Simulating Digital Rail

From PlanPro railway plannings to SUMO simulations

Arne Boockmeyer

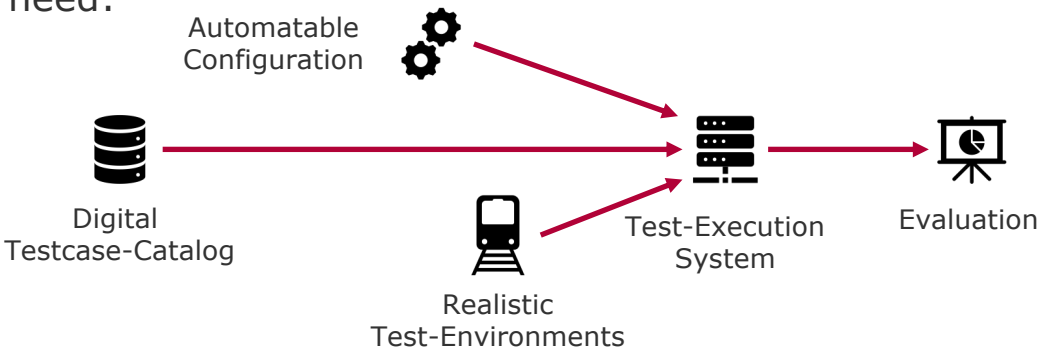
Professorship for Operating Systems and Middleware

Hasso Plattner Institute, University of Potsdam

Contact: arne.boockmeyer@hpi.de

Motivation

- The railway domain is getting more and more digital, with new digital devices, processes and open standards
- But introducing new devices is a complex process to guarantee interoperability and conformity with existing systems
- This so far contains several manual steps, this does not scale
- Our Idea: Use digital plannings of railway networks to operate/simulate the railway network, containing the new device and several existing ones
- Therefore we need:

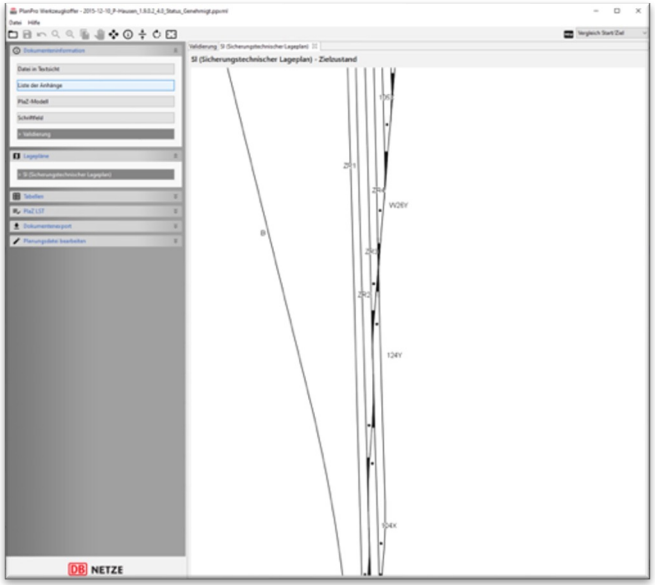


PlanPro

- A data format, that contains all details about a planned railway network
- Core-part of a full digital planning process – away from paper-based towards digital processes
- XML-based: (but in German)

```
<Container>
      <Fstr_Fahrweg></Fstr_Fahrweg>
      <GEO_Knoten></GEO_Knoten>
      <Signal></Signal>
      <TOP_Kante></TOP_Kante>
      <TOP_Knoten></TOP_Knoten>
</Container>
```

- Verbose format – already small examples creates massive file sizes

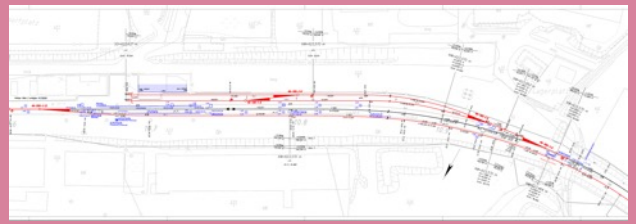


(Picture shows the PlanPro Werkzeugkoffer (toolbox), a software to display and analyse PlanPro files)

PlanPro Network Creation

ProSig 7 / ProVi

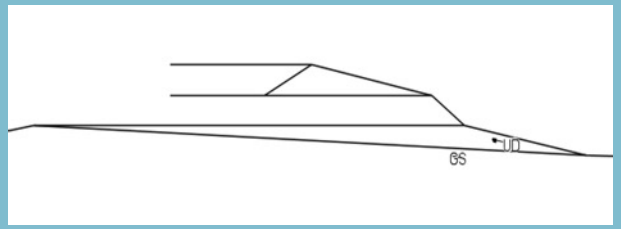
- Typical planning tools of DB can already export to PlanPro files
- Contains many details about infrastructure, geography and more
- Extensive to create, so only a few railway networks are exported to PlanPro through ProSig 7 / ProVi



- Closed-source, expensive

PlanPro Generator

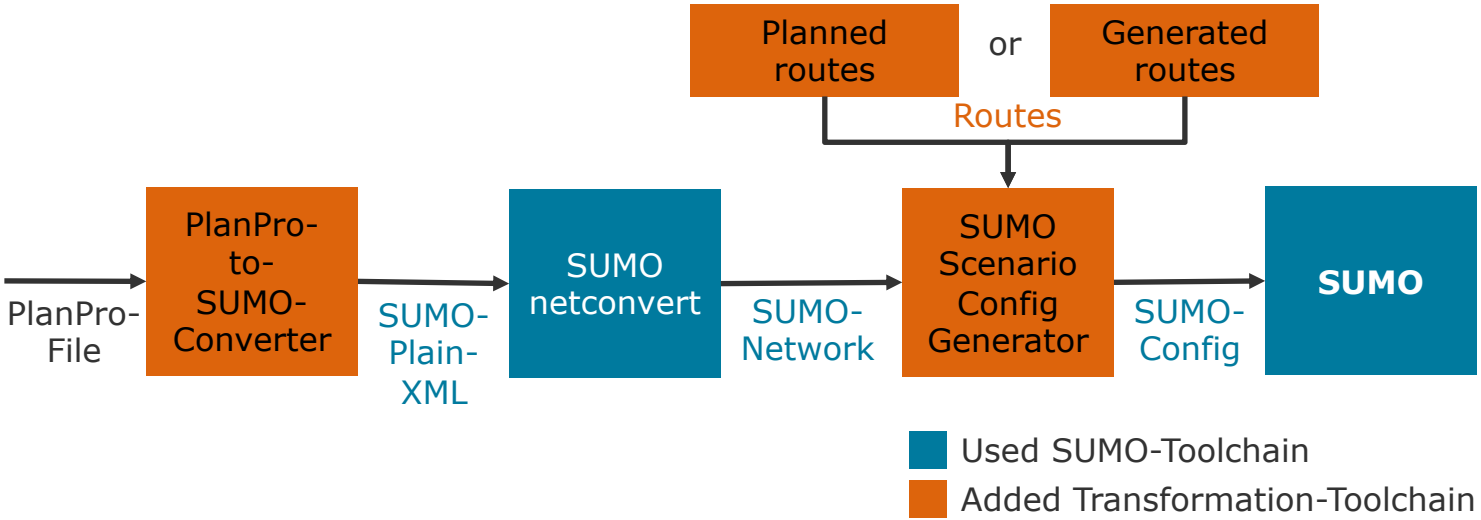
- Create small PlanPro examples
- Only contains the most relevant aspects of a planning
- Strong abstraction with large focus on test automation
- Allows us to create many different scenarios



- Available on GitLab: https://github.com/arneboockmeyer/planpro-generator

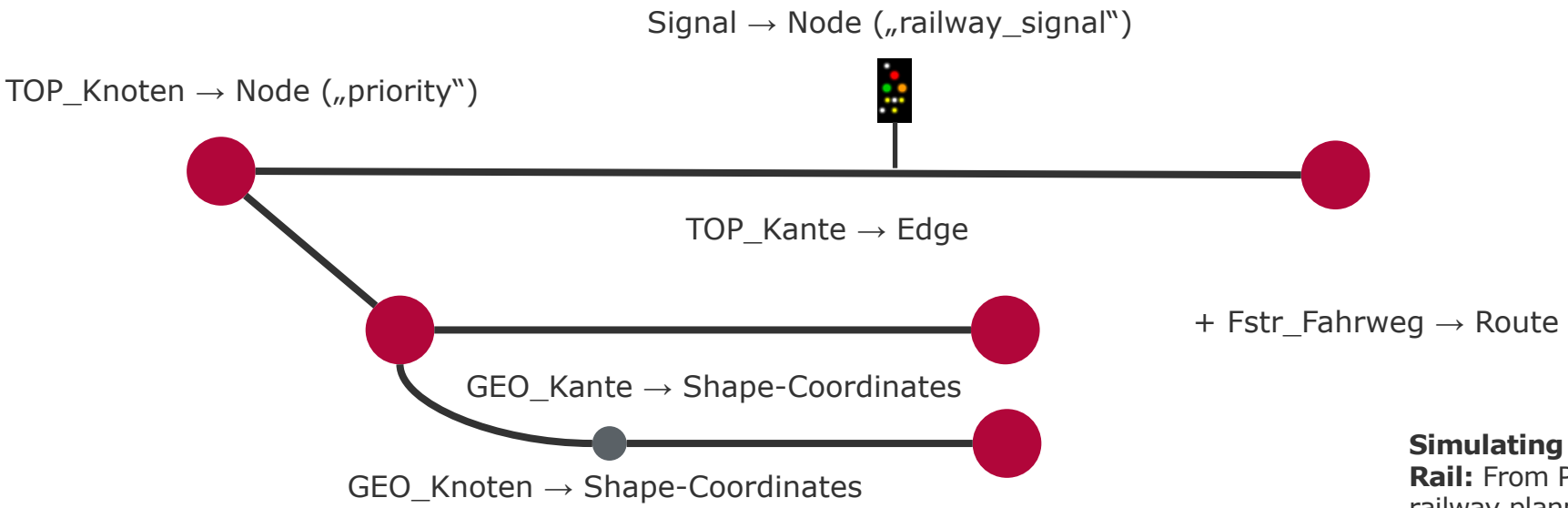
Transformation from PlanPro to SUMO

- To simulate the railway networks in SUMO, a transformation between the two file formats is necessary
- SUMO-Plain-XML is used as a step in between:



- Available auf GitHub: https://github.com/arneboockmeyer/planpro-sumo-converter

PlanPro to SUMO Dictionary

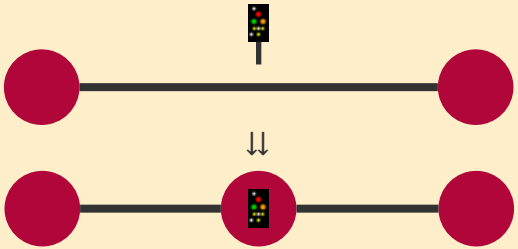


- Every TOP_Kante can cover multiple GEO_Kanten
- A GEO_Kante is a function defining the pathway of the rails (Straight, Arc, Clothoid, …) – right now all interpreted as straights

Challenges during Development

Signals are Nodes

- In PlanPro, Signals are annotations to TOP_Kanten
- SUMO uses nodes as signals



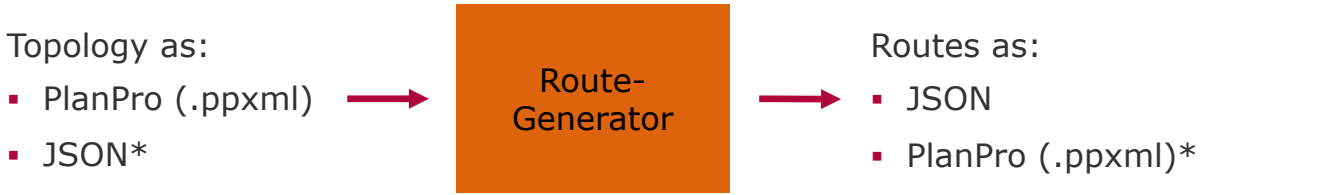
- Separating TOP_Kanten causes a lot of confusion during the processing of routes

Position of Signals

- The position of signals is defined by the distance from the start of the TOP_Kante
- SUMO needs exact coordinates
- Since every TOP_Kante can cover multiple GEO_Kanten, the related GEO_Kante needs to be determined
- The position of the signal than was estimated by the remaining distance

Generation of Routes

- Every route starts at a signal, ends at a signal and covers (multiple) edges
- To generate the routes, a Python-package was developed:

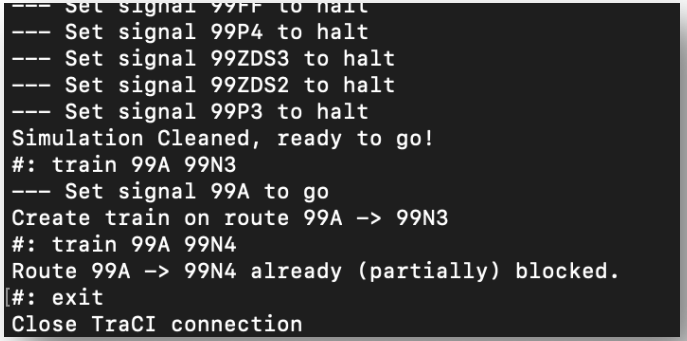


(*: planned)

- The algorithm is a DFS on the topology with a modified neighbor-function
- Available on GitHub (in the next days): https://github.com/arneboockmeyer/planpro-running-track-generator

Test Controller

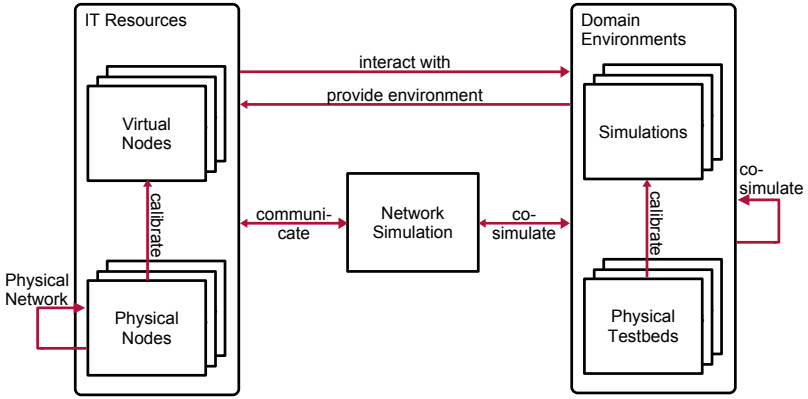
- The Test Controller is a console application that allows the test manager to:
  - Print details about the setup of the railway network (incl. routes)
  - Show conflicts between routes
  - Create trains on routes
  - Run schedules and collect evaluation data
- It contains a basic interlocking
  - Manage state of network
  - Orchestrating the signals
  - Detect conflicts
- … and run the SUMO-Simulation!
- Available on GitHub: https://github.com/arneboockmeyer/sumo-railway-test-controller



Demo

Marvis

- Marvis is a hybrid IoT-Testbed, containing:
  - Co-Simulation of SUMO and ns-3
  - Simulated Nodes and Hardware-in-the-loop
  - Fault injection capabilities



- Research-Paper:
  - Beilharz et. al., “Towards a Staging Environment for the Internet of Things.”, PerCom Workshops, 2021

- Available on GitHub: https://github.com/diselab/marvis

Next Steps

Take more details from the PlanPro-file to enrich the SUMO-network
(e.g. more types of signals, train detection systems, …)

Get geography more precise
(solve functions to get more points in between)

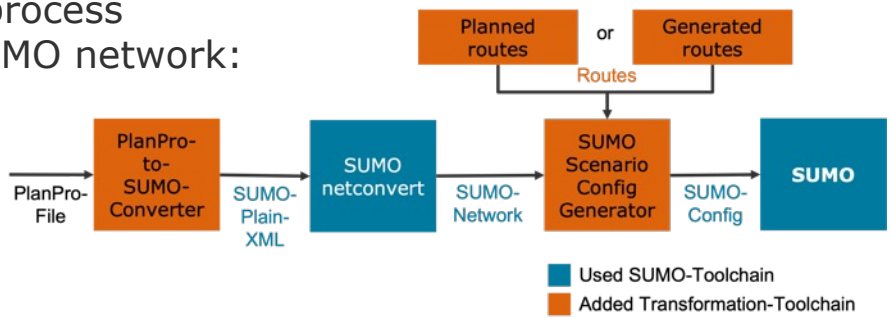
Connect to execution environments to have full test environment
(to Marvis or the EULYNX-Live lab)

(if anybody has further ideas, knowledge on some of these points, or anything else, please contact us under arne.boockmeyer@hpi.de)

Summary

- The PlanPro-Format should be the standard for future full-digital plannings
- It contains many details about infrastructure, geography, …

- The transformation process transforms it to a SUMO network:



- The test controller allows the test manager to operate trains in the SUMO network to achieve test automation

Image Sources

Signal: Markus4linger, https://commons.wikimedia.org/wiki/File:Ks-Mehrabschnittssignalschirm_(voll_ausgeleuchtet).png

Other images:

Manufactures-Logos from their organizations and company's

Icons by Microsoft Office

Graphics from our research group

Screenshots from PlanPro tools and plannings

Screenshots of SUMO (https://sumo.dlr.de)