

# Konzeptionen innerhalb der Robot Operating System (ROS)-Umgebung im geodätischen Zusammenhang

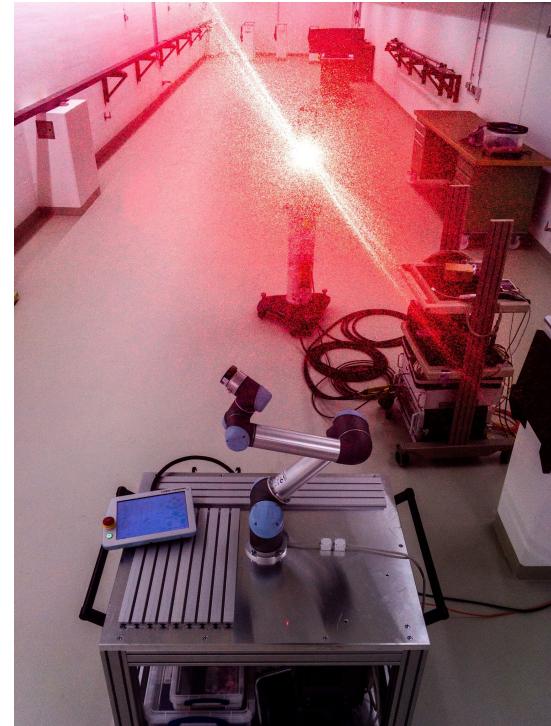
Ausgewählte Kapitel Ingenieurgeodäsie

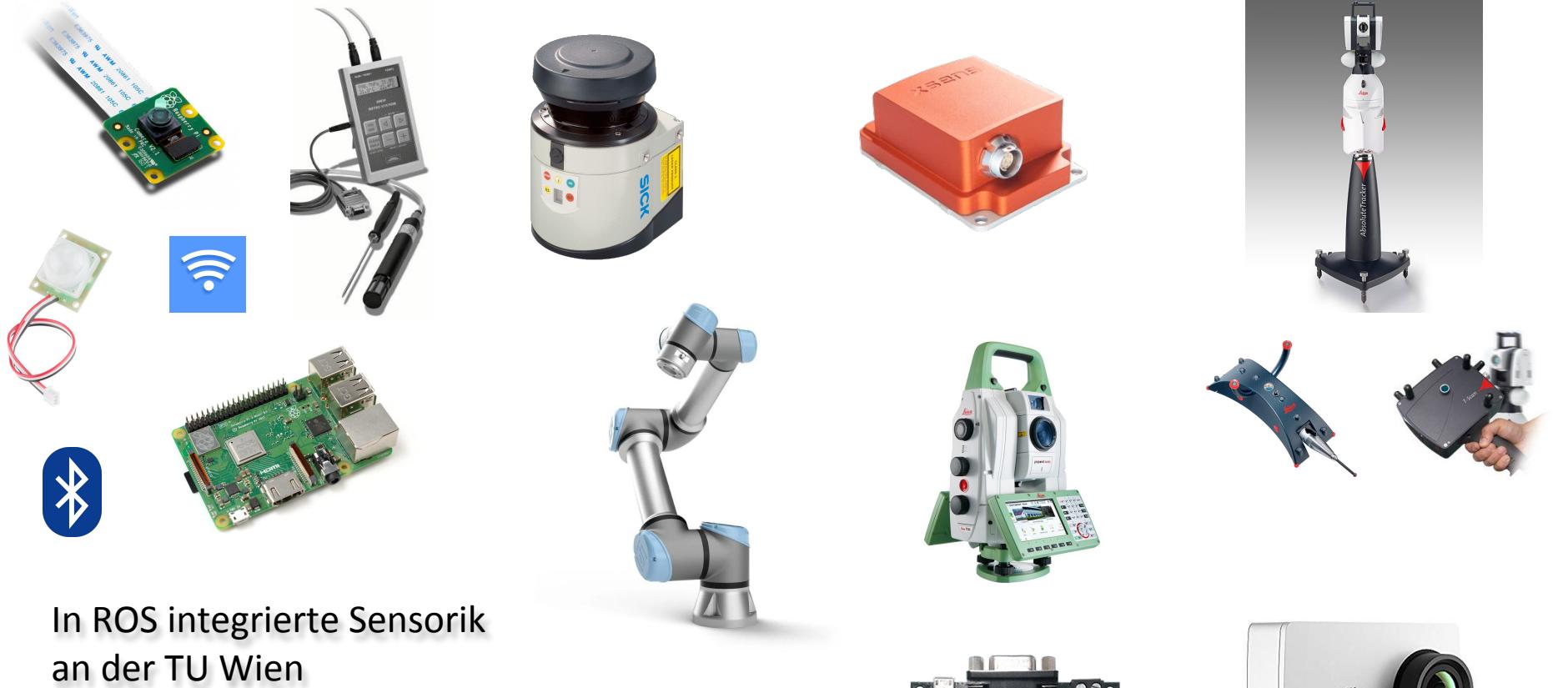
# Timetable

- 1. Block
  - Einführung ROS, Vorstellung Paper
  - Präsentation und Begrifflichkeiten
- 2. Block
  - ROS-Master Anmeldung, Linux
  - Python “Hello World”, installation
  - Rosbags
- 3. Block
  - ROS Publisher und Subscriber
  - Git
  - Auswerten der Rosbags

# Motivation

- Schnelles Erfassen von räumlichen Strukturen
- Modularer Aufbau
- Effizientes Testen und Umsetzen von innovativen Ansätzen
- Abstraktion und “Standardisierung” von Messsensorik

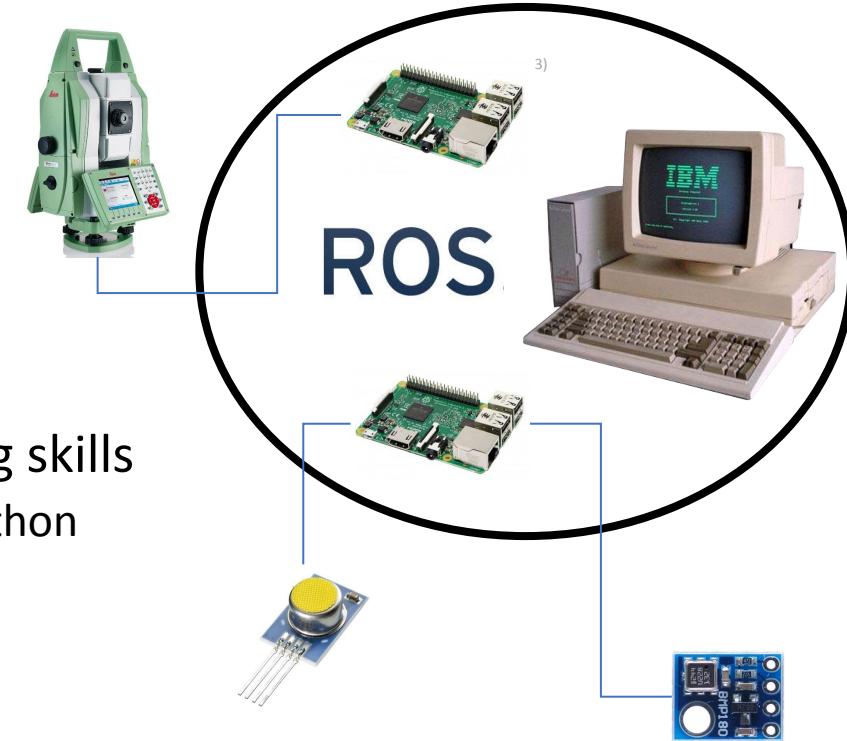




In ROS integrierte Sensorik  
an der TU Wien

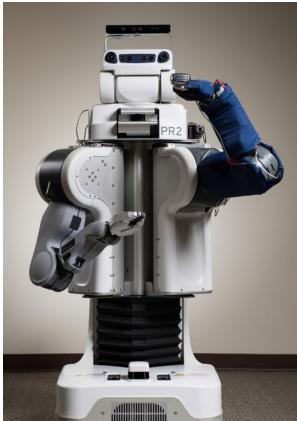
# ROS.org

- Modular
  - 11700 Packages, free to use
- Can be used without any programming skills
  - Further development in either C++ or Python
- Open-Source
  - Constantly growing community
- Manufacturer awareness
  - Some manufacturers are providing ROS packages for their products

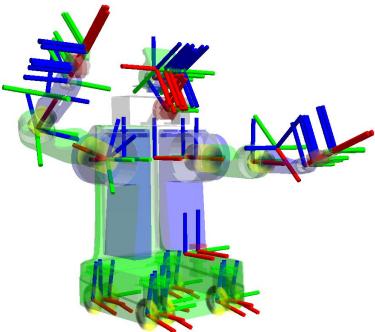


# ROS Projekte - Roboter und Algorithmen

- PR2-Roboter ist vollständig über ROS steuerbar
- Alle Funktionalitäten die ein Roboter benötigt werden unterstützt

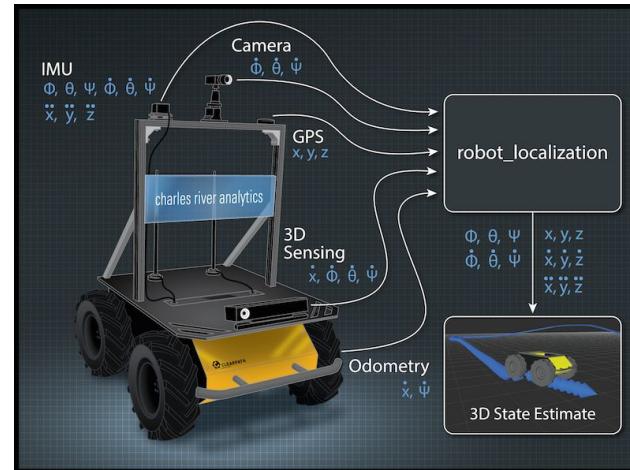


[1]



[2]

- Viele verschiedene Sensoren können mit ROS angesprochen werden
- Das gilt im großen Umfang auch für die Sensorik der Geodäsie



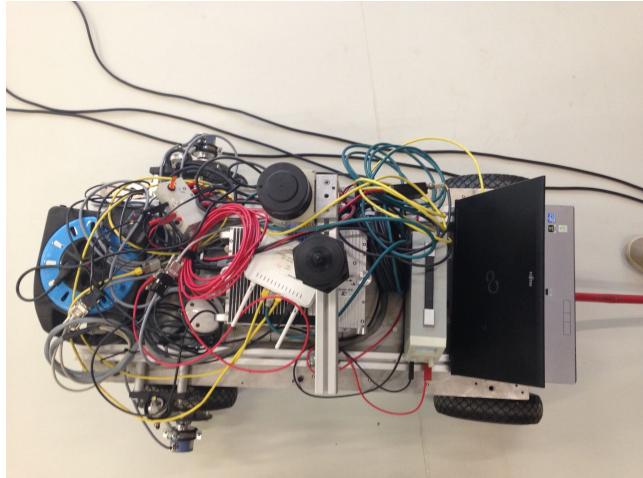
[3]

# ROS - Robot Operating System

Ermöglicht den modularen Aufbau komplexer Strukturen in der Robotik

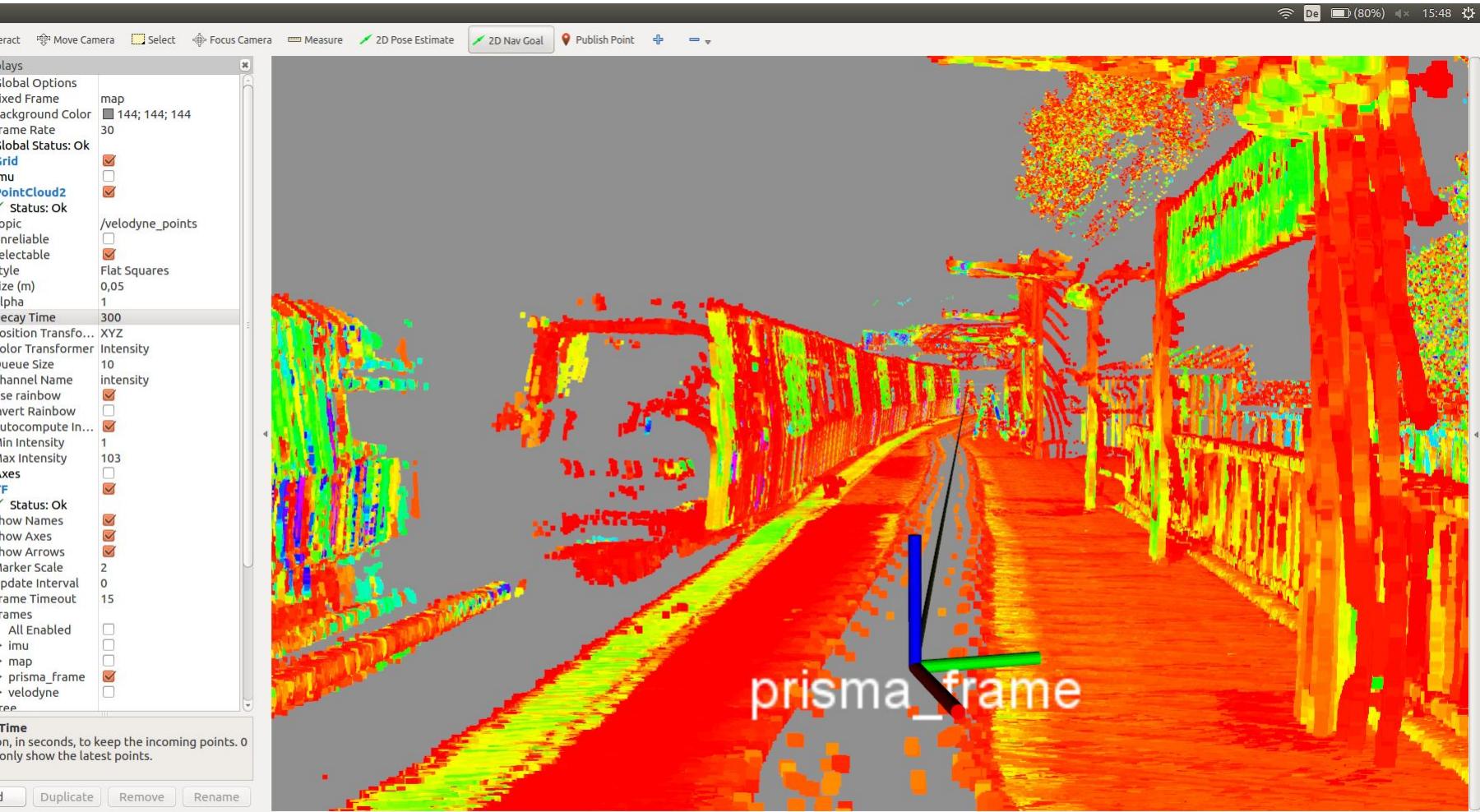


- Linux
- Abstraktion
  - Einfach
  - Modular
- Debugging
  - Schnell
  - Flexibel
- Datenverarbeitung



# Geodetic ROS components





# Projektziele

- **Sensornaher Programmierung**

C, C++, Python, Bash, Socket, UDP,  
Pointer, Treiber, Linux, Open Source



*“Wir bekommen die Daten, die wir haben wollen.”*



- **Flexibles Framework**

Kapselung Hardware & Algorithmen,  
Foren, Community, Hersteller



*“Neue Ansätze werden auf ihren Nutzen hin überprüft.”*



- **Fachbezogene Entwicklung**

Anwendbar von ungeschultem Personal  
Debugging, Dokumentation, Testumgebung



*“Wir haben für jeden etwas zu tun.”*



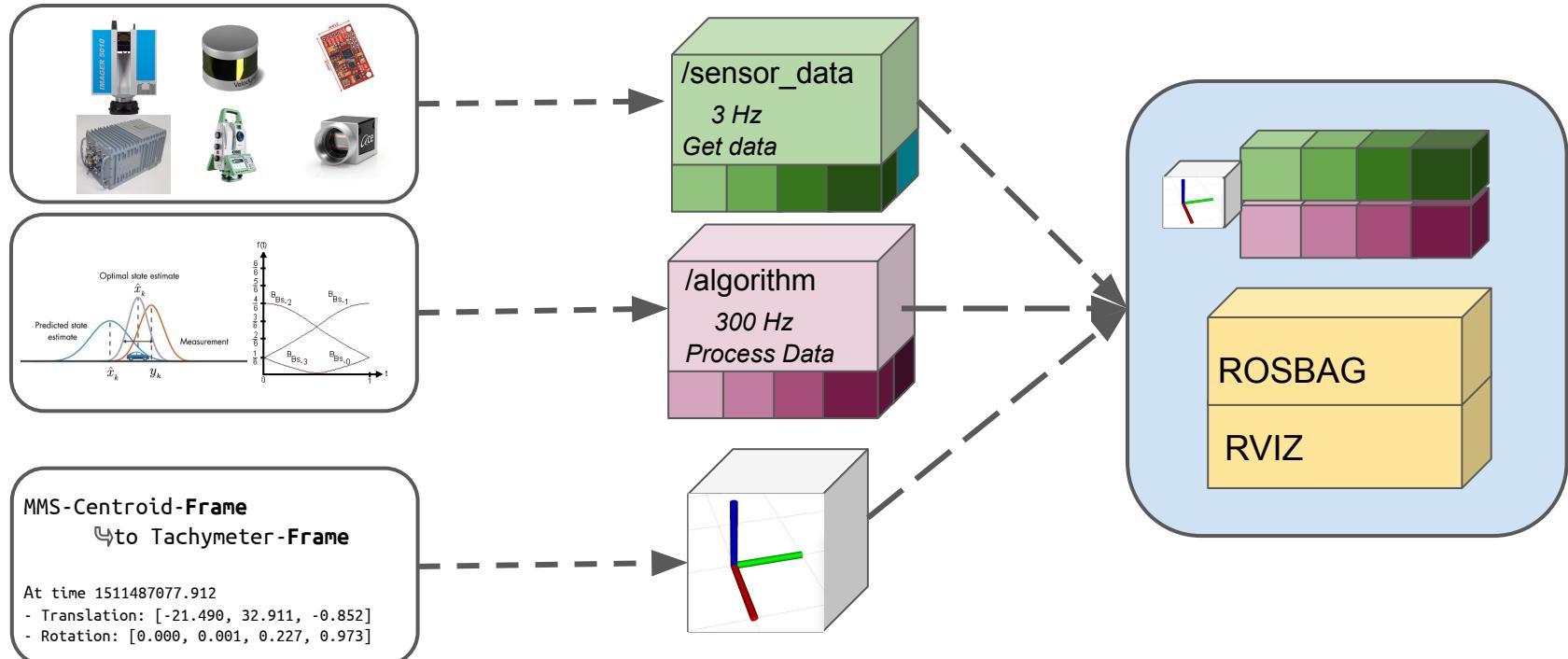
- **Schnittstellen öffnen**

Einbinden häufig verwendeter Protokolle  
Einbinden geodätischer Messinstrumente

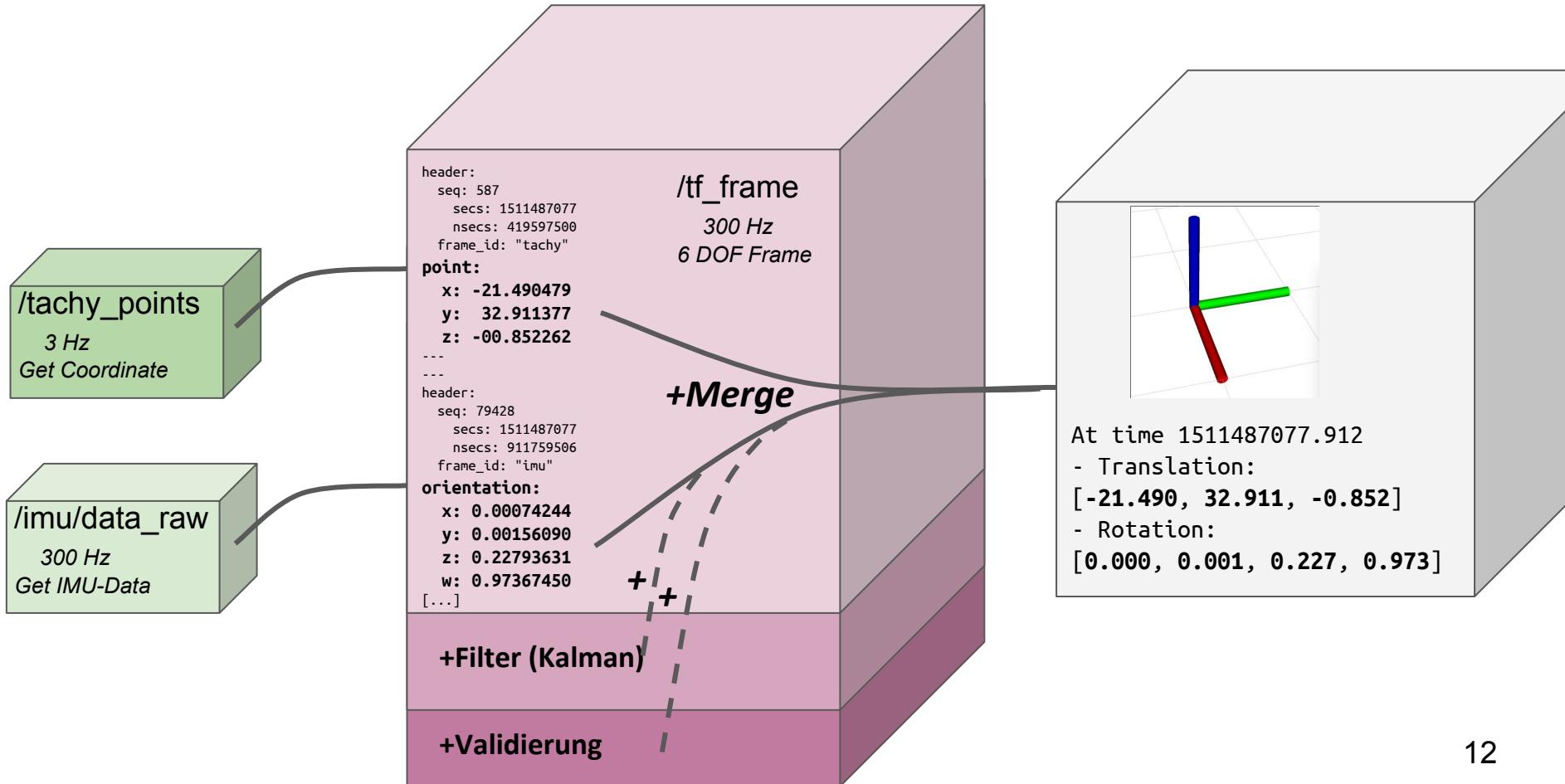


*“Wir können (alle) Messinstrumente problemlos einbinden.”*

# Modularität



# Entwicklung Interface



# HCU\_ROS

ROS

## Hardware-abstraktion

/imu/data\_raw  
300 Hz  
Get IMU-Data

/tachy\_points  
3 Hz  
Get 3D-Coordinate

/pcl\_raw  
5 Hz  
Get Pointcloud

## Algorithmen

/tf\_frame  
3 Hz  
Set 6 DOF Frame

/tf\_pcl\_points  
3 Hz  
Referenced Ptcl.

/pcl\_tools  
3 Hz  
Export Ptcl.

\*.txt

## Interprozesskommunikation

Topic	Hz	Daten
/imu/data_raw	300	x, y, z, w
/tachy_points	3	$\Delta x, \Delta y, \Delta z$
/pcl_raw	5	$\Delta x, \Delta y, \Delta z @ pcl$
/tf_frame	3	x, y ,z, w, $\Delta x, \Delta y, \Delta z$
/tf_pcl_points	3	x, y, z @ pcl
/pcl_tools	3	Export x, y, z to *.txt

ROSBAG  
RVIZ

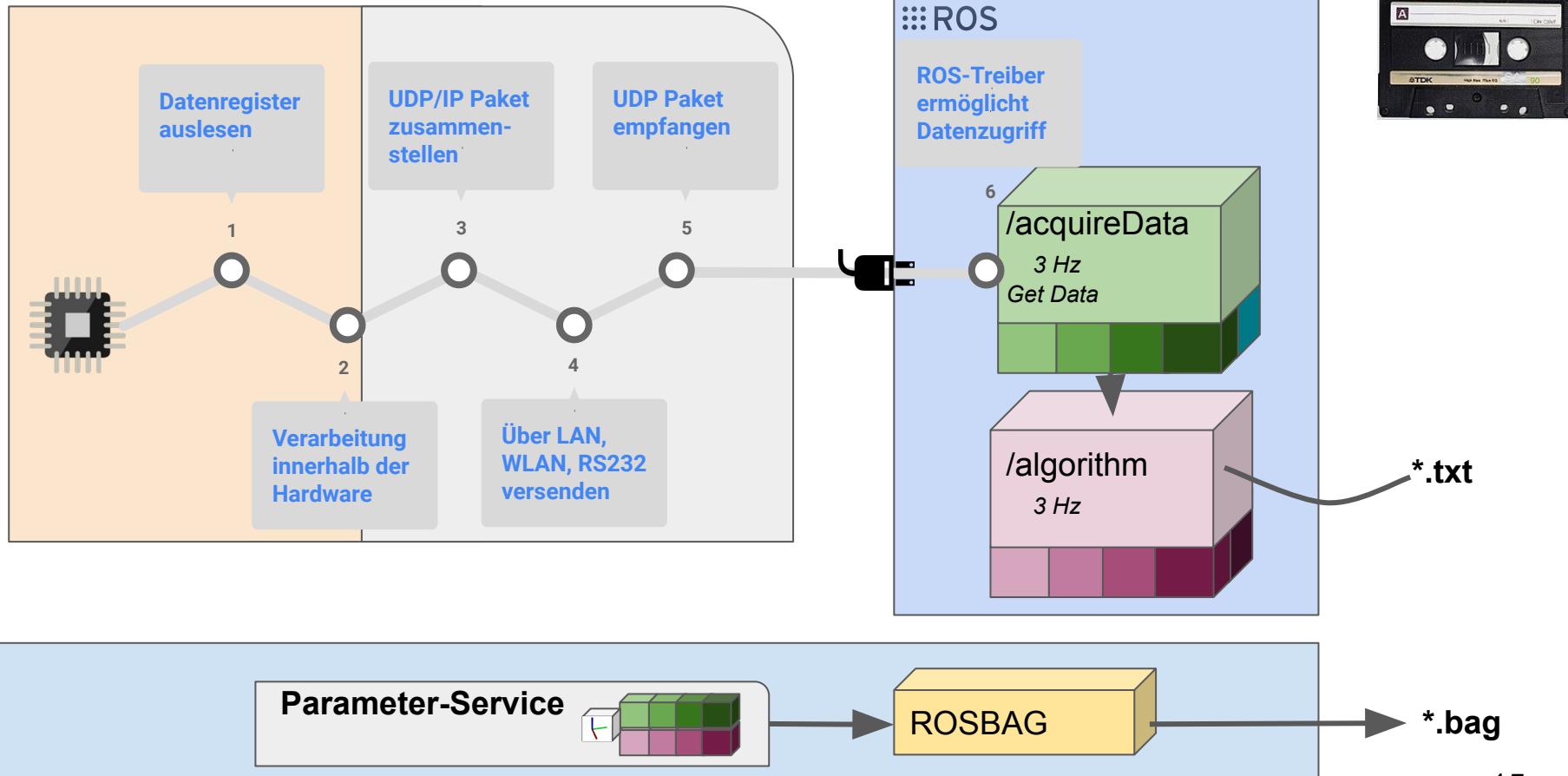
\*.bag

# Usage Levels of ROS

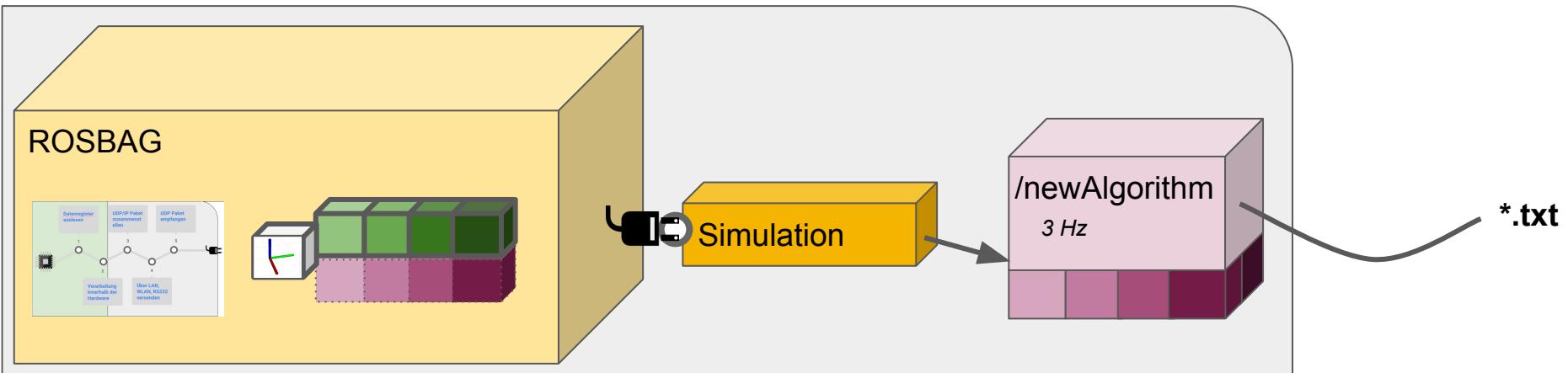
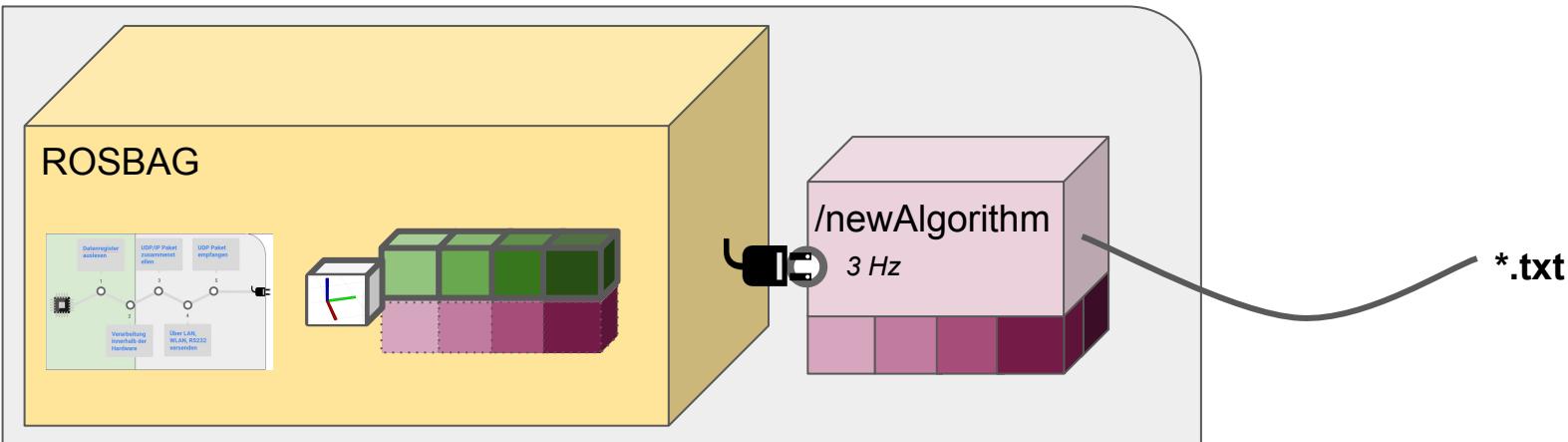
- Standard usage levels of ROS
- Developer Level
- User Level



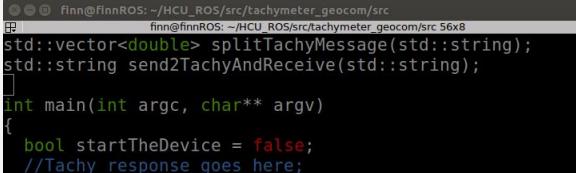
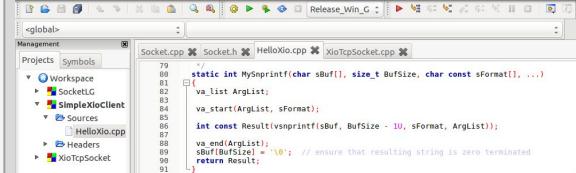
# Protokolle



# Simulation

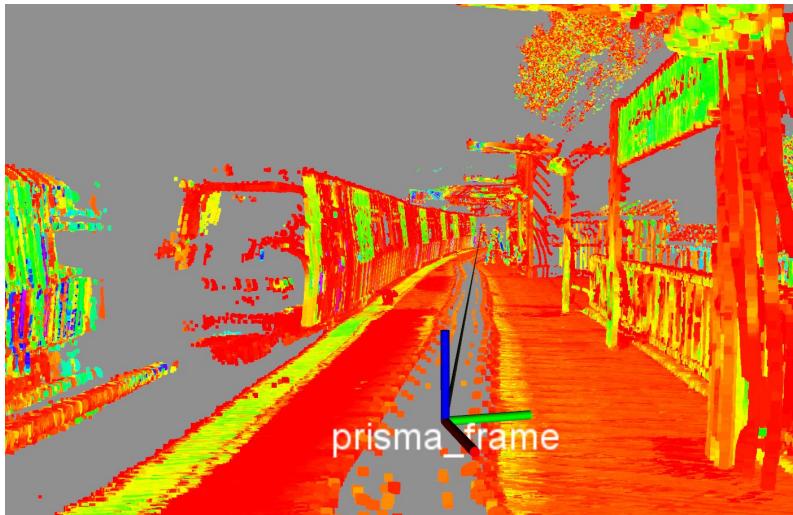


# Implementierung Sensorik Übersicht

Initial development	Code expansion	Packaging
		
<b>Einbindung</b> geodätischer Instrumente durch eigene Codeentwicklung	<b>Erweiterung</b> bestehender Strukturen für den geodätischen Anwendungsfall	<b>Nutzung</b> von bereits zur Verfügung gestellten Implementierungen
		
<ul style="list-style-type: none"><li>• Leica MS50</li></ul>	<ul style="list-style-type: none"><li>• iMAR iNav IMU (INS, GNSS, Odometry)</li></ul>	<ul style="list-style-type: none"><li>• Velodyne 16-VLP</li><li>• Xsens Mti</li></ul>
Trimble environment	Arduino/Raspberry Sensors	Range Finder 1D & 2D, Laserscanner, Speech Recognition, Cameras, Environmental, Force/Torque, Motion Capture, GNSS/IMU

# Fazit & Ausblick

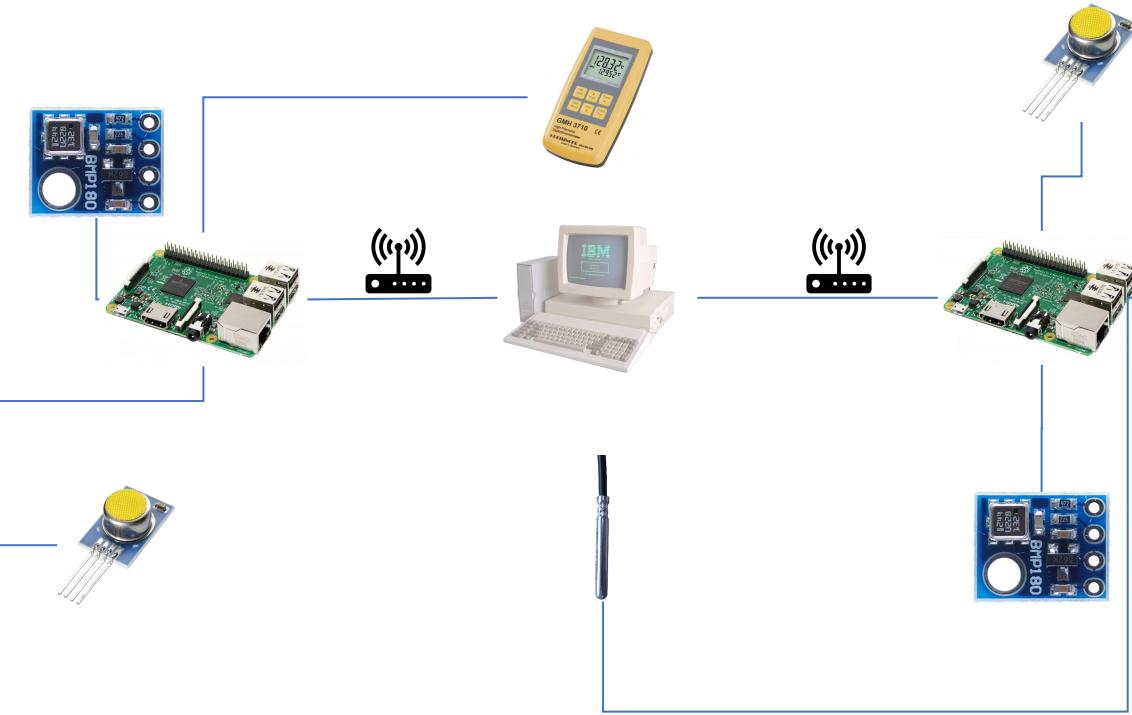
## Robot Operating System



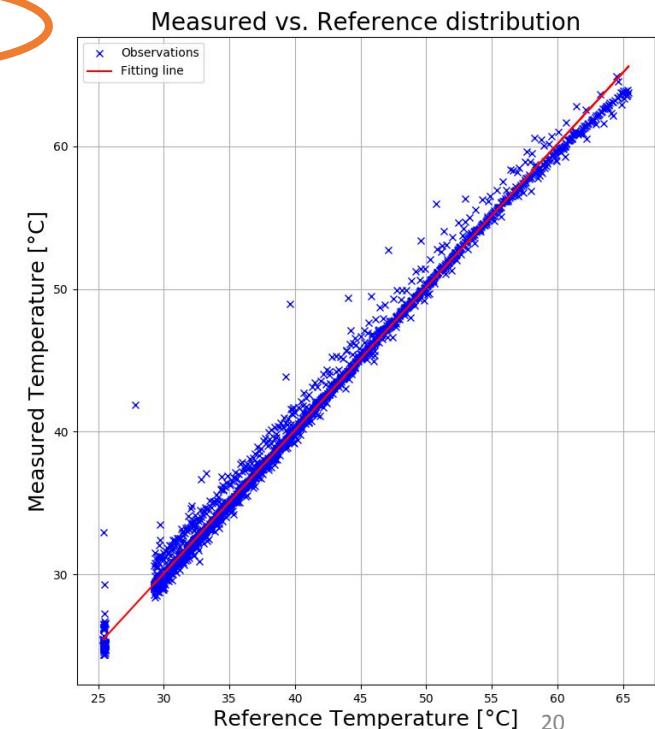
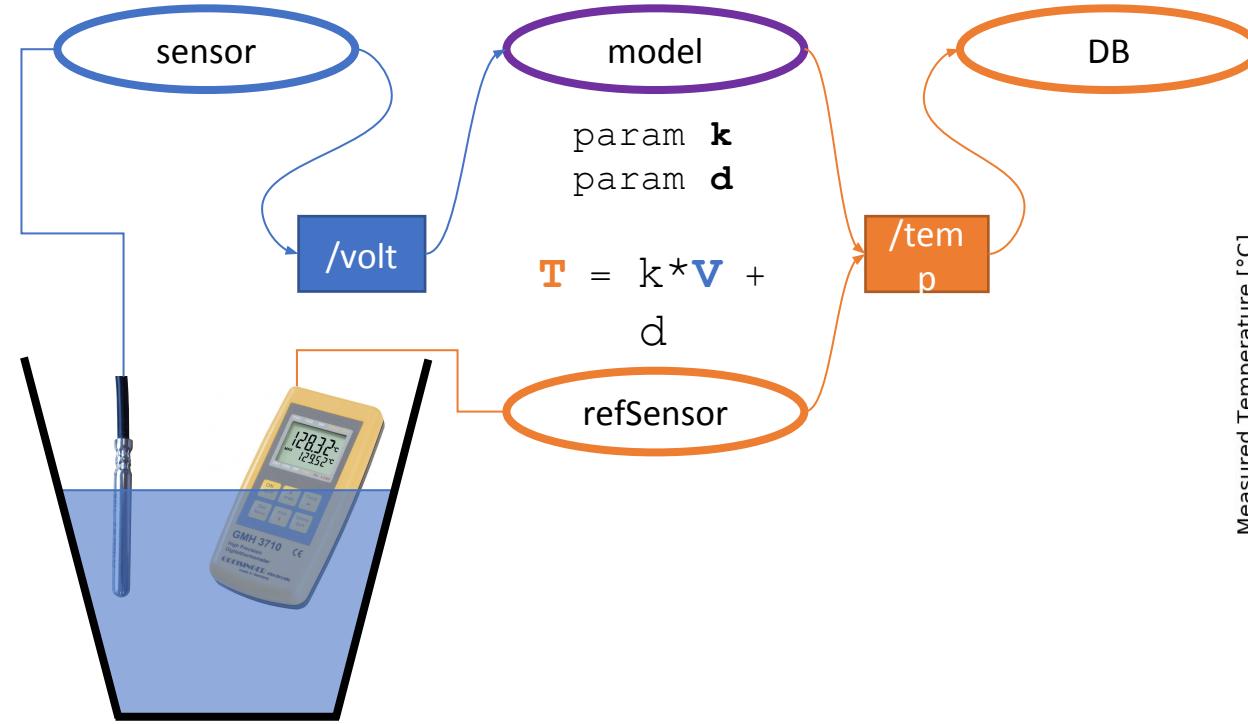
- Vollständige Entwicklungsumgebung vom Objektraum zur 3D-Punktwolke
- Flexibler Einsatz von Algorithmen im Feld
- Kontrollmöglichkeit direkt im Feld
- Struktur für modulare Systemerweiterungen
- Algorithmenentwicklung

# Example: Temperature Sensor Network

- Distance measurement distorted by atmosphere
- Permanent meteorological sensor network
- Interpolation of temperature field
- Calibration of the sensors



# Calibration of an temperature Sensor



# Example: Kinematic Systems using RTS

