Navigating the Open

Our Bosch eBike Flow Journey with GraphHopper & OSM

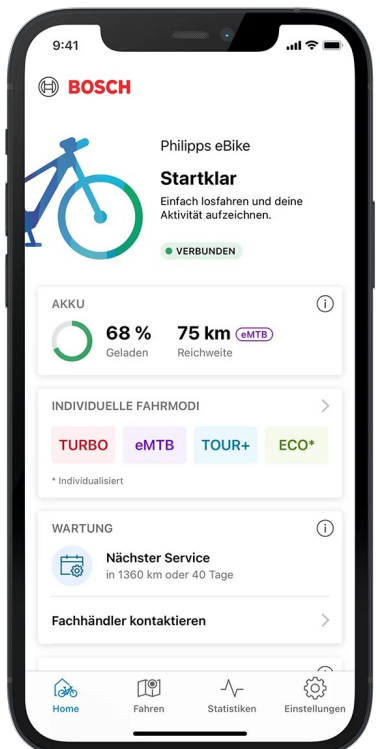
Presenter: Dr. Olaf Flebbe

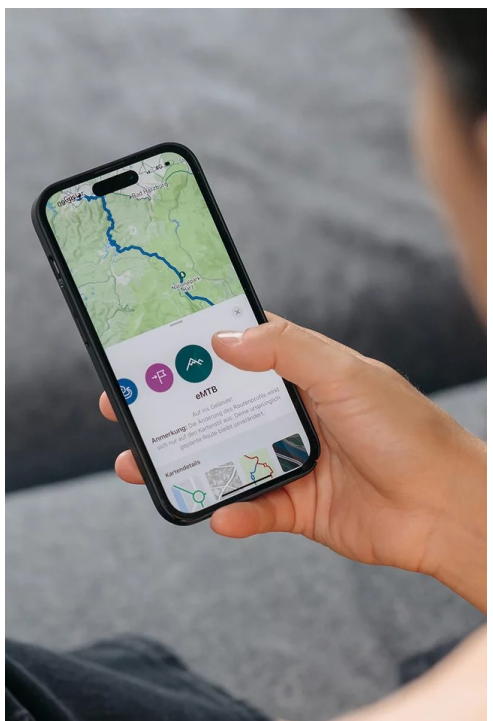
BOSCH

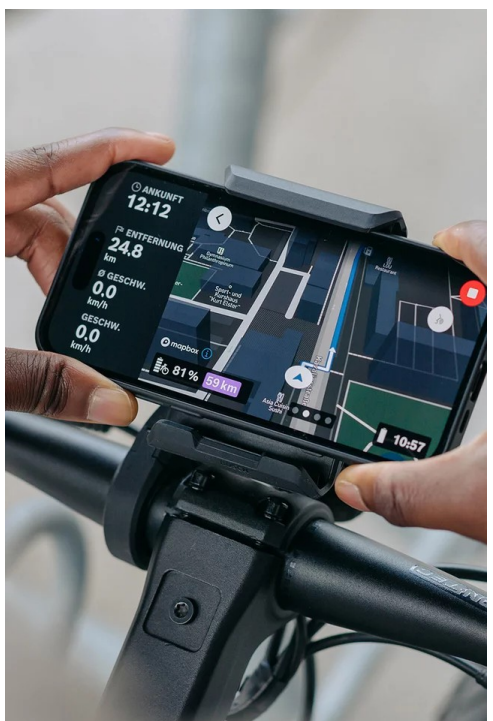
About me

- PhD in theoretical astrophysics (computational physics)
- OpenSourceSoftware since 1993 (Minix: gcc, Linux: libm, Psion: perl, python)
- 30 years in Software Consulting C/C++/Java/Python:
  Portable SW, 3D High End Viz, Performance, Security, HW Driven
  (Daimler, Bosch, BMW, Porsche, Bundestag, BSI)
- Asam ODS Working Group
- Member of the Apache Software Foundation and
  Member of PMC Apache Bigtop (Hadoop Distribution)
- Since 2017: eBike. Leader Developer Navigation Team

Flow App: Navigation for eBikes







OSS Legacy

- Mapbox : now closed source, still a major source of important OSS tech
  - Maps: MVT (Vector Tile Format), Mapbox-gl.js Map Rendering, ear cutting algo etc.
  - Directions API (once OSRM), SDK
  - Repositories abandoned or taken over by community
- Mapzen Elevation Data
  - Most accurate and complete free Digital Elevation Model compiled from free sources
  - Data set we are using is about 500 GB with a ~ 100x100m DEM raster
  - Data sources taken over by AWS

BACKEND MAPBOX

SMARTPHONE (Android, iOS)

BACKEND BOSCH

Maps

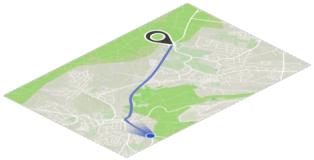
mapbox

OSM

Navigation

mapbox

Routing



BOSCH

graphhopper

Third-party connections

/

BOSCH

Search

mapbox

BOSCH

Nominatim

Kiox Navigation

BOSCH

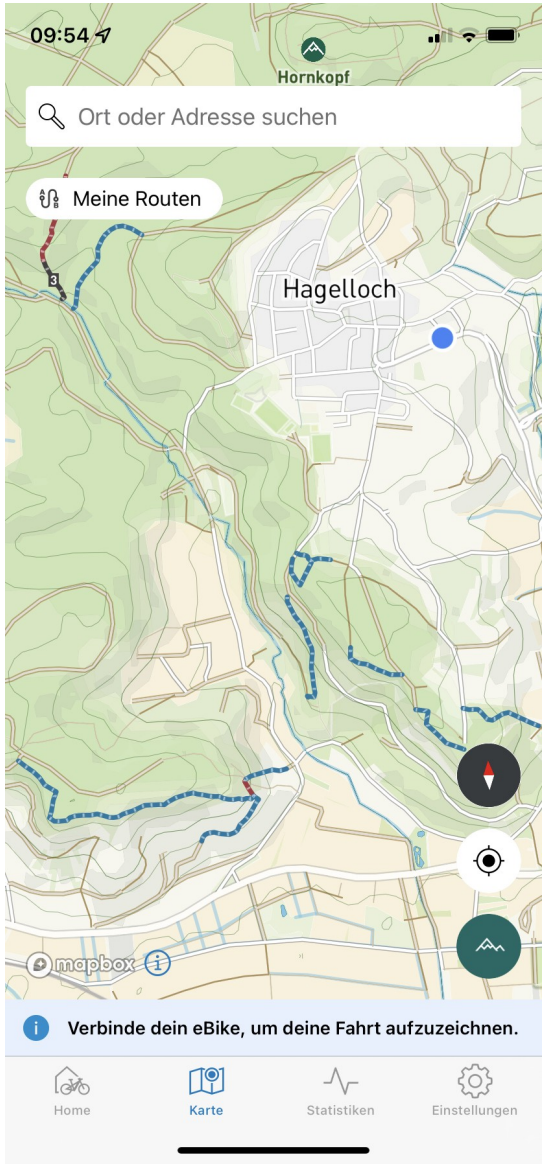
Range

BOSCH

BOSCH

Mapping

- Using the solution from Mapbox almost OOB
- We added from OSM to the map tiles (for instance)
  - Bike Network information (for instance BW bike network)
  - MTB Trails



Routing Data Sources

- Open Streetmap
  - „wikipedia for maps“
  - Openstreetmap License
  - Special Data Format: Heavily compressed and optimized data format „pbf“ using protobuf
  - Planet file is > 60 GB
  - Updates bi weekly (Feeds with in time updates available)

- Mapzen Elevation Data
  - Most accurate and complete free Digital Elevation Model compiled from free sources
  - Data set we are using is about 500 GB with a ~ 100x100m DEM raster
  - Attribution needed

- Both of them are available as S3/Objectstore datasources in the cloud.

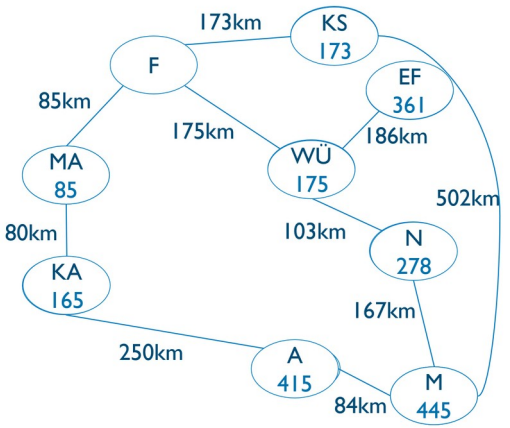
Routing
From data to route...

Data sources

- OpenStreetMap (OSM): Link data
  - Length
  - Road type / speed limits
  - Surface / smoothness
  - Bike lane / network / MTB scal
  - Access restrictions
  - Derived values (e.g., “in a forest”)
- Mapzen Open Digital elevation model (DEM)
  - Slope
- Usage data
  - Popularity
  - Rider speeds

Routing

Find the “cheapest” path to your destination



Routing profile

“Link cost” as trade-off:

- Duration
- Priority (or “attractiveness”)
  - Rider-dependent
  - Different “profiles” : eMTB, Leisure, Daily



Routing with Graphhopper

Optimal Route on a "graph“ Or Map-Matching

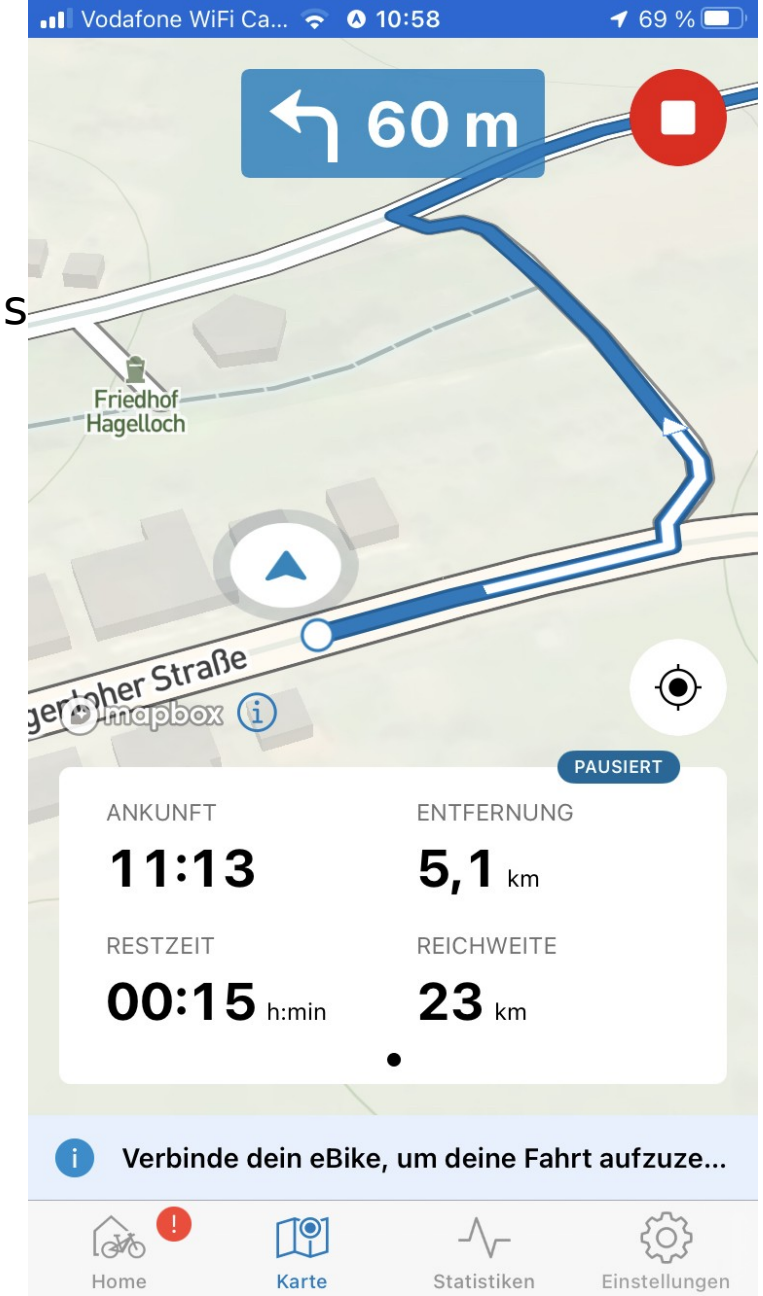
OSS with Bosch specific Overlays:

- routing Profiles (weighting) (Leisure, eMTB, Daily, GPX)
- Use information about water in surrounding, bike network, use OSM Tags
- Conversion to Mapbox Directions API Format (for digesting in Mapbox SDK)

AWS-Batch: Packing/Store of graphs/jobs to automatically generate maps for a specific graphhopper release

Routing

Using Mapbox Navigation API by replacing calls to Directions with Graphhopper Output



Maintaining a Graphhopper Fork

Graphhopper: upstream from github.com

Graphhopper-fork: Upstream releases pulled in with minimal changes (extension points), not accepted upstream

Overlay Navigation-graphhopper:

- Our business logic:
  - Specific weights for the routing graph (boosting BW bike network for instance)
  - Tensorflow Binding for evaluating speeds on links: ETA
  - Legal Stuff: S-pedelec routing, which link are you allowed to use in specific country
  - Smoothing elevation graph to be consistent with activity tracking

Very painful major releases of Graphhopper 5.x → 11.0

Contributing to Graphhopper

We contribute back fixes to the overall graphhopper:

- Fixes to the directions (aka OSRM) response format for the benefit of mapbox/maplibre SDK
- Supplying missing translations for voice instructions: swedish, polish, turkey

We planned to supply enhancements for faster map-matching:

- Rejected :) One reason the submitter used the “M*” word

Make some extension points public for us, shortly discussed and rejected

Contributing to Graphhopper

Graphhopper does only accept stuff fitting their vision, and we accept this.

This is quite different to my experience with OSS Projects (w/o corporate backing)

But it makes our live quite strange:

- We have to fight for contributing anything in the Organization
- We should not contribute Bosch specific Content
- Only contributions for the benefit of the whole OSS eco system are welcome

Makes it rather narrow

Other Contributions

OSMPBF go reader: github.com/paulmach/osm (MIT License)

Change of protobuf library, filtering of tags etc.

Used for displaying MTB routes and eBike Charging stations on the map.

Like a Bosch

- It‘s Teamwork!
  - OSM
  - Graphhopper
  - Mapbox OSS
  - Nominatim
  - Bosch