



Universität Augsburg
Fakultät für Angewandte
Informatik

Entwurf einer Hindernisserkennung und -vermeidung für die Mikromobilitätsplattform Scoomatic

Bachelor-Abschlusspräsentation

Henri Chilla

Augsburg | 16.03.2020

Agenda

- 1** Einleitung
- 2** Projektziel
- 3** Sensoren
- 4** SLAM
- 5** Lokalisierung und Navigation
- 6** Systemaufbau
- 7** Testing
- 8** Fazit
- 9** Ausblick

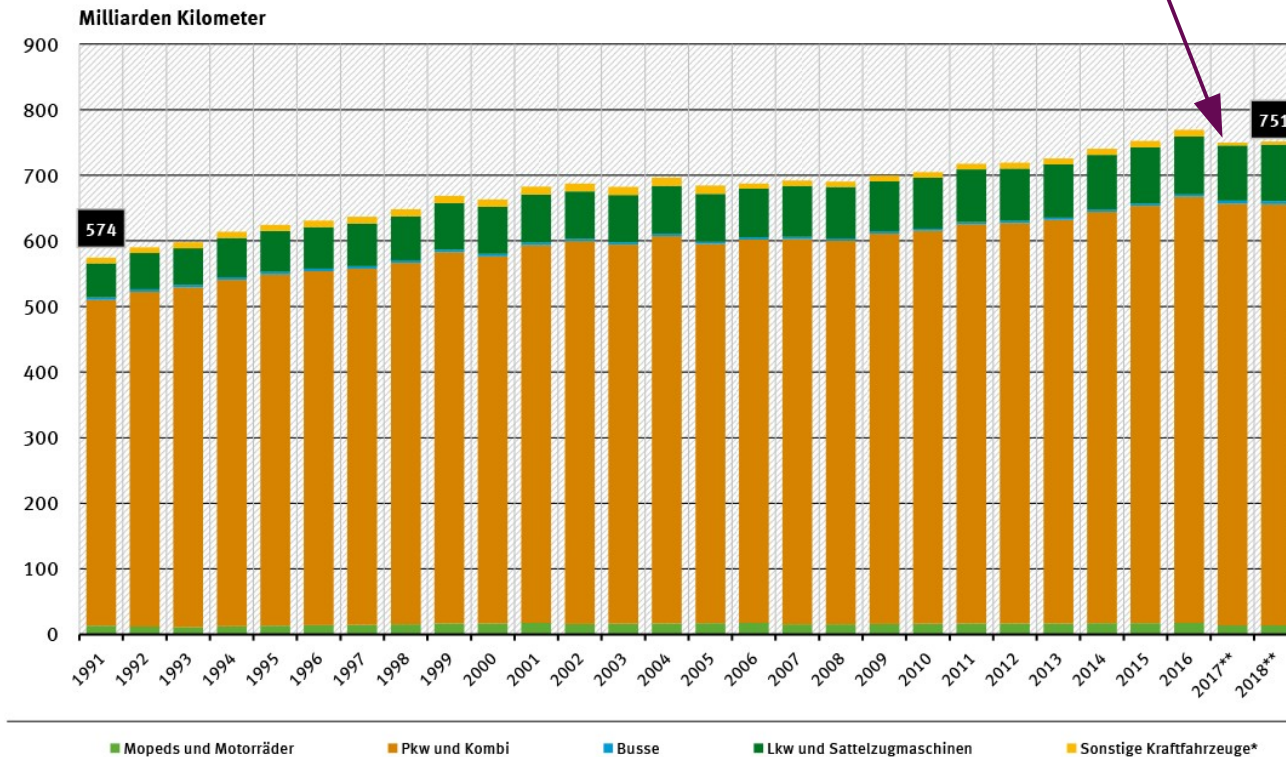
Einleitung

Motivation

Linkes Diagramm: <https://www.umweltbundesamt.de/daten/verkehr/fahrleistungen-verkehrsaufwand-modal-split#fahrleistung-im-personen-und-guterverkehr>

Rechtes Diagramm: <https://www.zukunft-mobilitaet.net/78246/analyse/flaechenbedarf-pkw-fahrrad-bus-strassenbahn-stadtbahn-fussgaenger-metro-bremsverzoegerung-vergleich/>

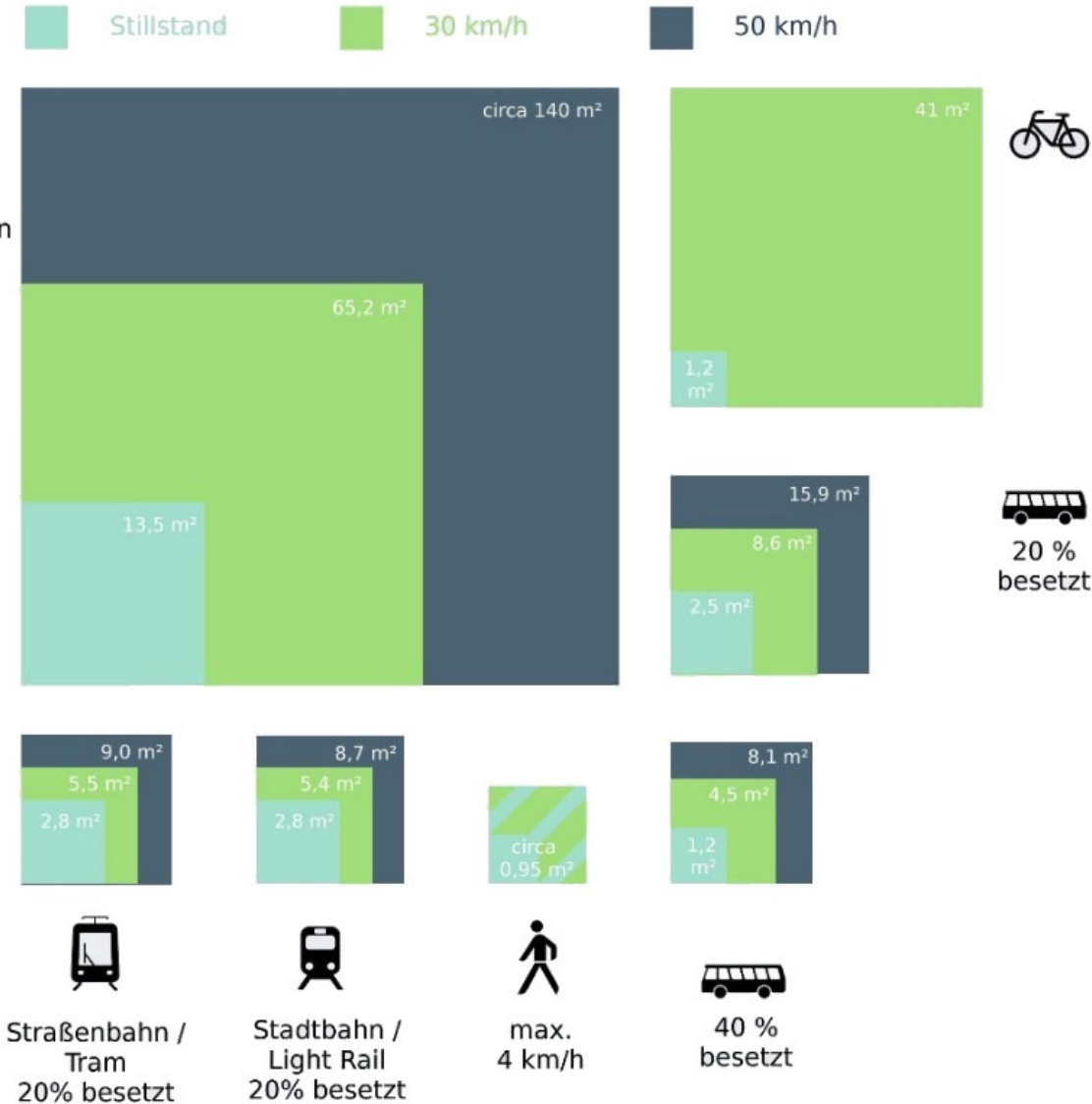
Gesamtfahrleistungen nach Kraftfahrzeugarten



Berechnungsanpassung



mit 1,4
Personen
besetzt



Einleitung

Motivation

- Landflucht & Mobilität steigt in Deutschland
- Problematik des KFZ Verkehr
 - Verkehrsstau
 - Umweltverschmutzung
 - Flächenverbrauch
- Probleme mit E-Scootern und Verwandten Konzepte

Lösung: Die Mikromobilitäts Plattform Scoomatic

Projektziel

Ausgangslage

- Umgebautes Hoverboard und alternativer Firmware
- Umgebungssensoren
- Raspberry Pi 3B
 - Mit Ubuntu 18.04 & ROS1 Melodic und ROS2 Crystal
- ROS1 und ROS2 parallel mit ROS Bridge
- Steuerung mit Gamepad und Joystick möglich



Projektziel

- Bestehende Softwarearchitektur vollständig auf ROS1 umbauen
- Vorhandene Sensoren zur gleichzeitigen Lokalisierung und Kartengenerierung verwenden
- Wegpunkte in Karte abfahren
- Hindernisse erkennen und vermeiden

Projektziel

Verwendungsvorgang

1. SLAM Kartenerstellung
2. Fahrzeug in Karte Lokalisieren
3. Ziel-Punkt festlegen
4. Fahrzeug fährt zu Ziel-Punkt

Sensoren

Verfügbare

- Steuerung
 - **Gamepad** EasySMX 2.4Ghz Controller
 - **Joystick**
- Raumdaten
 - **8 Ultraschall Sensoren** HC-SR04 (*2cm bis 3m & bis zu 50 Samples/s*)
 - **2D LIDAR** RPLidar A1 (bis zu 10 U/s, bis zu 8000 Samples)
 - **IMU** MPU9250 (9DOF)
 - **GPS** Ublox SAM-M8Q
- Motor-/Hoverboarddaten
 - **Odometrie**
 - (Diagnose Daten)

Sensoren

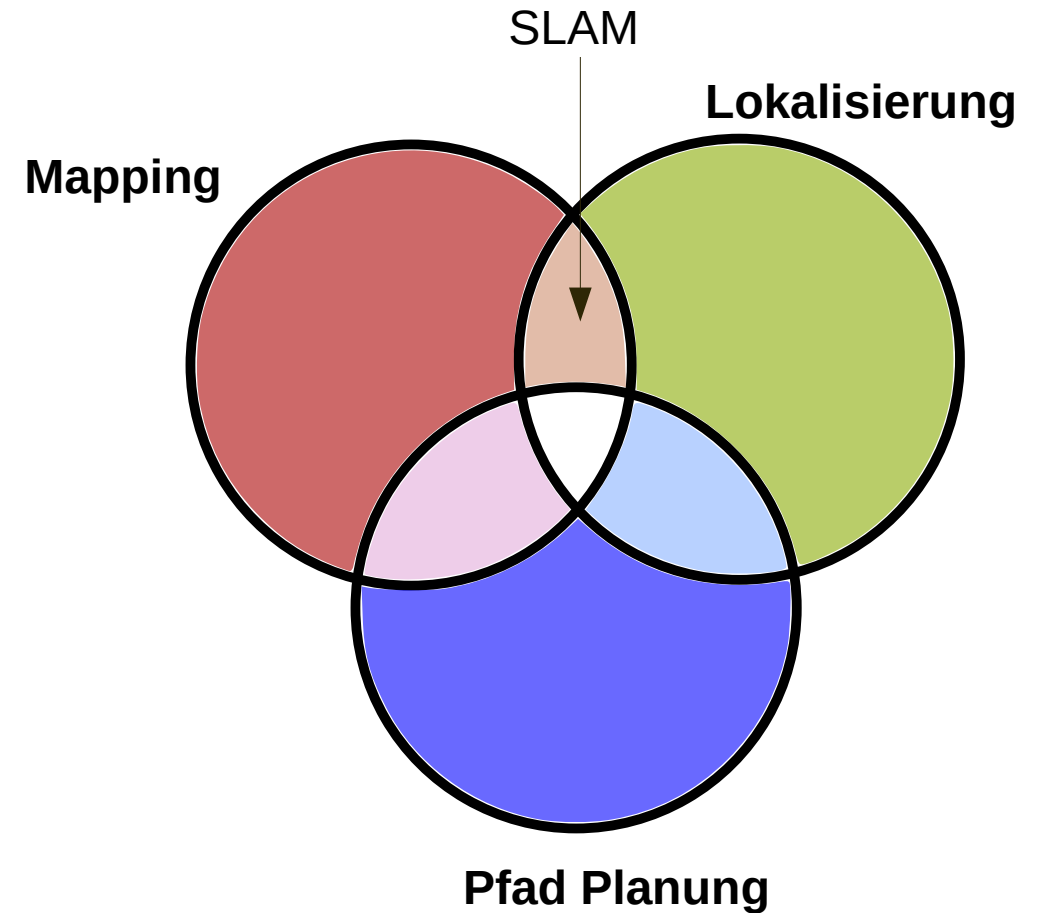
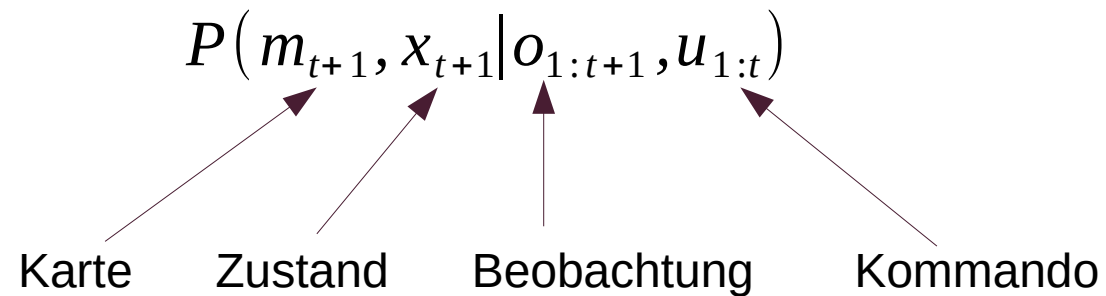
Verwendete

- Steuerung
 - **Gamepad**
 - Joystick
- Raumdaten
 - 8 Ultraschall Sensoren
 - **2D LIDAR**
 - IMU
 - GPS
- Motor-/Hoverboarddaten
 - **Odometrie**
 - (Diagnose Daten)

SLAM

Begriffsbedeutung

- Simultaneous Localization and Mapping
- Problem:
 - Lokalisierung und Karte sind unbekannt
- Inkrementelle Kartenerstellung



SLAM

Verfügbare ROS Packages

- **hector_slam**
- gmapping
- cartographer
- tiny_slam
- ohm_tsd_slam
- mrpt_icp_slam_2d
- mrpt_rbpf_slam
- mrpt_graphslam_2d
- mrpt_ekf_slam_2d
- orb_slam2_ros
- slam_karto
- ...

SLAM

HectorSLAM

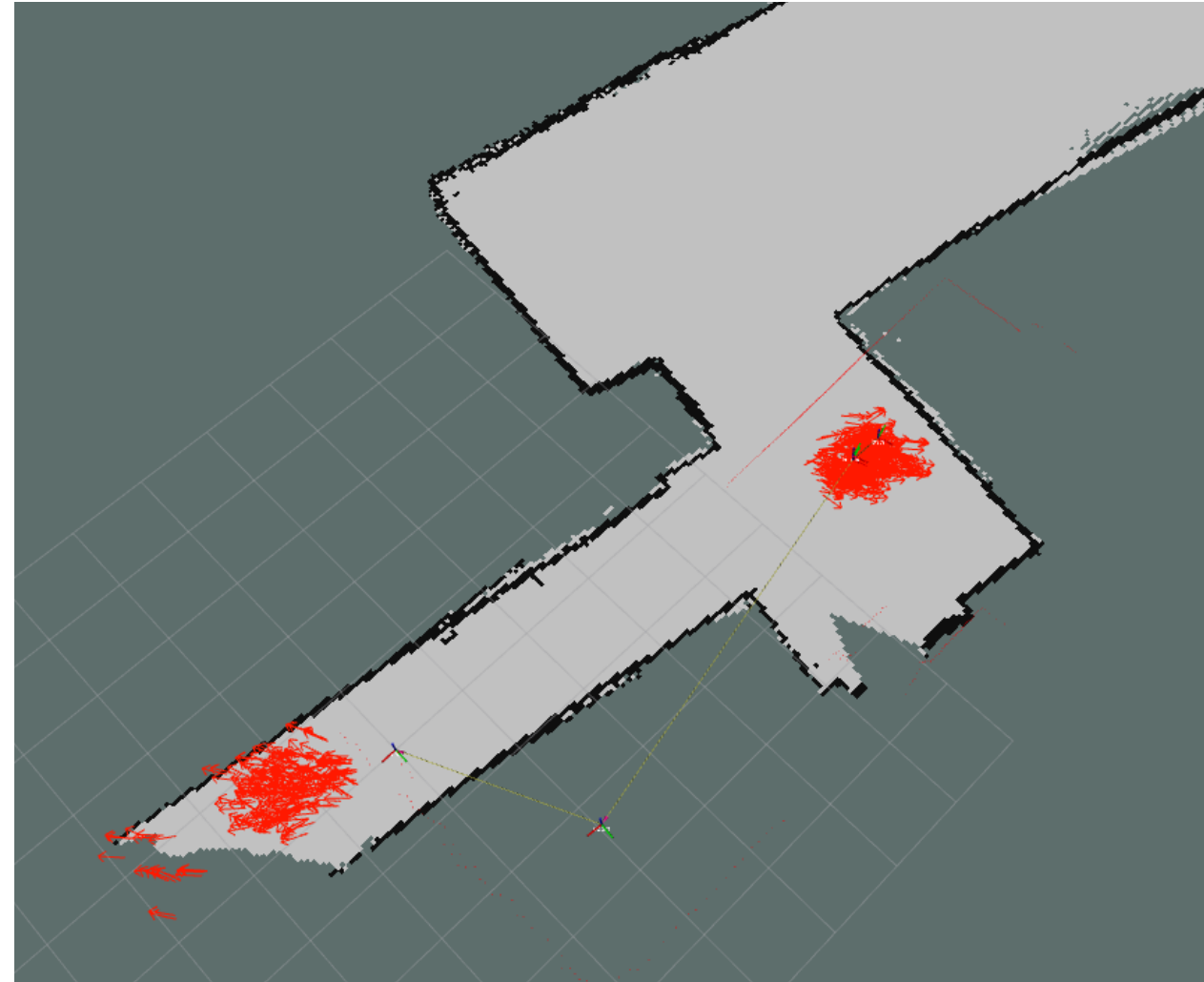
- Ausgewählt, weil
 - Breiter Community Support
 - Verfügbar für ROS Melodic
 - Odometrie & IMU optional
 - Geringe Rechenkapazität
 - Aktive Weiterentwicklung
- Für USAR gedacht
 - Urban Search and Recovery
- Hier: mit LIDAR und Odometrie



Lokalisierung und Navigation

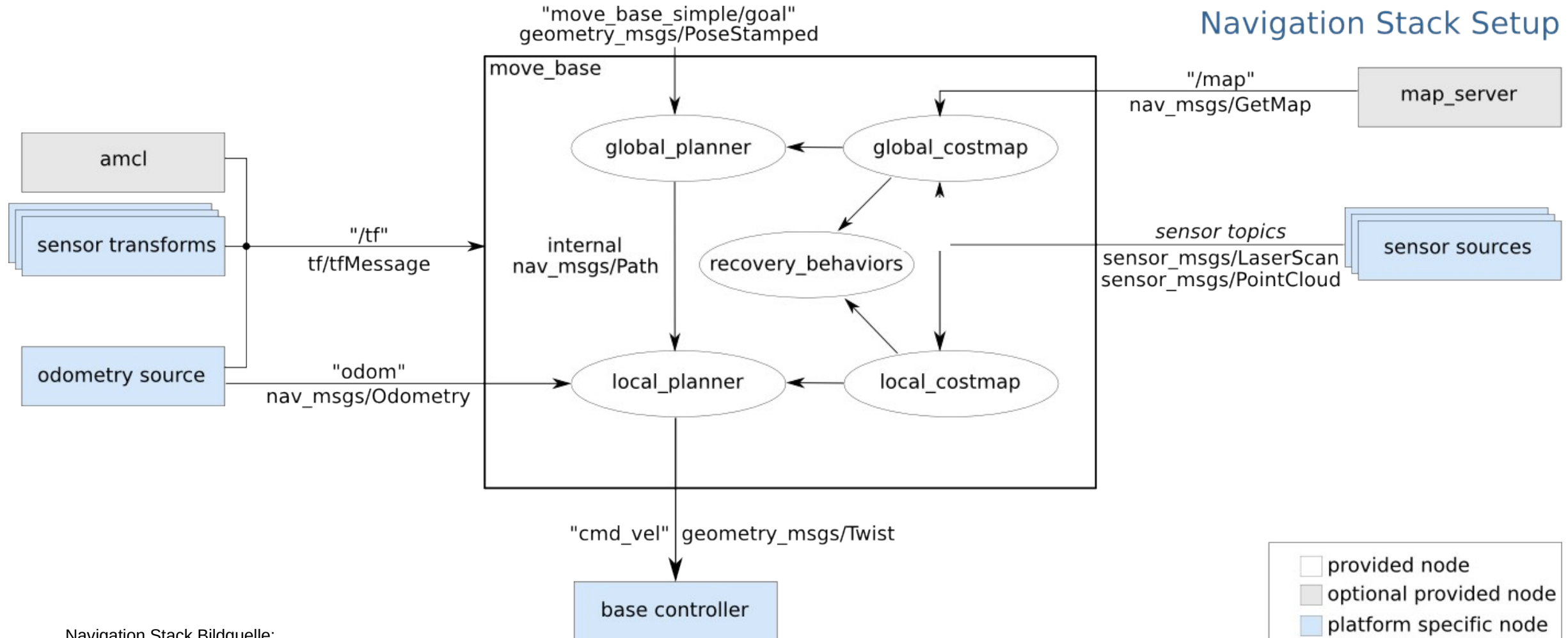
Lokalisierung

- AMCL zusammen mit Navigation Stack (move_base)
 - De facto Standard
 - Nutzt LIDAR und Odometrie
- AMCL
 - Adaptive Monte Carlo Lokalisierung
→ Partikelfilter



Lokalisierung und Navigation

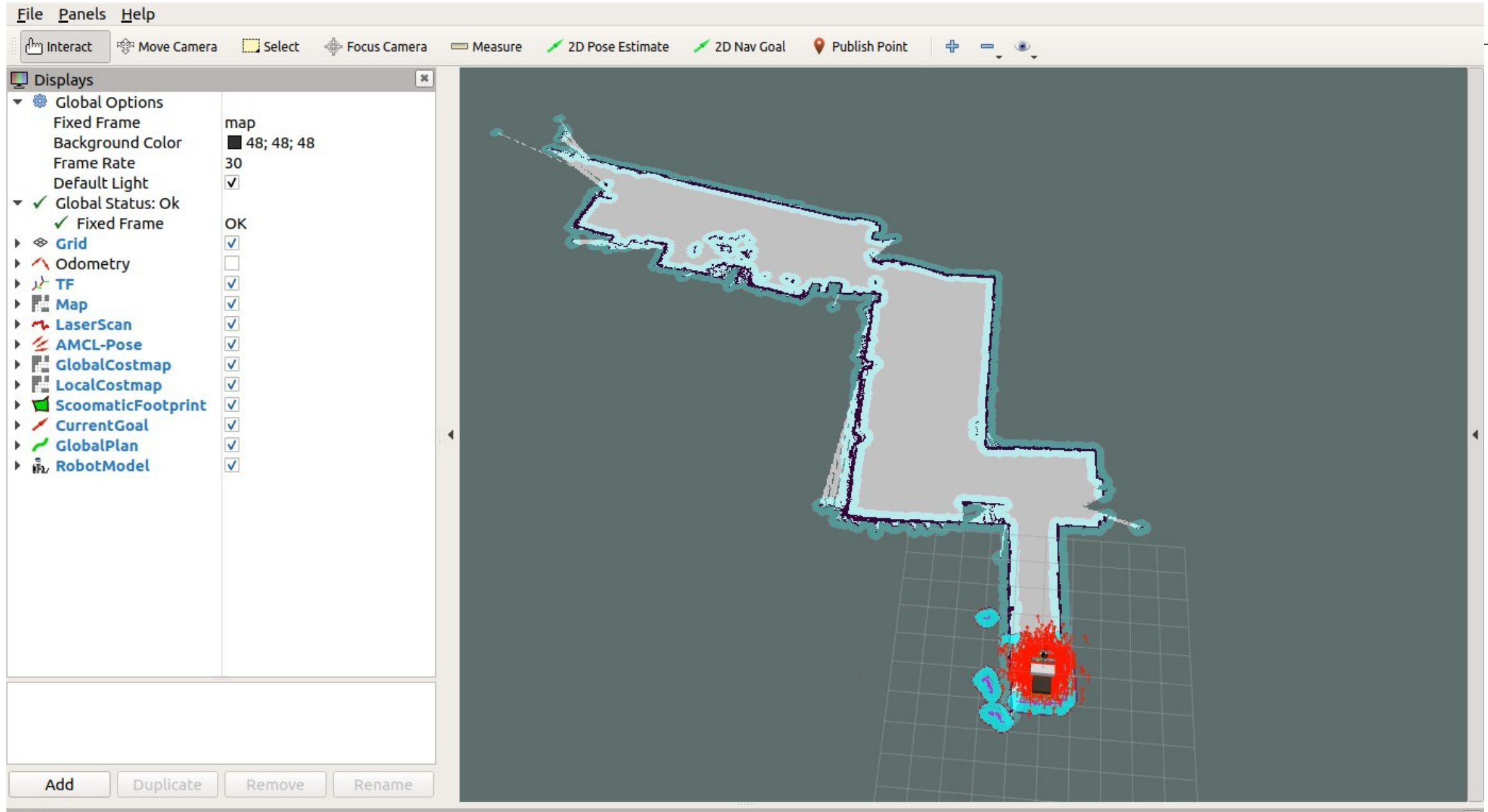
Navigation



Navigation Stack Bildquelle:
http://wiki.ros.org/move_base

Lokalisierung und Navigation

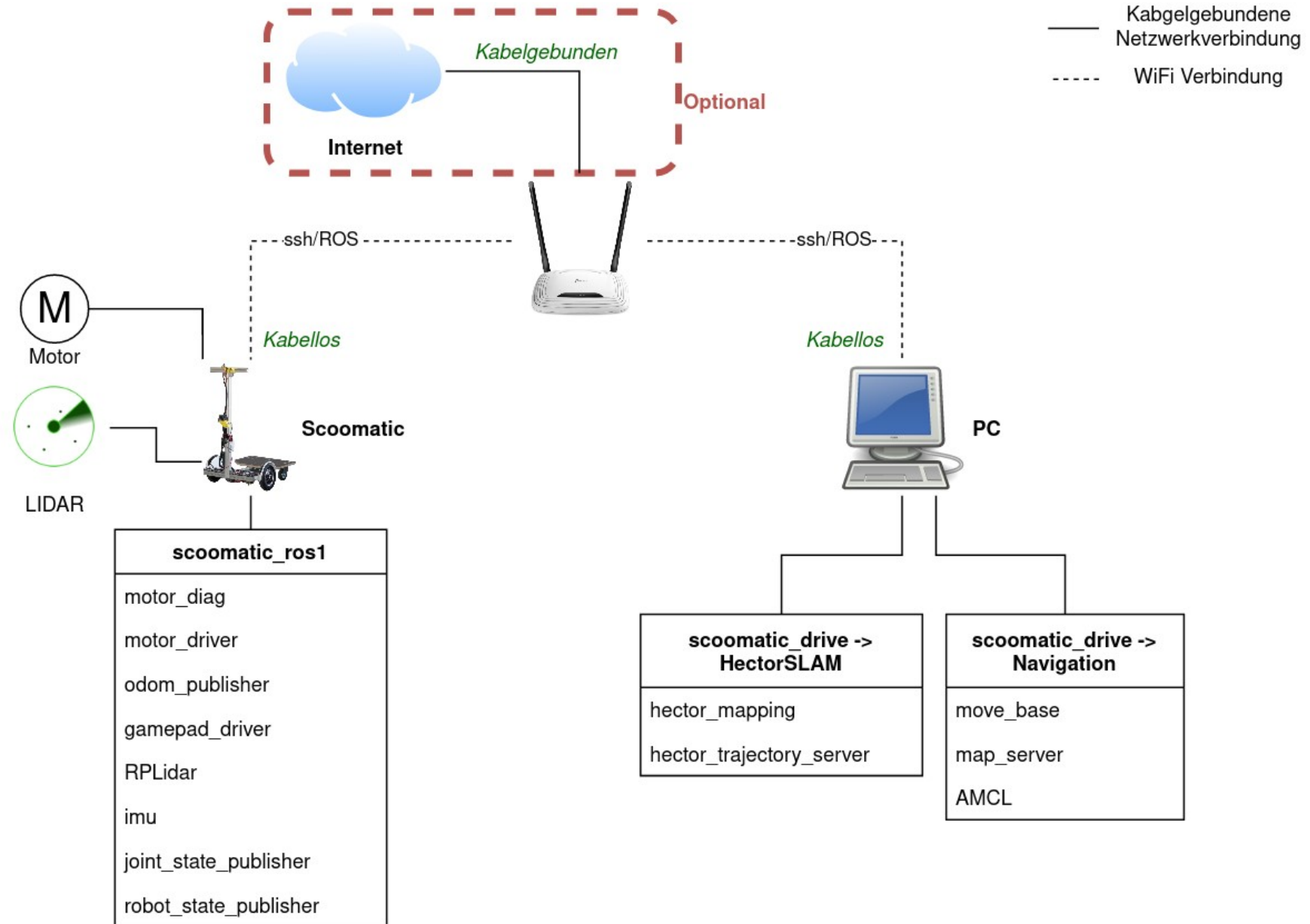
Navigationsansicht in RViz



Systemaufbau

System-Netzwerkstruktur

- Allgemeine Struktur
- Access Point entweder Eigenständig oder von PC aus
- Internet notwendig für Software Update



Systemaufbau

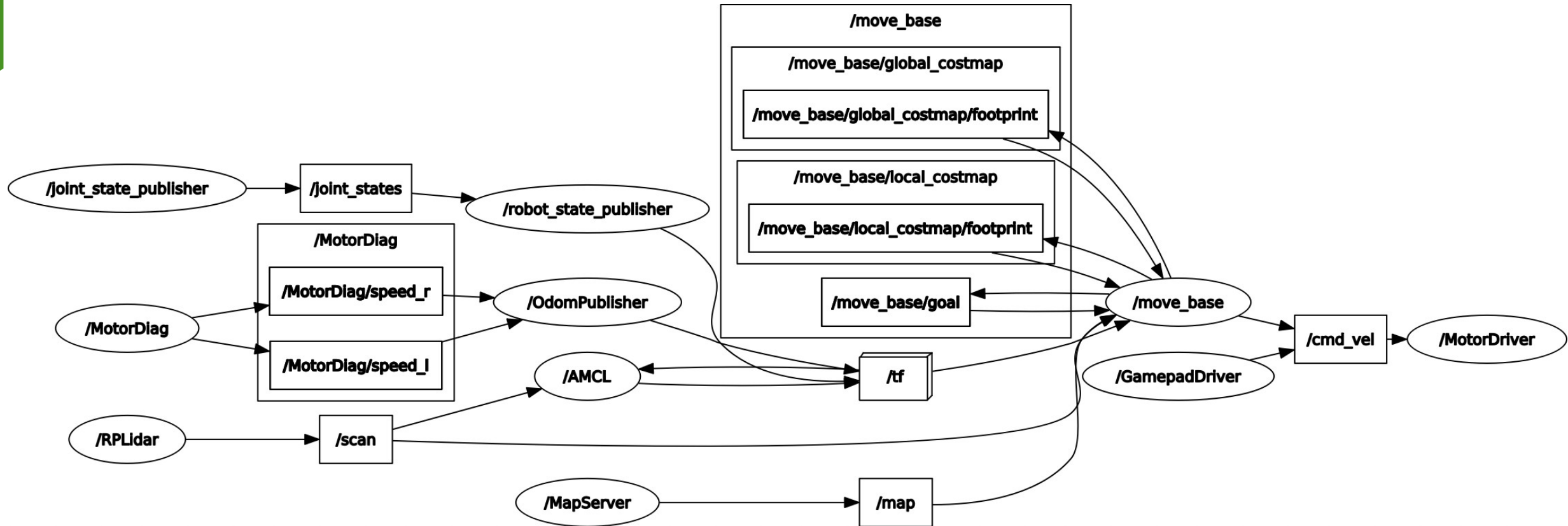
Realer Aufbau

- WLAN Hotspot (Access Point) läuft auf Laptop
- Vollständig mobil
- Gamepad Steuerung bzw. autonome Navigation



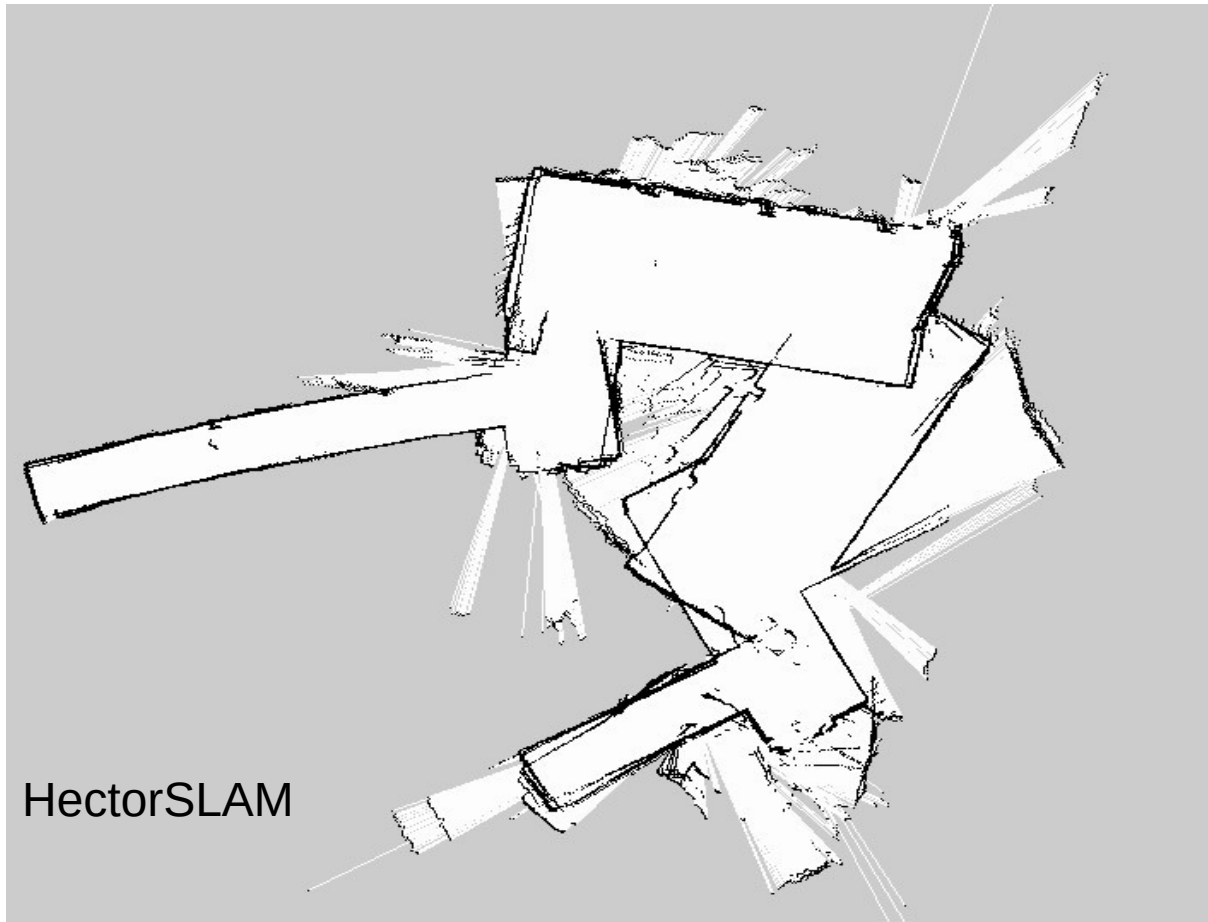
Systemaufbau

ROS Nodes und Topics bei Navigation

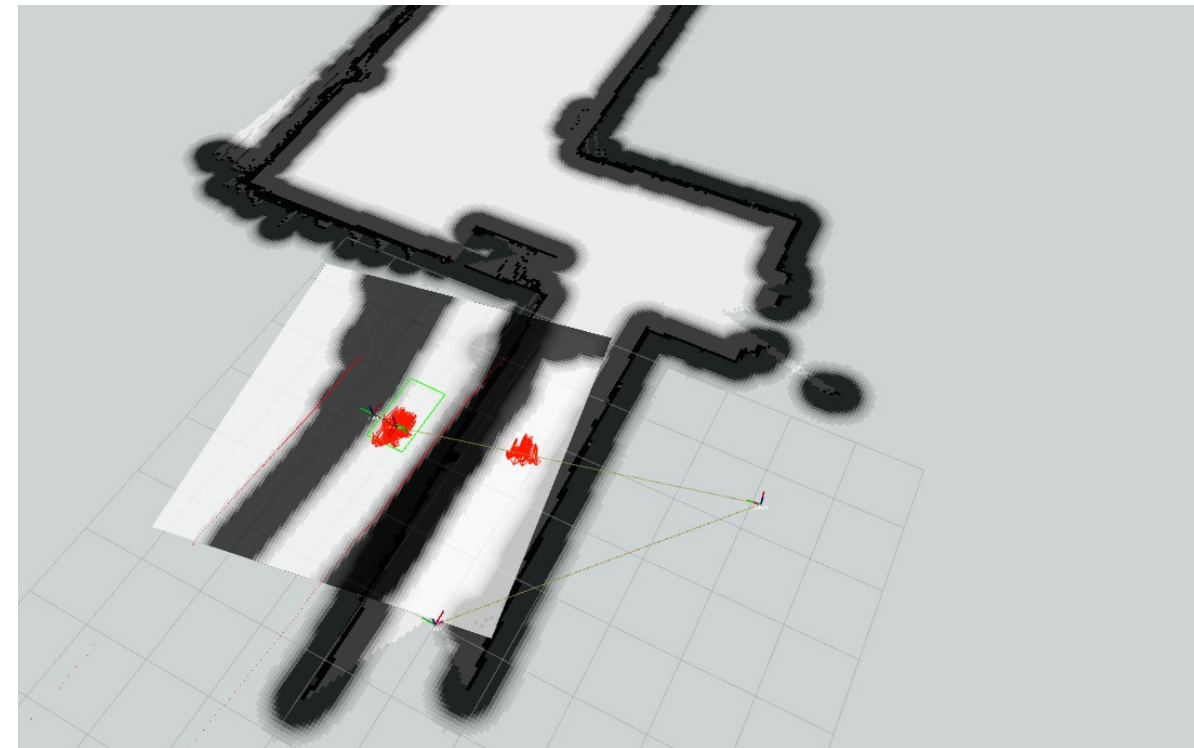


Testing

Negativ Beispiele

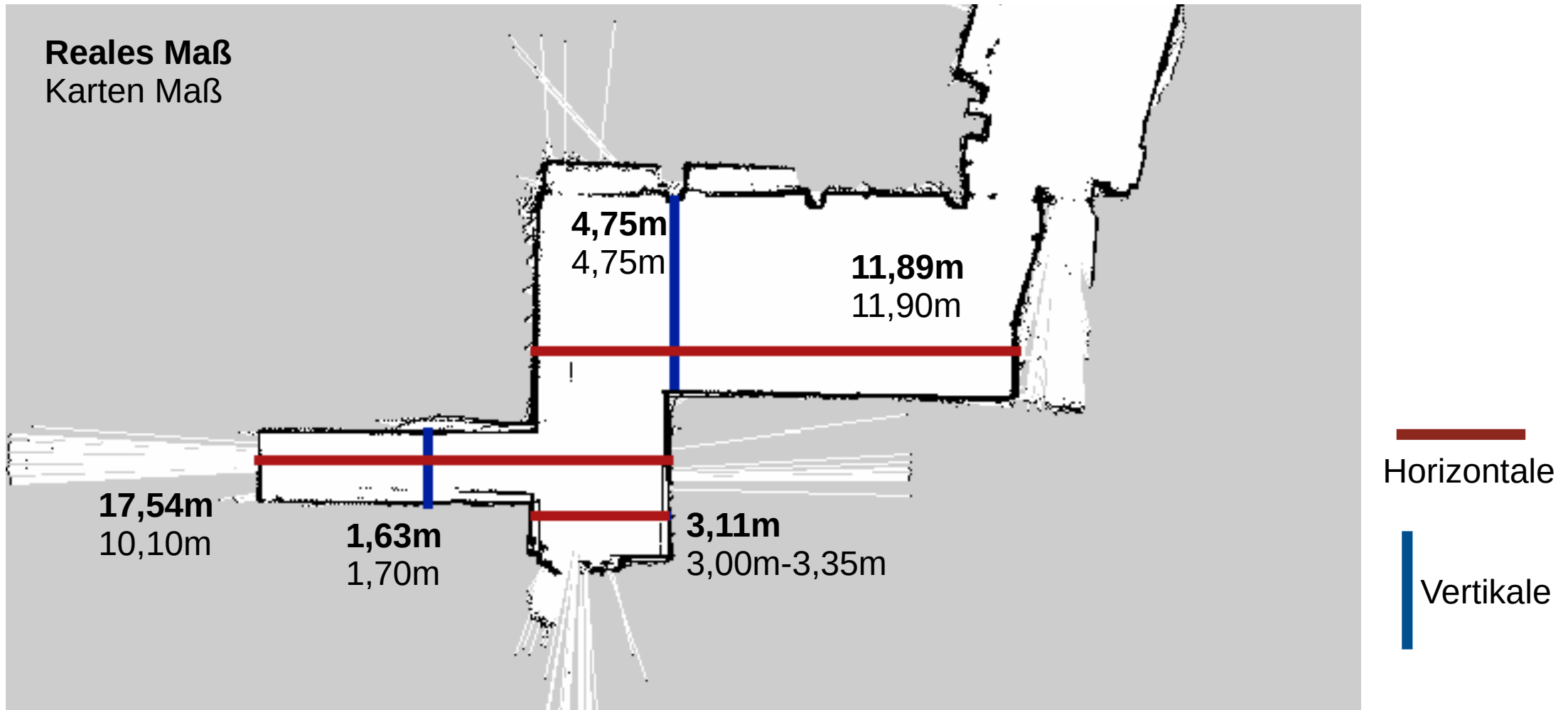


AMCL



Testing

Kartenanalyse



Fazit

Erreichte Ziele

- ROS1 Migration
- SLAM Kartenerstellung
- AMCL Lokalisierung
- ROS Navigation
- „Odometrie“-Bereitstellung von Motor

Bestehende Probleme

- HectorSLAM ungeeignet
 - 2D SLAM
 - Für kleine Umgebungen
 - Kein Loop Closure
 - Karte nachträglich nicht erweiterbar
 - Glastür-Problematik
- Rechenleistung
- Outdoor Fähigkeiten
 - LIDAR Range gering (12m)
 - GPS noch nicht funktionsfähig
- Motor „Odometrie“
- IMU Daten Bereitstellung
- Sicherheit (Notstopp)

Ausblick

Verbesserungsvorschläge

- Bessere Rechenleistung → z.B. Nvidia Jetson
- RGBD Kamera → Visual SLAM
- 3D LIDAR → 3D SLAM
- Automatisches SLAM → Exploration
- Motor Geschwindigkeits-Regelung
- Odometrie Alternative → Wheel Encoder o.ä.

Vielen Dank
für Ihre Aufmerksamkeit

