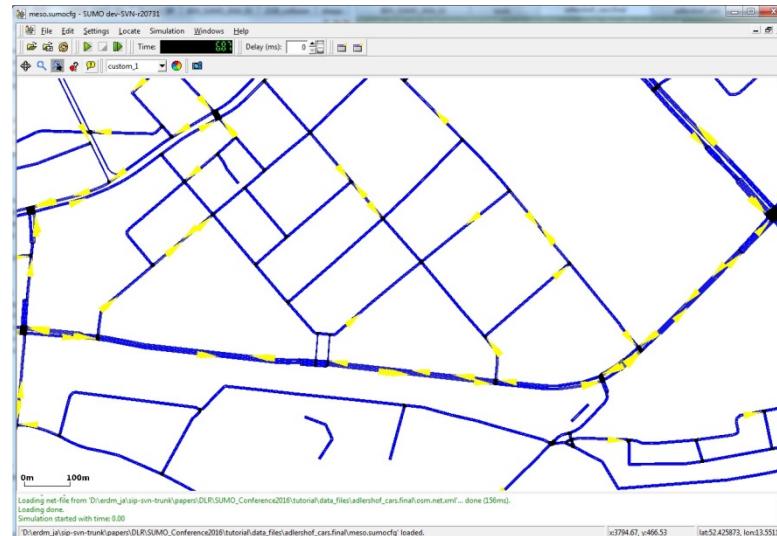
SUMO in Production Logistics

- Using existing coupling of Plant Simulation and Malaga



MESO

- Uses the same inputs as SUMO
- Running time of microsim ~15s (avg vehicle TimeLoss ~95s)
- Run scenario again with option --mesosim ~1s
 - MESO is fast!
 - TimeLoss < 1? Add option --meso-junction-control
 - -> TimeLoss 50.0.
 - MESO does not model vehicle acceleration, impact on urban dynamics



Intermodal Routing

- Intermodal Trip chains
- Input
 - Network with bus stops
 - Transfer times
 - Timetables
 - Persons and their daily plans
 - Availability of modes
- Output
 - Fastest intermodal route
 - Respecting transfer times
 - To be run directly in the Simulation

```
<flow id="bus" from = "beg" to ="end"  
line="bus" begin="0" end="1000"  
period="300">  
    <stop busStop="beg_0" until="10"/>  
    <stop busStop="left_0" until="20"/>  
    <stop busStop="end_0" until="30"/>  
</flow>  
<person id="p0" depart="0">  
    <personTrip from="beg" to="end"  
        modes="public"/>  
</person>  
  
<person id="p0" depart="0.00">  
    <walk edges="beg" busStop="beg_0"/>  
    <ride busStop="end_0" lines="bus"/>  
    <walk edges="end"/>  
</person>
```



Intermodal Routing Outlook

- Bicycle traffic
 - Taking it with you in car and public transport
 - Transfer at defined stations
- Integration into the running simulation
 - Current travel times
 - Intermodal rerouting
- Import
 - VISUM
 - OSM
 - GTFS

