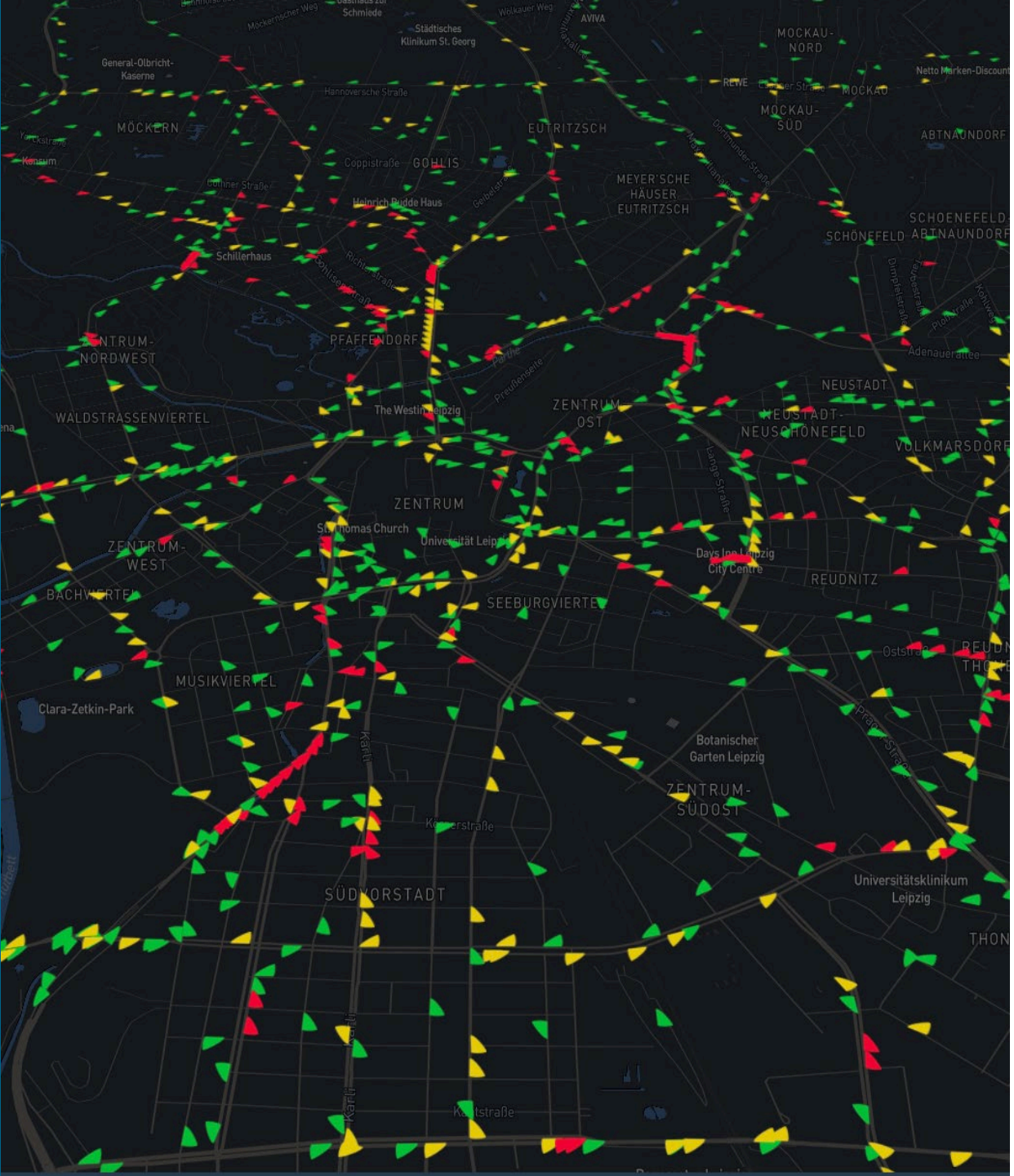
GMNS: General Modeling Network Specification and MATSim

William (Billy) Charlton <charlton@tu-berlin.de>
Technische Universität Berlin, VSP (Transport System Planning and Telematics)
German Aerospace Center (DLR)

VERKEHRSSYSTEMPLANUNG
VERKEHRSTELEMATIK
TU BERLIN

DLR



MATSim:

an open-source platform for transport microsimulation

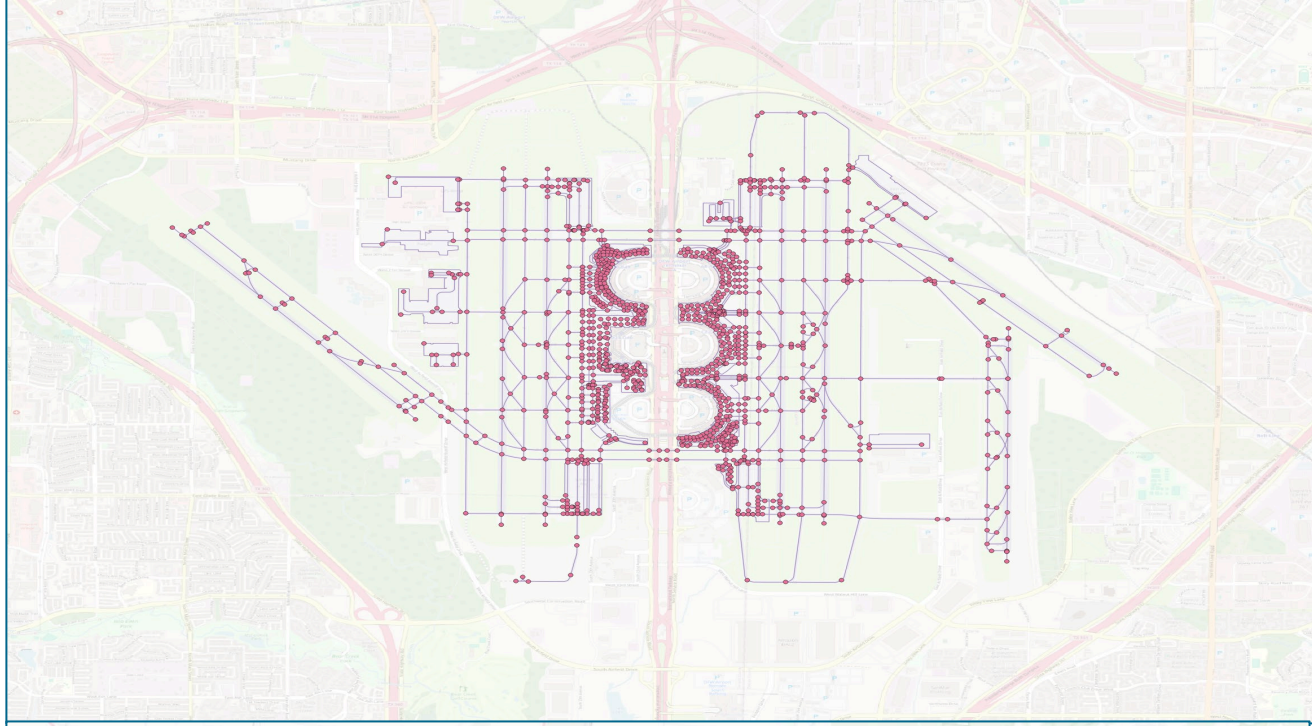
matsim.org

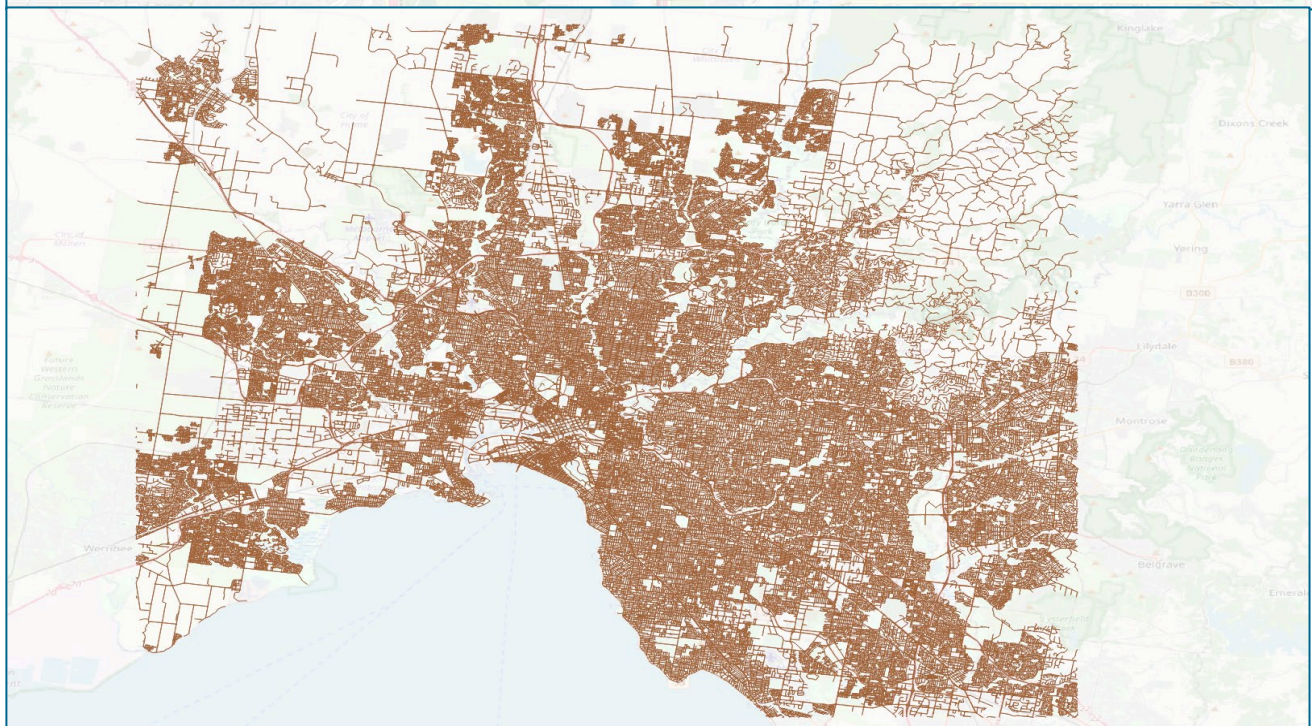
GMNS:

General Modeling Network Specification

sponsored by Zephyr!
first presented in 2022

github.com/
zephyr-data-specs/GMNS





GMNS:

General Modeling Network Specification

sponsored by Zephyr! first presented in 2022

github.com/zephyr-data-specs/GMNS

Network Management Tools

Community members and researchers have begun to develop an ecosystem of tools to facilitate use of GMNS, including:

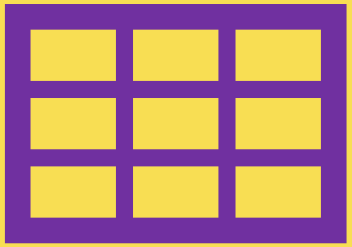
- NetworkWrangler - NetworkWrangler is a suite of tools for defining, creating and building network scenarios, starting from a base network and adding a series of project cards (network diffs) to create future networks and variants.
- OSM2GMNS – osm2gmns is an open-source Python package that enables users to conveniently obtain and manipulate any networks from OpenStreetMap (OSM). With a single line of Python code, users can obtain and model drivable, bikeable, walkable, railway, and aeroway networks for any region in the world, and output networks to csv files in GMNS format for seamless data sharing and research collaboration.
- Path4GMNS - Path4GMNS is an open-source, cross-platform, lightweight, and fast Python path engine for networks encoded in GMNS. Besides finding static shortest paths for simple analyses, its main functionality is to provide an efficient and flexible framework for column-based (path-based) modeling and applications in transportation (e.g., activity-based demand modeling).

GMNS *and* MATSim: let‘s bring them together!

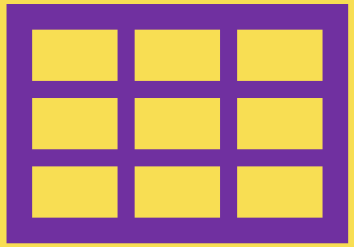
GMNS:

Flat CSV tables for everything

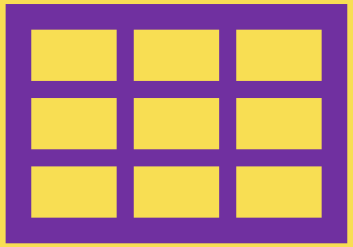
config.csv



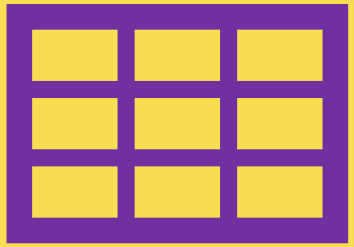
nodes.csv



links.csv



geometry.csv



MATSim:

Native format is XML-based

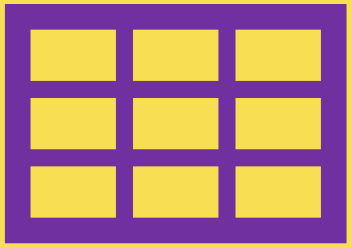
```
<node id=“1“ x=13.5 y=31 … />
<node id=“2“ x=13.6 y=31 … />

<link id=“a12“ dist=14.1 cap="600“ … >
    <attributes>
        <attribute bikelane=“true“ />
    </attributes>
</link>
<link id=“…“>
</link>
….
```

GMNS *and* MATSim: let‘s bring them together!

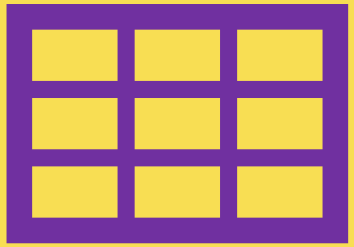
GMNS:

Flat CSV tables for everything.

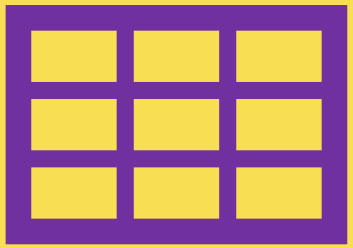
config.csv



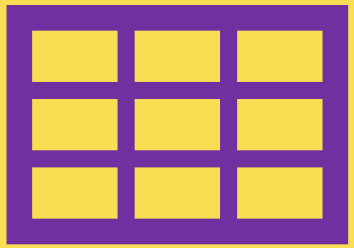
nodes.csv



links.csv



geometry.csv

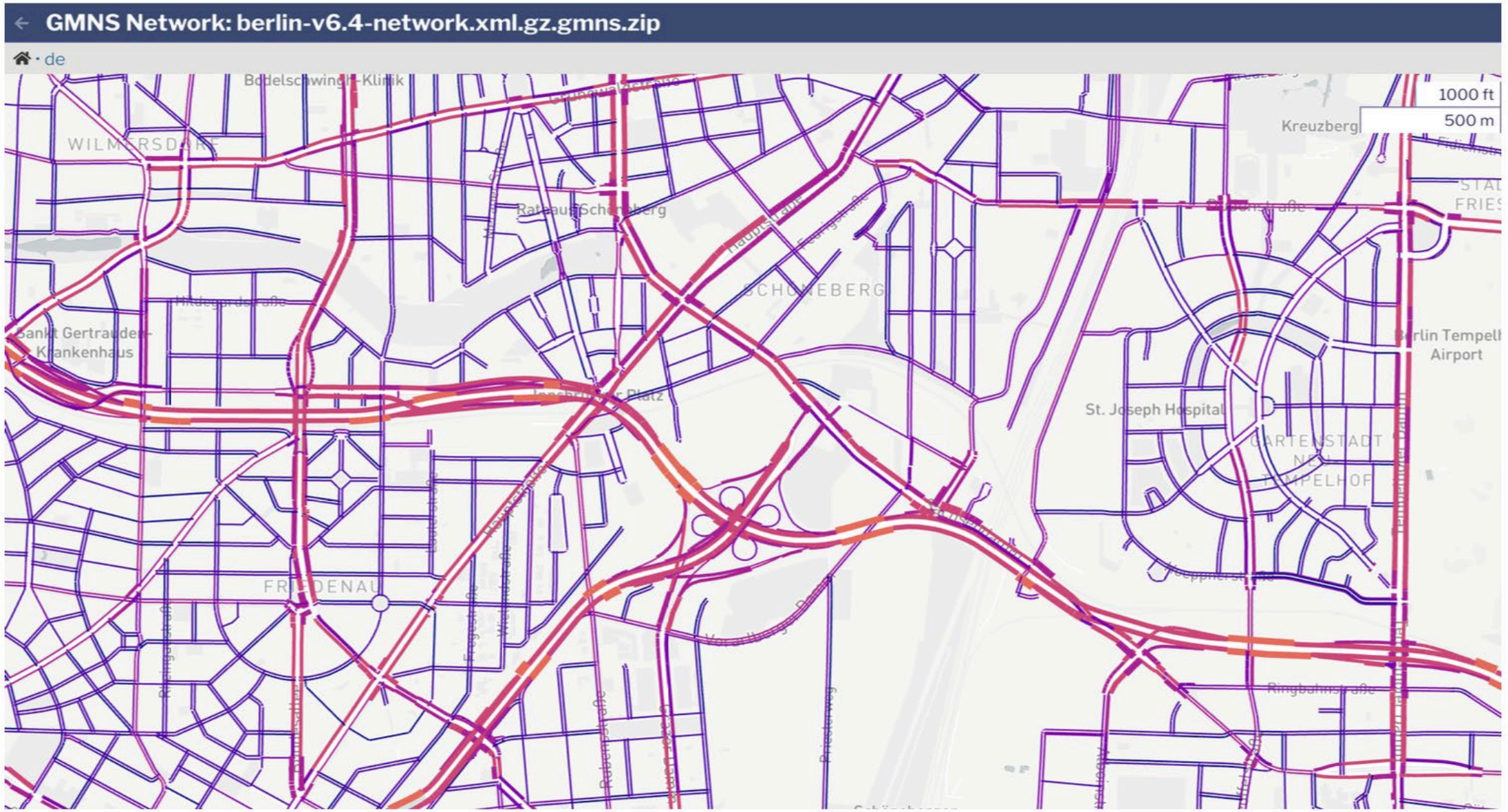


seriously for everything:

zone.csv
segment.csv
location.csv
lane.csv
segment_lane.csv
link_tod.csv
segment_tod.csv
movement.csv
movement_tod.csv
signal_controller.csv
signal_coordination.csv
signal_detector.csv
signal_phase_mvmt.csv
signal_timing_plan.csv
use_definition.csv
use_group.csv
curg_seg.csv

IT WORKS!

Berlin MATSim network converted to GMNS, loaded in SimWrapper



Let‘s talk about the ubiquitous, problematic CSV format

- **CSV is “lowest common denominator“, dead simple!**
  …but causes real problems here.

- **All ID fields** in GMNS are type “any“:
  Excel and CSV parsing libraries convert numeric strings to numbers. This isn‘t just a problem for ZIP codes like 02134, but for any ID fields.
  - *This immediately broke our first attempts at conversion*

- **CSV files are enormous and don‘t compress well**
  - *GMNS uses WKT „well-known text“ for curvy geometries, which don‘t compress.*
  - *Sample GMNS networks have 3000 links.*
    Some MATSim networks have… 1.2 million

CSV Griping, part II

- **You can‘t load a GMNS network: you have to load multiple files.**
  This is a problem for interactive use, e.g. SimWrapper

- **Parsing is error prone and extremely slow**
  02134, quoted fields.
  Binary formats load 20X faster!

- **ESRI Shapefiles had the same problem,**
  which is why GeoPackage .gpkg exists now

ADVOCATING FOR A CONTAINER

- GMNS documentation claims the spec is **“container agnostic“** meaning: bring your own container!

- I tried **.zip files** – memory problems loading in browser

- I tried **SQLite** – even bigger than CSV!
  And without setting up column indices, impossibly slow

- I tried **Avro** / **Arrow** binary formats – very promising!!
  …but now I need to explain these formats to anyone else who wants to use my “GMNS“ files…

- Did I try YOUR favorite container format?
  …Seriously, how many are there?

ADVOCATING FOR A CONTAINER, II

Remember OMX, the Open Matrix Specification?

- Broad adoption now for modelers, default for ActivitySim
- 100% supported by commercial vendors
- Binary format with multiple tables, compressed, shareable!
- Created by us, the modeling community! Just like GMNS!

What made OMX successful?
How can GMNS replicate that success?

FOLLOWING in OMX‘ FOOTSTEPS

OMX was

- Complete
- Technically sound
- Practical in daily use

Don‘t stop at defining what goes into GMNS, without

- Defining the standard container

- Thinking about practical use cases, i.e. users
  - *Interactive use*
  - *Memory constraints*
  - *Shareability*

GMNS: General Modeling Network Specification and MATSim

William (Billy) Charlton <charlton@tu-berlin.de>
Technische Universität Berlin, VSP (Transport System Planning and Telematics)
German Aerospace Center (DLR)

VERKEHRSSYSTEMPLANUNG
VERKEHRSTELEMATIK
TU BERLIN

DLR